Report of Summer 2007 NSERC USRA

Nikhil Jain under the direction of Scott Oser
September 17, 2007

SUMMARY

This report contains descriptions, methods, results and conclusions from my work at TRIUMF during the summer of 2007, as an NSERC USRA recipient. Specifically, this report explains my work regarding noise reduction on M11 beamline data, simulations and mathematical parameterization of MPPC behavior, and simulations of the Light Injection System for the FGD. My simulations indicate that the for the purposes of the Light Injection System, if one expects non-linearity in the LEDs, in addition to Poisson fluctuations in light output and Gaussian fluctuations in the power supply, approximately 30 light levels need to be calibrated for the LEDs, and each light level will need to be flashed approximately 300 times. These values are obtained under the assumption that a maximum of 5% error tolerance is acceptable in reconstructing the number of photons incident on the MPPC from the charge output of MPPC. Simulation results further indicate that photon resolution in the under-300 photon level (20-25 photoelectrons) shall be difficult.

1 – M11 Beamline Data Analysis

1.1 – Introduction

To better understand the properties of the wavelength-shifting fibers (WSF), the M11 beamline was directed at different points along a 2x4 array of scintillator bars, each through which a WSF was threaded. The beam was repositioned at different locations along the bars in order to measure the light yield of bars and WSFs, thereby collaboratively yielding the attenuation length. Data was read in through a combination of ADCs, Flash ADCs, and TDCs. This was done under the direction of Peter Kitching and Stan Yen. My work related to performing noise reduction on the collected data, so as to be able to better distinguish photoelectron peaks in charge histograms, through modification of the data analysis program designed by Konstantin Olchanski.

1.2 – Problems

The original version of the analysis examined waveforms of current vs. time obtained by the flash ADC. The code searched for the absolute lowest point in the waveform, as depicted by the circle in Figure 1-1, and added the value of this peak to a histogram filled with all the other peaks. Two major problems exist with the straightforward algorithm of finding the absolute peak height for each waveform. Firstly, the baseline of the current was shifted above or below the 0 mark differently for each event's waveform. This resulted in each peak being shifted randomly up or down by some amount, thereby distorting the collected peak height, which the program collected in absolute terms, not relative to the baseline. Secondly, there exist a number of 'dead' events in which no actual pulse is present, only noise. The brute-force algorithm used for peak finding then simply locates the lowest point along the electronic noise and counts that as a peak. Figure 1-2 shows a typical resulting histogram from this method of peak finding. We note that although distinct PE peaks are visible in this histogram, there exists a double peak at the lower explained by the dead events. Furthermore, it is conceivable that the spread of the peaks have been increased due to the baseline fluctuation, and, in fact, the current peaks in Figure 1-2 should be much narrower. A methodological problem also exists with the peak finding method, in that it obtains a value for current. Since the MPPCs release charge over some time, it is more valuable to obtain the total current released by

the MPPC, through integration of the waveform.

1.3 - Modifications to the Original Program

Taking into account these problems, upon the advice of Dr. Kitching, I modified Konstantin's analysis program to address these problems. Foremost was modifying the peak finder algorithm. The new version of the analysis program measures each peak relative to the baseline for that waveform. Also, I noticed a small sinusoidal fluctuation along the baseline and, upon the advice of Dr. Oser, considered this as electronic noise and used a fitting algorithm to subtract this from the waveform. This latter subtraction unfortunately resulted in a large increase in processing time for the program due to the computation-heavy nature of fitting. The combined fitting formula used for locating the baseline and for fitting the sinusoidal noise was of the form: $0.6\sin(Ax)+B$, where A and B are fitting variables. When these noise and baseline-adjusted peaks were located and plotted on a histogram, the resulting graph appears much clearer than the one generated by the original version of the program. The resulting graph, for the same data set, is shown in Figure 1-3. One can notice three particular improvements over Figure 1-2. Firstly, the double peak caused by dead events has disappeared. Secondly, the variance has slightly decreased for each Gaussian shape. Thirdly, we note that the first peak now has a higher frequency than the second, and the second greater than the third, and so on. This was to be expected as these current peaks are caused by dark noise and not the muon, electron, and pions from the beam, which have much higher energy (and have been excluded in the zoom window of Figures 1-2 and 1-3 for clarity purposes).

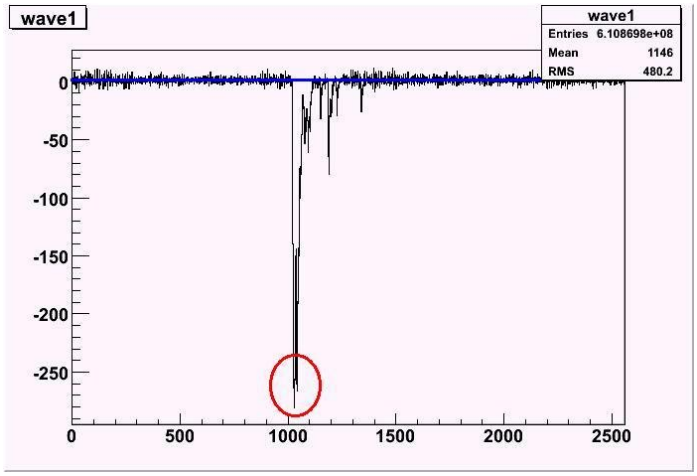

Figure 1.1 – A typical flash ADC waveform. The red circle notes the peak, whereas the blue line indicates the baseline (slightly above 0 in this case).

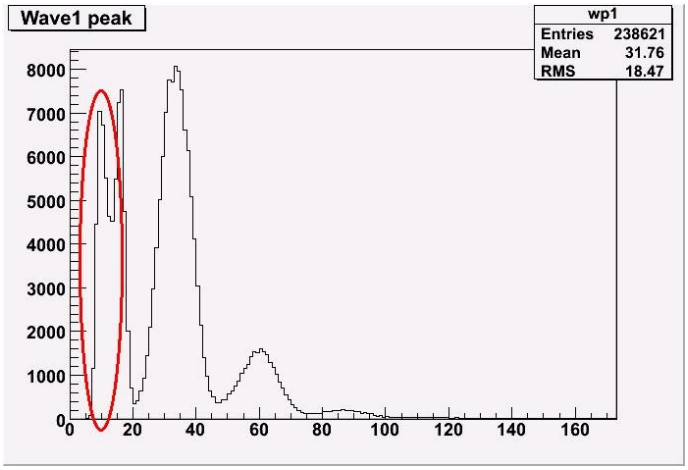

Figure 1.2 – A typical histogram of the absolute peak heights found from the Flash ADC data. The red circle highlights the

The second major modification to the original code was switching over to using the integral of the current vs. time pulse in place of the peaks on the current waveform. The major difficulty in designing the integral calculator was determining the range of integration. There were two methods by which to do this. One could specify an arbitrary range by examining a few waveforms by hand, and then use this range to integrate the peak, by first locating the

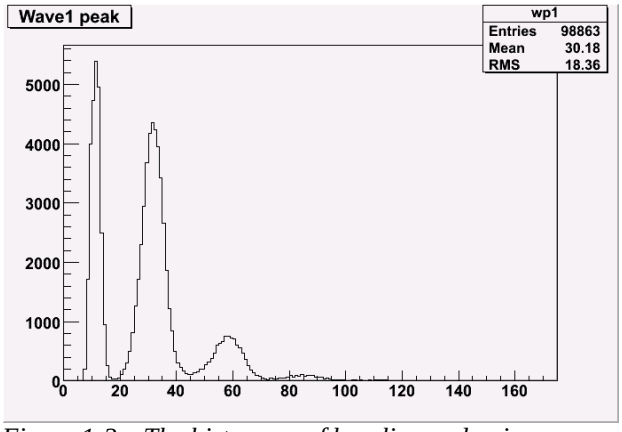

Figure 1-3 – The histogram of baseline and noise-adjusted Flash ADC peaks.

peak and then integrating forward and back of that point by the specified arbitrary value(s). The other method, which I chose to implement, is to use a dynamic range finder, which locates the end of the current waveform by following it down till it returns to the electronic noise baseline level, and then integrates to that range. For each event's waveform, one then takes the integral calculated and then adds it to a histogram. The resulting histogram is shown in Figure 1-4. Again we note distinct peaks representing, in order, one, two and three photoelectrons, excluding the leftmost peak. We can improve this resolution by relating the integrals of the peaks with their heights, and drawing a 2d histogram of Peak Height vs. Integral as demonstrated in Figure 1-5. We can improve the resolution of the peaks visible in Figure 1-4 by slicing the integral histogram into several, corresponding with the peak height groupings visible in Figure 1-5. For example, the first slice would be for integrals that result from peak heights ranging from 20 – 45 (the lowest slice represents noise and is thus disregarded). This would allow us to separate events where the integral was evaluated over a greater range due to extensive afterpulsing, which would extend the range of integration detected by the dynamic range finder. By specifying the peak ranges, we can create several slices, which should improve the resolution between the peaks visible in the integral histograms. Indeed, Figure 1-6 shows the first slice of this 2d histogram, and we can see an improved peak resolution as compared to Figure 1-4, where all the integrals are graphed together, without using a peak height cutoff. The two peaks are likely indicative of afterpulsing occurring. For example, the first peak would represent one photoelectron, and no afterpulsing, whereas the second peak would represent a case where a single photoelectron was detected, and an afterpulse occurred which was within the integration range. Using the slices such as those in Figure 1-6, we can determine the charge output from the MPPC that corresponds to one, two and three photoelectrons, which in term will allow us to determine the charge yield for muons, pions and electrons, as Dr. Kitching does in subsequent analysis.

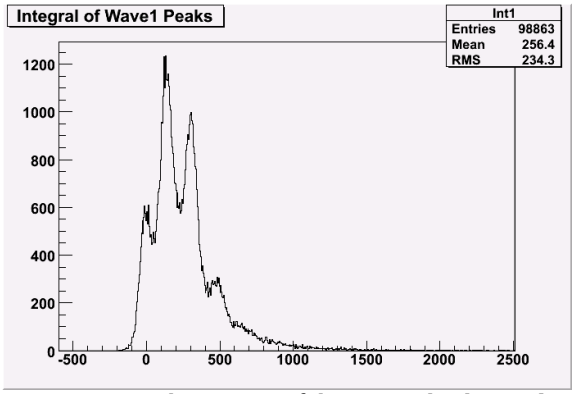

Figure 1-4 – A histogram of the integrals obtained using a dynamic range integrator.

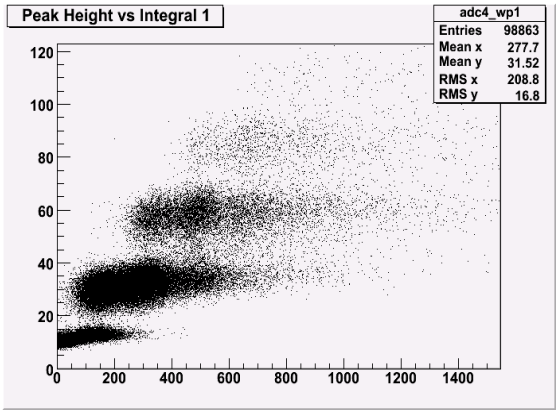

Figure 1-5 – A 2d Histogram of Peak Height vs. Integral of Peak Waveform

1.4 - Conclusion

The now improved method for determining the conversion ratio between photoelectrons and charge output allows greater accuracy than in the original version of the analyzer program, and allows examination of both current fluctuations and total current outputs of the MPPC.

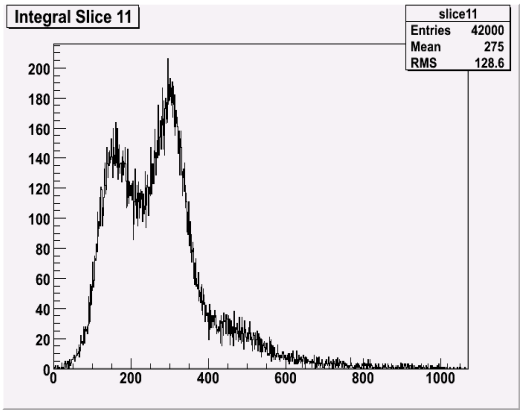

Figure 1- 6 – The first slice of the integral graph shown in Figure 1-5.

2 – Light Injection System Accuracy Simulations

2.1 – Introduction

The FGD will include a Light Injection system which consists of LEDs that shall be flashed at known light levels in order to detect the charge output of MPPCs in order to note MPPC charge drift over time, and correct for this through regular calibration. Theoretically, one could shine light at every conceivable light level and measure the charge detected from the MPPC. However, it requires considerable effort to calibrate a given LED to shine light at a given light level, and calibrating for the approximately 20000 different light levels that may be needed to saturate the MPPC would require an immense amount of time. It is much more practical and efficient instead, to use just a few light levels, parameterize the resulting function, and accept a small loss of accuracy in place for a large gain in efficiency. This is why I worked on determining the best ways to minimize the quantity of light levels that shall need to be calibrated for each LED, and also the number of flashes at each LED needed. Both of these values are strongly affected by a number of imperfections that are inherent to LEDs.

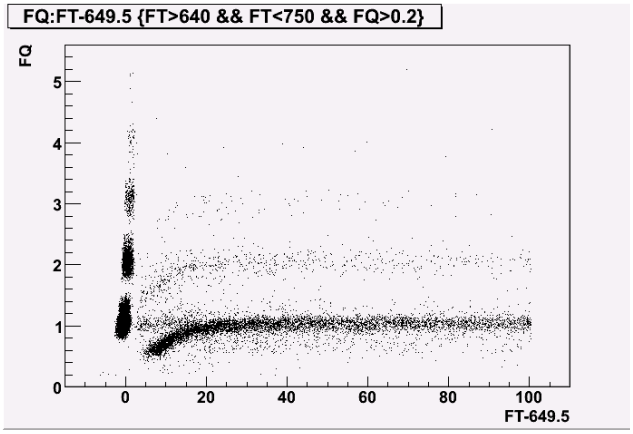

Figure 2-1 – Fabrice Retiere's graph of dark noise; Photoelectrons vs. time (ns)

The first amongst these is that the power supply to the LEDs, though expected to be accurate, is also expected to exhibit small random fluctuations about the mean in a Gaussian manner, thereby altering the light output of the LEDs. Secondly, as with all electronics, shot noise is expected; this noise shall result cause the actual output level of photons to fluctuate along a Poisson distribution. Thirdly, due the quantum efficiency nature of the MPPCs, we expect that the number of photons actually detected by the MPPC will be distributed along a Gaussian distribution. Fourthly, the photon yield of LEDs is expected to be non-linear, as a function of voltage. The purpose of the Light Injection accuracy simulations is to minimize the human effort while still accounting for these four effects.

2.2 - Methodology

Much of my tests relied heavily upon the Light Injection Simulation designed and implemented by Thomas Lindner. This simulation is capable of emulating the waveform output from the ASIC electronics for the FGD, with full functionality accounting for afterpulsing, crosstalk, quantum efficiency, recovery, and dark & electronic noise.

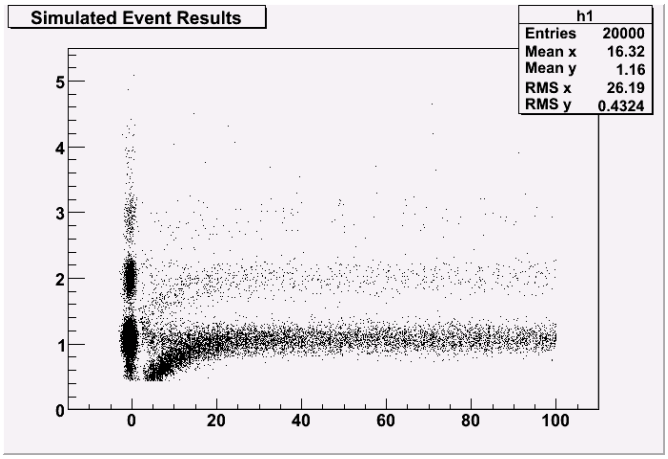

Figure 2-2 – Simulated replication of Figure 2-1

Ere I could commence my own tests, I had to ensure that the results yielded by his simulation agreed accurately to real data. As a basis for comparison, I used data collected by Fabrice Retiere. Dr. Retiere's used the MPPC to detect double dark noise pulses, occurring close to one another (less than 100ns apart), and then plotting them on a scatter plot of MPPC signal peak height vs. time. The graph he obtained from this method is shown in Figure 2-1. The distinct characteristics in this graph are that the recovery of the MPPCs after discharging is clearly visible, in addition to the relative frequencies of one, two three, and four photoelectrons. The former yields a good measure of the short recovery time of MPPC and also

afterpulsing probability, and the latter yields a good measure of crosstalk. It appears that crosstalk causes a factor of twenty difference between one, two, three and four photoelectrons. For example, there are twenty times as many events for one photoelectron as there are for two, and twenty times as many events for two photoelectrons as there are for three photoelectrons and so on.

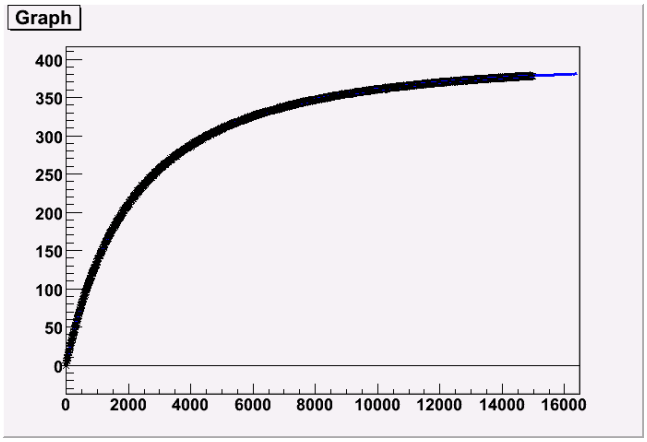

Figure 2-3 – Photoelectrons detected vs. photons incident on the MPPC, as generated by the simulation

With these characteristics and quantities in mind, I experimented with various settings for crosstalk and afterpulsing probabilities, and also with recovery time. The full set of values I used are cataloged in Appendix A, but the most relevant ones are: P(Crosstalk) = 0.032*V, P(Afterpulsing) = 0.2*V, and short pixel recovery time = 8.75ns. From using these values and Dr. Lindner's simulation, the graph shown in Figure 2-2 was obtained, with the intention of fully emulating Dr. Retiere's real data.

The resulting simulated data matched closely all the characteristics evident in Dr. Retiere's real data, including event frequency of one, two, three and four photoelectrons. The one noticeable discrepancy between the real data and the simulated data was that the simulated data appears to have a higher variance than the real data. I was unable to correct for this.

With the ability to reproduce Dr. Retiere's graph's features both qualitatively and quantitatively, I came to believe that the simulation was sufficiently calibrated to produce data that was similar to what would be produced in real life. I wrote a program that simulates Poisson fluctuations in the light output, Gaussian fluctuations in the charge output of the MPPC, non-linear photon yield in the LEDs, and Gaussian fluctuations in the power supply to the LEDs. I then conducted trial runs at different light levels to determine the number of LED flashes needed at every light level to be able to reconstruct the number of photons incident on the MPPC from the charge detected to within 5% of the actual value.

With these trial runs, I generated data sets that used the parameters specified in Appendix A, in addition to the four fluctuations mentioned above,

2.3 - Results

My results indicate that approximately 25 different light levels shall be needed for this, with the specific photon values listed in Appendix B. At each light level, the LED should be flashed approximately 250 times. A curve of the form $y = A(1-e^{-x/B}) + C(1-e^{-x/D}) + E(1-e^{-x/F})$ fitted to the data points should yield a fit that allows photon reconstruction to within five percent of photon reconstruction possible if all light levels were sampled. Figure 2-3 depicts what a fit of this form would look like when sampled over the entire range of light levels. The optimized fit for this full range sampling is given in Appendix C. In simulating Figure 2-3, two methods were available. One could either use the peaks of the waveforms generated by Dr. Lindner's simulation or use the integral of the waveforms. I compared both methods, and although the latter is technically the correct method, I chose the former as both were correlated strongly in addition to the fact that I was concerned for processing time. In section three, I return to the integration method in order to replicate results found by the British photosensor group.

Gaussian fluctuations in both the power supply to the LEDs and in the charge output of the MPPC, when separated out, do not contribute significantly to the inaccuracies generated by using only a few light levels for sampling MPPC charge output, especially when on the order of 200 flashes are used. It appears that light level fluctuations due to power supply instability can have a standard deviation up to 30% of the nominal light level. Dr. Paul Poffenberger, at the University of Victoria, conducted tests of the LED's light output stability; with a fairly unstable power supply as compared to the ones expected to be used in

the final T2K FGD, he recorded a 7-14% standard deviation in the light output caused largely by, he believes, fluctuations in the power supply. Nonetheless, this 7-14% variation is well within the 30% upper bound simulations appear to indicate. In regards to the Gaussian fluctuation of the charge output of the MPPC, we note the simulation actually yields a higher variation than was observed in the real data through comparing Figures 2-1 and 2-2.

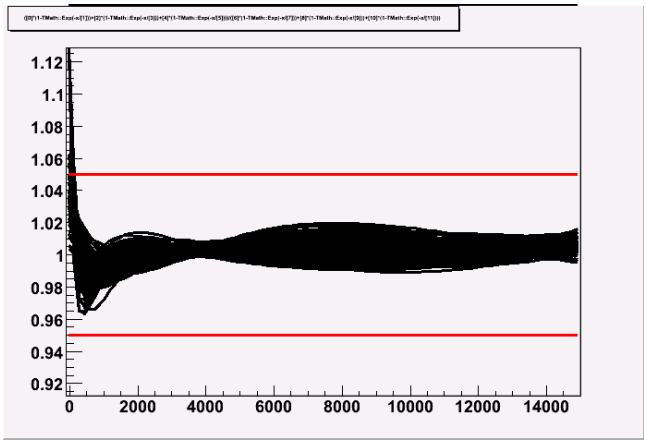

Figure 2-4 – Measure of errors produced when accounting for the above-mentioned fluctuations

The most significant contributors to error in photon reconstruction are Poisson fluctuations in the LED light output, and the non-linear response of the LED's light output to a change in voltage. It should be noted that these effects affect the error in reconstructing the number of incident photons in different fashions. The shot noise, which causes the light emitted by the LEDs to possess a Poisson distribution, largely affects the ability to accurately reconstruct the quantity of incident photons at the lower light levels, generally less than 2000. Unfortunately, simulations appear to indicate that shot noise shall make reconstruction of light levels under 200 photons, or about 35 photoelectrons, very difficult, as the error in reconstruction is expected to exceed the 5% threshold. This poses a problem as M11 beam tests conducted by Dr. Kitching indicate that a minimum ionizing particle is expected to produce in the 30-35 photoelectron range also. Increasing the number of light levels at which to sample the MPPC output does not seem to eliminate this problem. Experiments with increasing the number of LED pulses at each light level also did not yield significant improvement in the error in this range.

The non-linearity of the photon output with respect to the voltage supply is not a large concern in the lower light levels, where the light output is still approximately linear. Non-linearity plays a significant role, however, in the upper light ranges where non-linearity becomes an issue (one is presuming here that light output is of the form $Photons = AV(1+\alpha V)$, where A is some arbitrary constant and α is the nonlinearity factor.

After determining the individual contributions of each of these effects, I simulated the low statistics data (with ~25 light levels, and 200 pulses at each level), and used the ROOT package to fit the above-mentioned triple exponential to the data. I then took the ratio of the function fitted to this low statistics set to function fitted to the high statistics set sampled over the entire light range. I repeated this process for one hundred different generated sets of low statistics data and produced a graph of the evolution of the ratios. This gives a good measure of the expected amount of error at different light levels. The resulting graph is shown in Figure 2-4, where the red lines represent the 5% error tolerance. As visible in Figure 2-4, we are well within our expected error tolerances, except in the very low light level range.

3 – Analysis of the Importance of Accurately Measuring various MPPC parameters

3.1 – Introduction

This last portion of my time at TRIUMF was spent examining the effect of measurement errors in several MPPC parameters: decentralization of the wavelength-shifting fiber relative to the MPPC; incorrect measurement of non-uniformity of light distribution; measurement of the afterpulsing parameter; measurement of the crosstalk parameter; and measurement of the short recovery constant. In addition, I also examined whether the simulation corresponds qualitatively with results from the British photosensor group. By assuming that the nominal values in Appendix A were the correct and/or assumed values of the MPPC during calibration, and then using the simulation to vary these parameters, fitting the generated data, and then taking the ratio of the fit to the incorrect parameter data to the fit of the correct parameter data, I was able to estimate the amount of error expected if various MPPC parameters are mismeasured. The full results of this section are visible in Appendix D, with the variations I tested, and also the maximum resulting error.

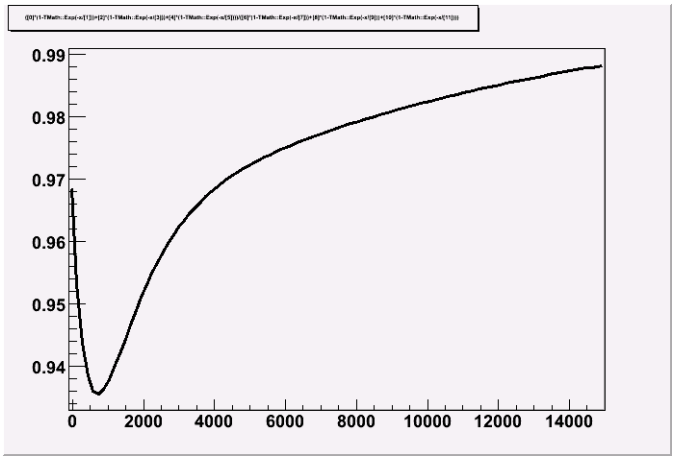

Figure 3-1 – A ratio-error graph of what occurs if the fiber is accidentally 0.2mm off-center of the MPPC.

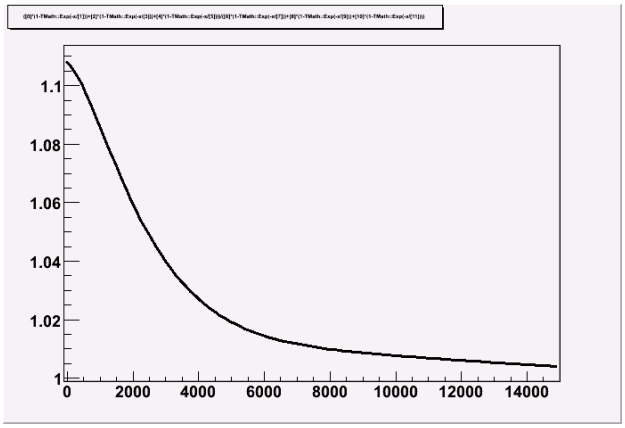

Figure 3-2 – Error ratio graph if crosstalk probability is 200% of the nominal value.

3.2 - Wavelength Shifting Fiber is not centered to MPPC Properly

The first parameter I examined was the positing of the wavelength shifting fiber relative to the MPPC. I varied the position of the fiber, so that it was off-center by various amounts. In the worst case scenario, when the fiber is 0.2mm off-center (a significant amount since the MPPC size is either 1.0mm or 1.3mm), I obtained the error graph presented in Figure 3-1. It is fortunate to note, that even in this worst-case scenario, the maximum error expected is on the order of 5% around 1000 photons, which, although capable of corrupting photon reconstruction by about 20 photons, is not particularly egregious. I also tested other amounts of misalignment, whose results are indexed in Appendix D. Each of the above-mentioned parameters were varied independently of one another; there was no cumulative modification of one property after another. Rather, each value was reset to its nominal or 'correct' value before the next test was conducted, so as to be able to isolate the effect and importance of each parameter.

3.3 - Modifying the Probability of Crosstalk

The LIS simulation uses a linear model of the probability of crosstalk occurring, which is a linear function of overvoltage (voltage over the breakdown voltage). The nominal value for this was 0.3/V. Rather than modify the overvoltage to affect the probability of crosstalk occurring, which would affect a series of other voltage-dependent effects, I modified the crosstalk probability to various values to examine the importance of correctly measuring the crosstalk parameter. Figure 3-2 shows the error-ratio graph if the actual crosstalk probability parameter of an MPPC is 200% of what it is believed and fitted to be. We can note here that the

resulting maximum error is about 12%, which is more than double the error from a worst-case scenario misalignment of the fiber. Again, the full results for other variations are listed in Appendix D.

3.4 - Modifying the Probability of Afterpulsing

Like crosstalk, the present simulation uses a linear probability of afterpulsing, as a function of overvoltage. Similar to my method for testing the importance of correct parameterization of the crosstalk probability, I similarly examined the effects of afterpulsing probability, by varying the slope of the linear probability. Figure 3-3 shows the resulting error-ratio graph if the afterpulsing probability is 200% of the expected probability. We note here that the maximum error here is approximately 17%, which is somewhat larger than the error that resulted from doubling the crosstalk value. Appendix D contains the full results of my testing for other values.

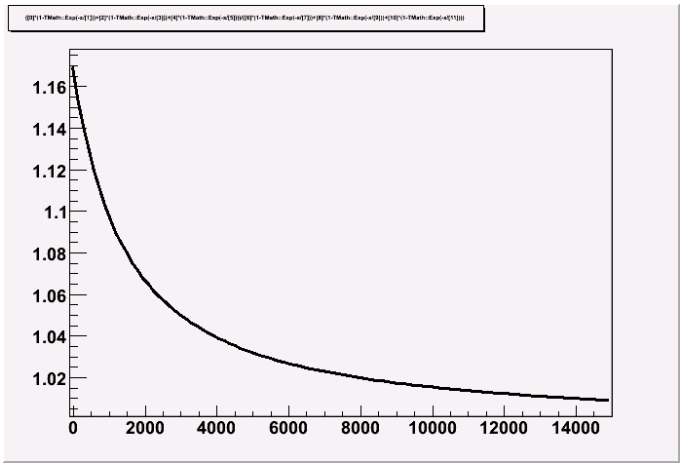

Figure 3-3 – Error ratio graph if afterpulsing probability is 200% of the nominal value.

3.5 - Modifying the Short Recovery Time

It is believed, as detailed in Dr. Oser and Dr. Lindner's article on the MPPC, that the MPPC has two recovery times: one which is through MPPC pixels redistributing charge amongst themselves after some pixels fire; the second which occurs from the external circuit pumping charge to the MPPC as a whole. The former is the "short recovery time", believed to have a nominal recovery constant of 8.75ns, and the latter is the "Long recovery time" which has a recovery constant on the order of several microseconds. For the purposes of this test, I examine the short recovery time constant, since the long recovery constant is more readily adjustable and controlled through modifications to the external circuit, whereas the short recovery constant is a property inherent to the MPPC device, which needs to empirically measured and can not be easily modified. The MPPC recovery is believed to be exponential in nature, explained by the following formula $Ratio\ of\ Recovery = 1 - e^{-t/\tau}$, where τ is the recovery time constant. Figure 3-4 shows the resulting error ratio graph for if the short recovery time constant is only 50% of its expected value. We can note here that the maximum error is approximately 8%.

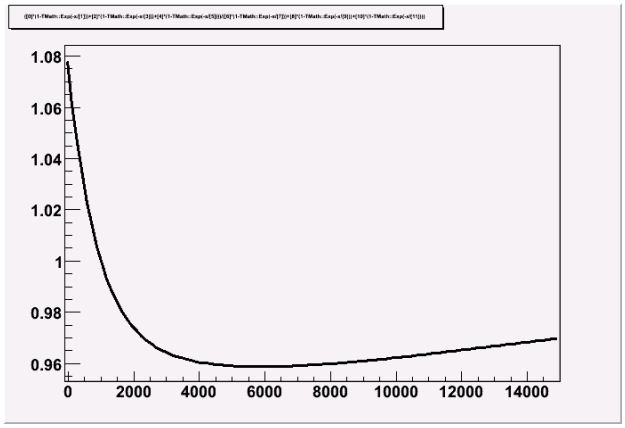

Figure 3-4 – Error ratio graph if the short recovery constant is only 50% of the nominal value.

3.6 - Using Integration Method in place of Peak-finding

Results from the British photosensor group (A. Carver et al) show saturation curves that indicate that more than the maximum number of pixels on the MPPC are firing. As this was not exhibited in the peak finding method, as a safety check, I switched to the integration method to check whether the MPPC simulation also causes this effect to be observed. Figure 3-5 shows the saturation curve for the 1.0mm MPPC, which has 400 pixels. The saturation level found by this curve is approximately 426 pixels, which agrees qualitatively with the British group's results: oversaturation occurs. I further tested their hypothesis that this oversaturation is due to afterpulsing by disabling afterpulsing in my simulation. The resulting curve saturated at less than 400 pixels (approximately 398). This lower value is what is expected since when a high number of pixels are discharged, the entire MPPC is effectively depleted of charge and needs to

recover from the external circuit, which takes several microseconds. These latter results corroborate the British group's hypothesis that this oversaturation is due to afterpulsing.

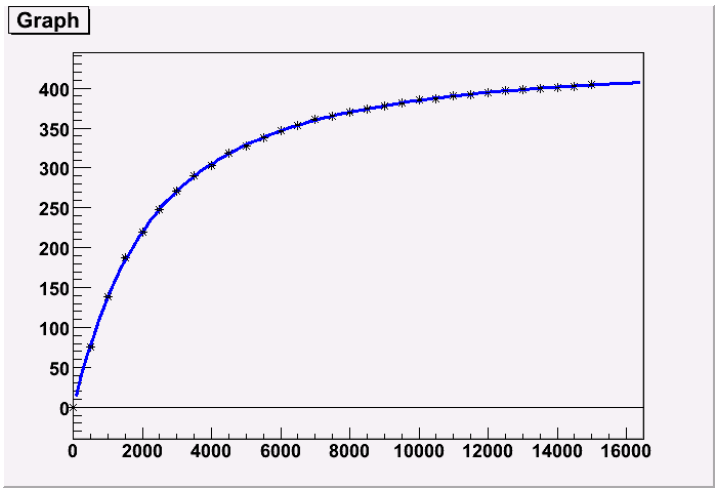

Figure 3-5 – Saturation curve using an integration gate method, as opposed to the peak finding method

3.7 - Conclusions

The above simulations apparently indicate the relative importance of measuring and quantifying the various parameters related to the MPPC. The most important parameter that needs to be characterized and understood is apparently afterpulsing, which with can yield very large amounts of error for even a 50% variation, approximately 11%. We note that afterpulsing also significantly affects the shape of the saturation curve, as well as the saturation value.

The next most important parameter that needs to be understood is crosstalk, which also yields very high levels of error, in addition to significantly altering the shape of the saturation curve. Understanding the MPPC short recovery time is also approximately as important as quantifying the crosstalk parameter.

A particularly significant consequence of my tests in section 3-6 indicate that it might indeed be more valuable to use peak height as a measure of pixel activation than to integrate the current over some range to obtain the charge. The simulation yielded very similar results in terms of photon reconstruction regardless of whether I used the current peak height or the total charge over some integration gate, with a single caveat that the light arrives at the MPPC fairly close together (within a 5-15ns window). If one uses the peak height method, one will not see any oversaturation occurring. It should, in principle, be possible to take data at low light levels, and measure both the peak height and the charge, and obtain the number of photoelectrons corresponding to the charge. This will yield a conversion factor for the peak height, which will be linear in relation to photoelectrons over the entire range, whereas the integration method will yield a nonlinear relation between charge and photoelectrons detected (resulting in oversaturation). As such, I believe it would be simpler to use the above suggested method for the production of saturation curves in order to minimize the consequences of afterpulsing.

Appendix A – Light Injection Simulation Parameters

These are the parameters used in the Light Injection Simulation, calibrated in order to create simulated data that closely resembles dark noise data presented by Fabrice Retiere.

1. Scintillator parameters:

< elecSim.Scintillator.SwitchBirks = 1 > Turn on/off birk's attenuation
< elecSim.Scintillator.BirksConstant = 0.005 cm/MeV > Birk's attenuation constant

< elecSim.Scintillator.SwitchPhot = 1 > Turn on/off MeV-Photon conversion
< elecSim.Scintillator.PhotPerMeV = 9.3 1/MeV > Photons per MeV (default)
< elecSim.Scintillator.PhotPerMeV.fgd = 9.3 1/MeV > (FGD value)

2. WLS fiber parameters

< elecSim.WLS.SpeedOfLight = 20.0 cm/ns > Speed of light in WLS fibre

< elecSim.WLS.SwitchMirr = 0 > Turn on/off far-end of fibre mirroring (default)
< elecSim.WLS.SwitchMirr.fgd = 1 > (FGD value)
< elecSim.WLS.ReflectionProb = 0.99 > Reflection Probability

< elecSim.WLS.SwitchDecay = 0 > Turn on/off fibre decay constant
< elecSim.WLS.SwitchDecay.fgd = 1 > (FGD value)
< elecSim.WLS.DecayTime = 11 ns > Fibre decay constant (default)

< elecSim.WLS.SwitchAtt = 1 > Turn on/off fibre attenuation length

3. PPD parameters:

< elecSim.PPD.GateDuration = 5800 ns >
< elecSim.PPD.PDE = 0.30 >
< elecSim.PPD.NumberPixels = 400 >
< elecSim.PPD.PulseDuration = 40 ns >
< elecSim.PPD.RecoveryTime = 30 ns >
< elecSim.PPD.PePeakWidth = 0.00 >
< elecSim.PPD.PePedestalWidth = 0.10 >
< elecSim.PPD.DCR = 700000. >
< elecSim.PPD.Crosstalk = 0.0 >
< elecSim.PPD.CT1 = 0.1 >
< elecSim.PPD.CT2 = 0.01 >
< elecSim.PPD.CT3 = 0. >

< elecSim.PPD2D.NumberPixels = 400 >
< elecSim.PPD2D.ActiveFraction = 0.9 >
< elecSim.PPD2D.PixelRecoveryTime = 1000 ns >
< elecSim.PPD2D.PixelCapacitance = 87.5E-15 >
< elecSim.PPD2D.PulseDuration = 50 ns >
< elecSim.PPD2D.PDE = 0.9 >
< elecSim.PPD2D.Temperature = 25.0 >
< elecSim.PPD2D.BreakdownVoltage = 68.5 >
< elecSim.PPD2D.VoltPerDegree = 0.055 >
< elecSim.PPD2D.OperationVoltage = 69.5 >
< elecSim.PPD2D.PePeakWidth = 0.02 >
< elecSim.PPD2D.VPDE0 = 68.5 >

< elecSim.PPD2D.PDESlope = 0.3 >
< elecSim.PPD2D.VDCR0 = 68.5 >
< elecSim.PPD2D.DCRSlope = 125000 >
< elecSim.PPD2D.VPCT0 = 68.5 >
< elecSim.PPD2D.PCTSlope = 0.032 >
< elecSim.PPD2D.VAP0 = 68.5 >
< elecSim.PPD2D.APSlope = 0.2 >

< elecSim.PPD2D.SwitchNonuniformLight = 1 > Turn on/off nonuniform pixel illumination
< elecSim.PPD2D.DistributionSigma = 0.47 mm > Sigma for gaussian pixel illumination
< elecSim.PPD2D.SensorSize = 1.0 mm > Length of one side of sensor active area.
< elecSim.PPD2D.RecoveryTimeShort = 8.75 ns > Short Recovery timescale
< elecSim.PPD2D.RecoveryTimeLong = 5000.0 ns > Long Recovery timescale

< elecSim.PPD2D.GateDuration.fgd = 2500 ns >
< elecSim.PPD2D.DCRSlope.fgd = 2000000 Hz >

< elecSim.PPD2D.APtauShort = 10 >
< elecSim.PPD2D.APtauLong = 40 >

8. Electronics parameters for the FGD:

< elecSim.After.SwitchCreateWaveforms = 1 > Switch: create After waveforms?
< elecSim.After.NTimeBins = 1500 > Number of time bins
< elecSim.After.MaxDigitalCounts = 4096 > ADC range
< elecSim.After.SamplingTime.fgd = 2.0 ns > Sampling time
< elecSim.After.PulseRange.fgd = 5000.0 ns > Range over which to evaluate the pulse shape.
< elecSim.After.SwitchUseCompression = 0 > Switch: use compression?
< elecSim.After.CompThreshold = 55 > Threshold above which compression algorithm saves samples.
< elecSim.After.SwitchFgdPulseShape = 1 > Switch: use the FGD shape for a single Q/T pulse?

The following are the parameters for the FGD pulse shape
< elecSim.After.FgdPulseShape.Normalization = 1.0 >
< elecSim.After.FgdPulseShape.ShapePar = 100.84 ns >
< elecSim.After.FgdPulseShape.WidthPar = 100.0 ns >

Appendix B – Suggested Operating Photon Levels

For optimal efficiency, it is suggested that the following light levels be used:

Photon output = 0 and approximate detected PEs = 0
Photon output = 30 and approximate detected PEs = 7
Photon output = 60 and approximate detected PEs = 12
Photon output = 90 and approximate detected PEs = 17
Photon output = 120 and approximate detected PEs = 22
Photon output = 150 and approximate detected PEs = 27
Photon output = 180 and approximate detected PEs = 32
Photon output = 210 and approximate detected PEs = 39
Photon output = 240 and approximate detected PEs = 41
Photon output = 270 and approximate detected PEs = 46
Photon output = 300 and approximate detected PEs = 52

Photon output = 330 and approximate detected PEs = 55
Photon output = 830 and approximate detected PEs = 138
Photon output = 1330 and approximate detected PEs = 165
Photon output = 1830 and approximate detected PEs = 198
Photon output = 2330 and approximate detected PEs = 226
Photon output = 2830 and approximate detected PEs = 243
Photon output = 3330 and approximate detected PEs = 260
Photon output = 3830 and approximate detected PEs = 284
Photon output = 4330 and approximate detected PEs = 242
Photon output = 4830 and approximate detected PEs = 244
Photon output = 5330 and approximate detected PEs = 323
Photon output = 8330 and approximate detected PEs = 313
Photon output = 11330 and approximate detected PEs = 368
Photon output = 14330 and approximate detected PEs = 377

Appendix C – Full Light Range Simulation Fit

The approximate expected fit to when a full range light sampling is done using the Light Injection System Simulation is $y=191(1-e^{-x/1740})+31(1-e^{-x/536})+168(1-e^{-x/5700})$, yielding an approximate saturation point of 390 pixels at 15000 photons incident on the MPPC.

Appendix D – Varying MPPC Parameters

Parameter Varied	*Amount of Variation(compared to nominal value)*	*Maximum Resulting Error*
Location of the fiber relative to the MPPC	0.2mm in the x or y direction	5.0%
	0.1mm in the x or y direction	3.0%
	0.2mm in both x and y directions	9.5%
Crosstalk Parameter	0.0%	16.0%
	50.0%	9.0%
	200.0%	12.0%
	300.0%	42.0%
Afterpulsing Parameter	0.0%	16.0%
	50.0%	10.0%
	150.0%	11.0%
	200.0%	17.0%
	300.0%	75.0%
Short Recovery Constant	50.0%	7.5%
	150.0%	2.9%

Acknowledgements

The following articles were used as references for the purposes of the above report:

Oser, S. and Lindner, T. *Simulation of MPPC Photosensor Behavior*. 17 Aug, 2007.

Carver et al. *MPPC measurements and gate study with TFB*. 19 Sept. 2007.

"Shot Noise," *Wikipedia*. http://en.wikipedia.org/wiki/Shot_noise

The following individuals were used as references, sources, and mentors throughout the period during which this work was conducted:

Dr. Scott Oser, UBC Physics Department and TRIUMF, T2K Experiment
Dr. Peter Kitching, TRIUMF, T2K Experiment
Dr. Thomas Lindner, TRIUMF, T2K Experiment
Dr. Fabrice Retiere, TRIUMF, T2K Experiment
Dr. Paul Poffenberger, University of Victoria Physics Department, T2K Experiment
Dr. Rich Helmer, TRIUMF, T2K Experiment