

國立東華大學物理研究所
碩士論文

Department of Physics
National Dong-Hwa University
Master's Thesis

指導教授：林子強博士
章文箴博士
Advisor : Dr. Lin Chi-Yong
Dr. Chang Wen-Chen

Measurement of Sonoluminescence by a
Cross-Platform DAQ System



研究生：梁浩然
Graduate : Leong Hou-In

中華民國九十四年六月

致謝

這裡誠心感謝我的指導教授林子強老師及中央研究院物理研究所章文箴老師，讓我有幸在台灣中央研究院物理研究所進行「聲致發光」實驗的研究工作，並在他們的辛勤指導下順利把論文完成，在此致上由衷的感謝。

其次，特別感謝中研院物理所鄧炳坤老師，在實驗技巧上的指導。楓凱學長、大龍學長在實驗上的幫忙，燦隆學長在電腦程式上的指教。

也同時感謝東華大學物理所陳企寧老師在研究兩年中給予的支持與關懷，世傑學長、泰宏同學、碩濱同學、彥宏同學及彥廷同學，在這趟台北-花蓮學習旅途上給予交通及生活上的幫忙。

最後，感謝我的家人和怡頻，他們給了我最大的支持和鼓勵，以致完成學業，在此，致上謝意。

Contents

中文摘要

Abstract

Chapter 1 Introduction.....1

1-1 Brief History of SBSL	1
1-2 Evidence for Nuclear Reactions in imploding Bubbles.....	2
1-3 Momentum correlation in Sonoluminescence.....	2
1-4 Angular correlation.....	5

Chapter 2 The Data Acquisition system.....7

2-1 About DAQ.....	7
2.1.1 Trigger.....	7
2.1.2 Hardware Implementation (Front-End-Electronics).....	9
2.1.3 Software Implementation (Read-Out System).....	10
2-2 Experimental apparatus.....	12
2.2.1 Introduction of devices in SBSL experiment.....	12
2.2.2 Preparation of water.....	13
2-2.3 Instrument Operation.....	14
2-2.4 Identify the Sonoluminescence.....	15
2-3 Data Acquisition in SBSL experiment.....	16
2-4 ADC readout system.....	17
2-5 VME-bus.....	19
2-6 Buffer management.....	19

Chapter 3 UNIDAQ:Software for Unix-Based Data Acquisition.....22

3-1 Data and control flow.....	23
3-2 The Recorder Process.....	24
3-3 The Collector Process.....	25
3-3.1 Modification of the Collector Process.....	25
3-4 The Analyzer Process.....	26
3-5 The Message Template.....	27

3-6 The Event Buffers.....	28
3-6.1 Number and Size of Event Buffer.....	30
3-7 Structure of memory and control programs.....	30
3-8 Installation of UNIDAQ on RedHat 9.0 platform.....	31
Chapter 4 DAQ Test.....	32
4-1 Gamma ray source experiment.....	32
4-2 LED Test Experiment.....	35
Chapter 5 Results and Discussion.....	37
5-1 Background Measurement.....	38
5-2 Sonoluminescence Measurement in 1-D histograms.....	40
5-3 ADC value versus Event Number.....	44
5-4 The comparison of the correlation between 2 PMTs.....	46
5-5 Angular correlations in SL-light.....	48
Chapter 6 Summary and Outlook.....	50
6-1 Summary.....	50
6-2 Outlook.....	51
Reference	52
Appendix A.....	54
Experimental Verification of Rutherford's Nuclear Theory of the Atom.....	54
Appendix B.....	55
B-1 Installation of UNIDAQ on Red Hat Linux 9.0 (kernel-2.3.20-8).....	55
B-2 SL-DAQ operation guide.....	57
Appendix C Sub-routine : user_event.c.....	58

中文摘要

單一氣泡的「聲致發光」現象是一種經由聲波激發液體中的氣泡以釋放出光子的物理現象，本研究是利用 UNIX 跨平台環境下執行 UNIDAQ 資料擷取系統並以 26kHz 之震盪頻率對「聲致發光」所釋出之光子進行量測。

Abstract

Phenomenon of Single Bubble Sonoluminescence (SBSL) is the emission of light by a bubble in liquid excited by sound. In our study, we perform the SBSL experiment and measure the SBSL emission light by ADC system with 26 kHz oscillating rate under the cross UNIX-based platform with UNIDAQ data-acquisition system.

1. Introduction

1-1 Brief History of SBSL

In 1894, the early tests for British destroyer discover severe vibration from propeller, suggested source : large bubbles.

In 1896, Henri Becquerel discovered that uranium salt could darken some regions on the photographic.

In 1917, chemical and biology effects of supersonic were report by Richard and Loomis, sonochemistry and sonobiology.

In 1934, H.Frenzel and H.Schultes exposed a negative plate near a acoustic field generated in the water also discovered some regions on the plate were darken. It's a crucial milestone to sonoluminescence, namely MBSL, "multiple bubble sonoluminescence " When tensile force in the water exceeds the threshold for cavitation, many bubbles were created. If the cavitation was intense enough, sonoluminescence occurred. But this phenomenon was hard to study at that time because of the randomness of position, size and number of the bubbles.

Single bubble sonoluminescence (SBSL), an air bubble in water being trapped in the velocity node of an acoustical pressure field, oscillates periodically in liquid and collapses so forcefully that energy focus inside the bubble leads light emission. The SBSL phenomenon was early discovered in 1989 by Felipe Gaitan, a graduate student at the Univ. of Mississippi working with Larry Grum, who had seen hints of light emission from a single bubble in 1985. In Gaitain's thesis, he aimed to build up a systematic setup of experiment on the oscillation and collapse of bubble. Stable position of the bubble is an important breakpoint to separate SBSL and MBSL. It's

easier for researcher to study this interesting phenomenon. After the great work of Gaitan, this field get explosive progress[1].

1-2 Evidence for Nuclear Reactions in imploding Bubbles

In 2002 , the Science magazine reported that Taleyarkhan et al. have achieved the conditions for $d + d$ nuclear fusion with acoustic shock waves generated in a sonoluminescence device [2 ,3, 4]. Besides photons, Taleyarkhan et al. report production of tritium and the detection of high-energy neutrons correlated in time with the sonoluminescence shock wave while the bubble collapsed. Some fraction of neutrons is detected in special neutron detectors. This triggered the speculation of possible nuclear reaction in SL.

1-3 Momentum correlations in Sonoluminescence

The size of reaction region in sonoluminescence can be studied by two-photon experiment with the Hanbury Brown-Twiss (HBT) effect (Fig1.1). HBT has its origins in astrophysics. It is the technique of the intensity interferometry and was developed by Hanbury-Brown and Twiss in the 1950s, but has seen significant theoretical development and widespread application in subatomic physics[5], especially in determining the dimension of distant astronomical objects.

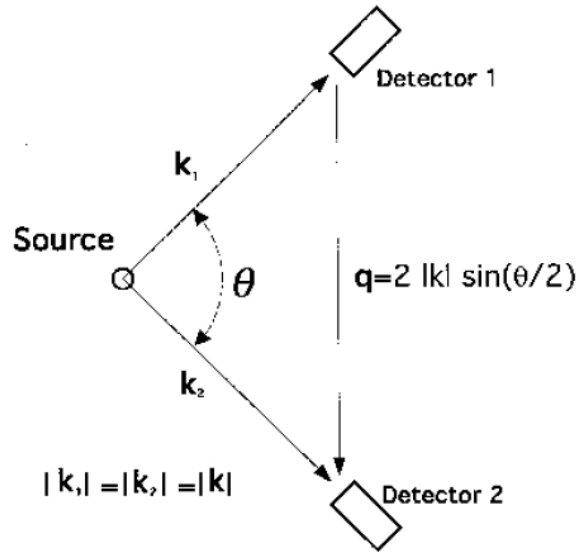


Fig.1.1. Schematic diagram of an intensity interferometer.

The two-photon correlation function $C(k_1, k_2)$ can be defined as

$$C(k_1, k_2) = \frac{P(k_1, k_2)}{P(k_1)P(k_2)} \quad (1.1)$$

where $P(k_1, k_2)$ is the probability density of detecting photons of 4-momenta k_1 and k_2 in the same pulse and $P(k_i)$ is the single photon probability density. Because the total wavefunction for identical bosons is symmetric, the maximum value of this function will occur at $k_1 = k_2$. When k_1 is not equal to k_2 , the strength of the correlation will depend on the degree of overlap between the single particle wavefunctions of the two particles. This “exchange density” is determined by the size of the emitting source. Assuming the photons to be emitted from a fireball with a stationary Gaussian distribution of radius R and lifetime τ [6] :

$$\rho(r, t) \propto \exp[-(r/R)^2 - (t/\tau)^2] \quad (1.2)$$

giving,

$$C(k_1, k_2) = 1 + \frac{1}{4} [1 + \cos^2(\theta)] \exp\left[-\frac{q^2 R^2}{2} - \frac{\Delta E^2 \tau^2}{2}\right] \quad (1.3)$$

where $q=|k_1-k_2|$ is the strength of relative momentum, $\Delta E=|E_1-E_2|$ is the energy difference of the photons and θ is the angle between them. From Eq.1.3, the width of the correlation function is inversely related to the source size R and the lifetime τ . If Assuming $k_1=k_2$ and fixing $\Delta E=0$ by applying a wavelength filter, the two photons are now emitted from the same source at the same duration with the same energy. The distance between the two detectors can now be determined as $q = 2|k|\sin(\theta/2)$.

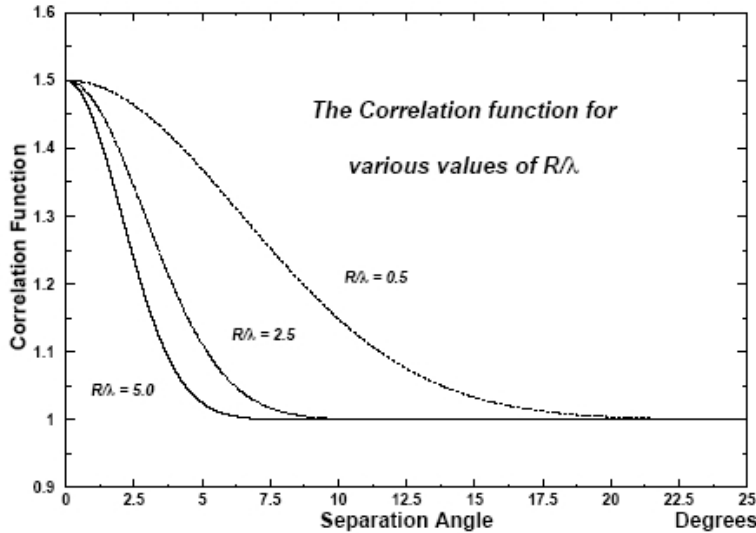


Fig1.2 The Correlation function and its dependence on the source radius and the measuring wavelength.

By substituting the terms $q = 2|k|\sin(\theta/2)$ and $|k| = h / \lambda$ into Eq.1.3., we can obtain the correlation function C as a function of the extended angle θ with a given wavelength selection of λ . If the λ is fixed (by adding a filter), we can fit the experimental data by using the curve plotted in Fig.1.2 and determine the radius as,

$$R_{bubble} = \lambda' (R / \lambda) \quad (1.4)$$

Once the size of the bubble is known, it would be possible to measure

the time duration of the flash. If the phototube positions are fixed within a region of large spatial correlation then the intensity correlation as a function of energy difference would give the lifetime. Again, with commercially available filters, the photon energy difference could be stepped through a range of values could probe flash lifetimes from 1fs (10-15 sec) to 1ps (10-12 sec).

1-4 Angular correlation

In this thesis, we present a study of the angular correlation about the pulse shape of the SL emission light by HBT method in IPAS (Institute of Physics, Academia Sinica, Taiwan). Among picosecond light sources of sonoluminescence is unusual in that the flash intensities are not considerably uniform over spherical shell. Angular dependence in the intensity of SL would be characterized by a nonzero value of the correlation,

$$\Delta Q_{AB} = \frac{1}{\overline{Q_A} \overline{Q_B}} \langle (Q_A(i) - \overline{Q_A})(Q_B(i) - \overline{Q_B}) \rangle_i \quad (1.5)$$

as a function of the angle θ_{AB} formed by the detectors A and B. The bubble is reckoned to sit at the vertex of them. In Eq.1.5, $Q_A(i)$ is the total charge recorded in detector A on the i th flash, $\overline{Q_A}$ is the running average of $Q_A(i)$, and $\langle \rangle_i$ denotes an average over i . It is found that there is a dipole component in the detector photon field [7],

$$N_A(i) = \overline{\Omega_A} (N_I + 3N_D \cos^2 \theta_{iA}) \quad (1.6)$$

In Eq.1.6, $\overline{\Omega_A}$ is the fraction of solid angle subtended by detector A, and N_I, N_D are the numbers of photons emitting isotropically, and as a dipole moment. The angular distribution of photons to strike the detector is

$N_A(i)$.

The work of this thesis is to characterize the pulse shape of the light source for which we can learn about the light emitting mechanism. Analog Digital Converter (ADC) system with 26kHz trigger rate is proposed as the measurement tool. On the other hand, an UNIX-Based data acquisition UNIDAQ which is used as the bridge between electronics and analyzer tools is adapted for our measurement.

In chapter 2 we illustrate the data acquisition system (DAQ) and the experimental setup. In chapter 3 we will introduce the UNIDAQ software. This chapter is provided the detailed design and implementation of 8 parts: 1. data and control flows, 2. the recorder process, 3. the collector process, 4. the analyzer process, 5. the message templates, 6. the event buffers, 7. structure of memory and control programs, 8. installation of the UNIDAQ on RedHat9.0. In Chapter 4, we will illustrate the developments of the analysis software and the software test under Gamma-Ray experiment and LED Test experiment. In chapter 5, we will discuss the data from the analyze process. In chapter 6, we will summarize the main conclusions obtained with this thesis work.

2 The Data Acquisition system (DAQ)

2-1 About DAQ

In this chapter, we will give an overall introduction of the data acquisition system (DAQ). In general, data acquisition system (DAQ) is the process of gathering information in an automated fashion from analog and digital measurement sources such as photomultiplier (PMT) or Charge Coupled Device (CCD) (see also 2.2 for further details). Data acquisition uses a combination of PC-based measurement hardware and software to provide a flexible, user-defined measurement system. In High Energy experiment, the DAQ system would be composed of 3 parts which are trigger, hardware and software implementation.

2-1.1 Trigger

Trigger is an electronic signal indicating the occurrence of a desired temporal and spatial correlation in the detector signals. The trigger system is aimed to select all good event candidates and reject most of the background events. To do this, the trigger must impose certain criteria which characterize the reaction, e.g., coincidence among two or more detectors, number of outgoing particles, etc. The criteria depend on the detector set-up. While the events satisfy these criteria, the other operations following, e.g., recording instruments, etc are activated.

Example of trigger for One-Body Scattering

A simple example is the elastic scattering of alpha particles from gold, the classic *Rutherford experiment* (Fig.2.1), (see also in Appendix A).

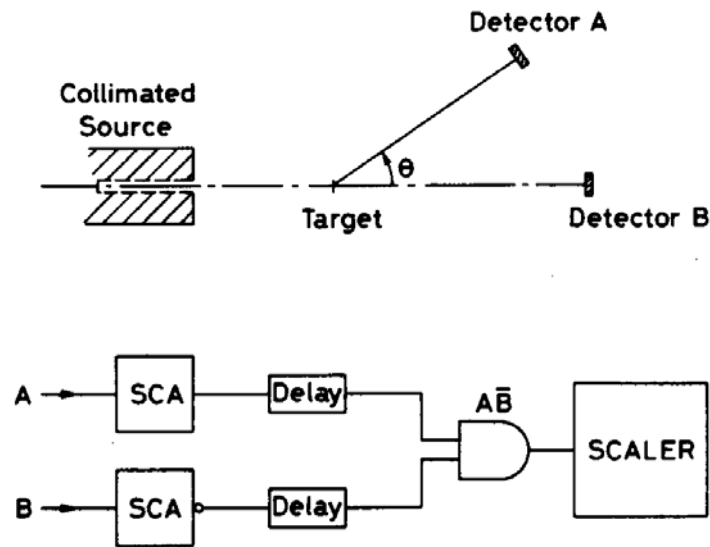


Fig.2.1 Simple set-up and trigger for Rutherford scattering

Fig.2.1 shows a beam of alpha particles is incident upon a gold foil (of the appropriate thickness) and the scattered particle detected by a single detector located at an angle θ behind the foil. Since gold nuclei are much more massive compared to the alpha particle, the energy of the scattered particle should not be too different from its incident energy. This provides one criteria for elastic scattering. Using an SCA, we can set values for the upper and lower discriminators to select out this energy range and eliminate all other events. The *single-channel-analyzer*, SCA, is a unit which counts the number of pulses fed into its input and represents this information on a visual display. *Discriminator* is a device which responds only to input signals with a pulse height greater than a certain threshold value (Fig.2.2).

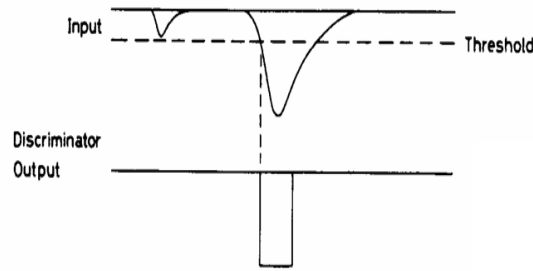


Fig.2.2 Discriminator operation : signals whose amplitude is greater than only signals whose amplitude is greater than the fixed threshold trigger an output signal

Still, this criteria does not reject all the background. In addition, particles scattered from other parts of the apparatus will also be close to the same energy and may also be accepted. More strict requirements can be made by using a second detector located behind the target directly in the beam line to eliminate the accidental. A stricter requirement can be made by using a second detector located behind the target directly in the beam line. A true scattering should give a signal in the first detector but not in the second. The logic for this trigger is now \overline{AB} , i.e., an anticoincidence.

2-1.2 Hardware implementation (Front-End-Electronics)

The incoming photons are measured and signals are amplified, for example, by the *photomultiplier tube* (PMT). Here electrons are emitted in the photocathode by the photoelectric effect and accelerated onto the dynodes of the tubes. Analog information from PMT is then converted to digital information by the Analog-to-Digital Converters (ADCs) module. Generally, the ADC is a device which converts the information contained in an analog signal to a digital form for the convenience of storage and handling (Fig.2.3). The input to the ADC is sampled and the result is stored as charge on a capacitor. After a short interval, the capacitor is discharged at

a constant rate, producing a time proportional to the input charge. The time is measured by counting the number of oscillator pulses during the discharge interval.

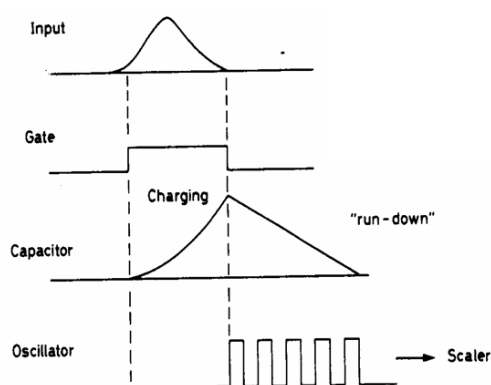


Fig2.3. The input signal is used to charge a capacitor which is then run down at a constant current. At the same time, a fixed frequency oscillator is gated on and the number of pulses generated during the time it takes the capacitor to discharge counted.

For example, the V965 ADC accepts input pulses in the range of 0 to 200 pC in low range and 0 to 800 (, the range should be something 200×8 or 200pC in the low range to be 100 pC for low range) pC in high range level are capable of outputting digital numbers from 0 to 4096(12-bit). The resolution of the ADC depends on the range of digitization. The maximum of voltage signal is digitized. It is used with current-generating devices such as a fast detector (PMT).

The time duration of integration or the time period over which the ADC seeks is usually determined by the width of a *gate signal*. Gate signal is generated from the *gate generator* [8] (Fig.2.4) ranging from a few nanoseconds to as long as few second depending on the width of input signal. The purpose is to inform the hardware (ADC module) to start the data taking. The gate signal would be inhibited while the ADC busy signal broadcasted or DAQ is busy.

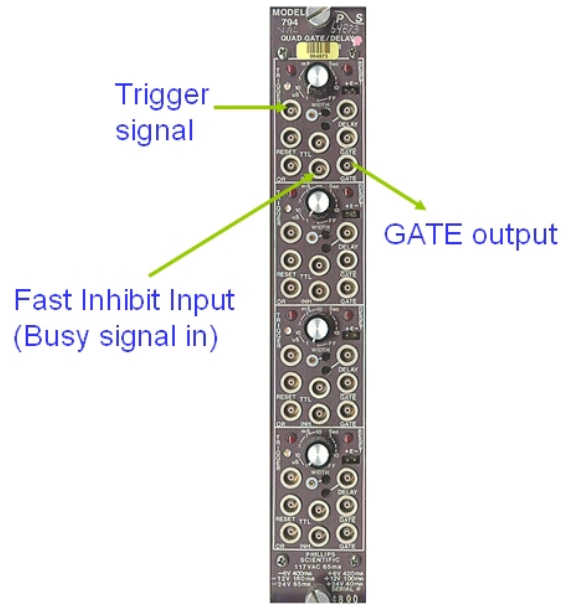


Fig 2.4 Schematic diagram of Gate Generator

2-1.2 Software implementation (Read-out system)

The software system controls the overall experiment. It includes subprograms for reading the data buffers from the ADC memory and for recording the data to disk or magnetic tape. User can monitor the pulse shape for checking the criteria is acceptable or not using online visualization and monitoring. User can have an ease way to program the overall of the data taking. It also offers user to make a pre-experimental run for data checking. In our experiment, **UNIDAQ** is used as the software tools for DAQ system. Fig 2.5 shows the Global structure of SL-DAQ system which is composed of 3 levels :

1. Trigger Level, 2. Hardware Level, 3 Software Level.

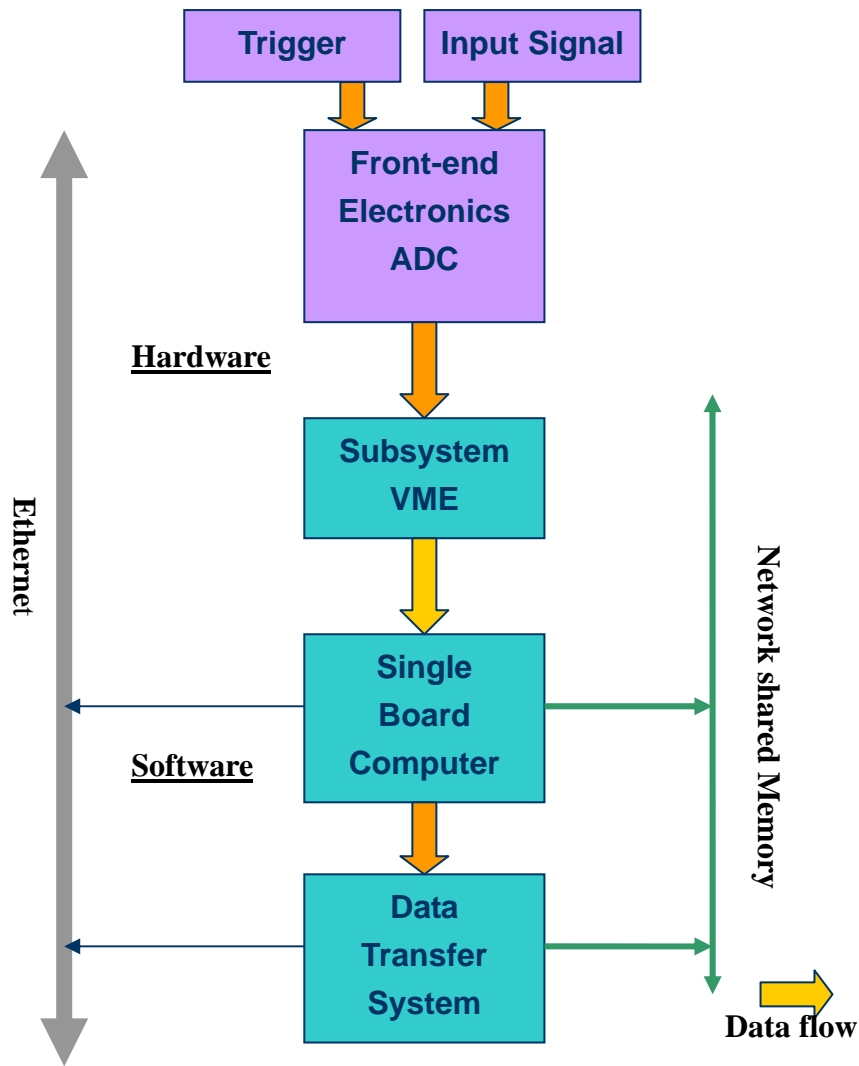


Fig.2.5 Global structure of SBSL DAQ system

In the next section, we will illustrate how we build up the instrument setup and construct the DAQ system in SBSL experiment.

2 - 2 Experimental Apparatus

2-2.1 Introduction of devices in SBSL experiment:

Before the data taking by our DAQ system, there are three basic apparatus what we used in the SBSL experiment.

1. *Function generator*: a signal generator that can produce sine, square, triangle and saw-tooth output waveforms with variable frequency and amplitude.
2. *CCD monitor*: charged-coupled devices to store the information from the sense wires for later readout by slower ADCs. There are silicon devices consisting of a two-dimensional array of tiny potential wells each covering a surface area of a few square microns. One chip contains tens of thousands of elements. When struck by radiation, electrons are released which shifting the charge from one well to the next until it reaches the output electronics.
3. Oscilloscope: Before the data taking in SL experiment, a real-time signal check is needed to clarify the pulse shape threshold by oscilloscope.

2-2.2 Preparation of Water

In order to evaporate the gases inside the water, the flask of water should be placed at the vacuum chamber for removing the gases. While the pressure inside the chamber is reached to 10^{-3} atm, the chamber can be opened and the flask can be took off again. Here the water was ready for the SL experiment. Cover the bottle before cool down and placed the bottle in the fridge to keep its temperature around 4°C , drop water into the flask.

2-2.3 Instrument Operation

Turn on the function generator, we can set the output amplitude and frequency at about 10 ~15V and 25~27KHz. Turn on the oscilloscope and move the iron core of inductance to adjust the voltage and current signals in the same phase. We then placed the 4 sensitive PMTs at the separation angle of 15° to measure the angular correlation of the SL-light (Fig 2.6, 2.7).

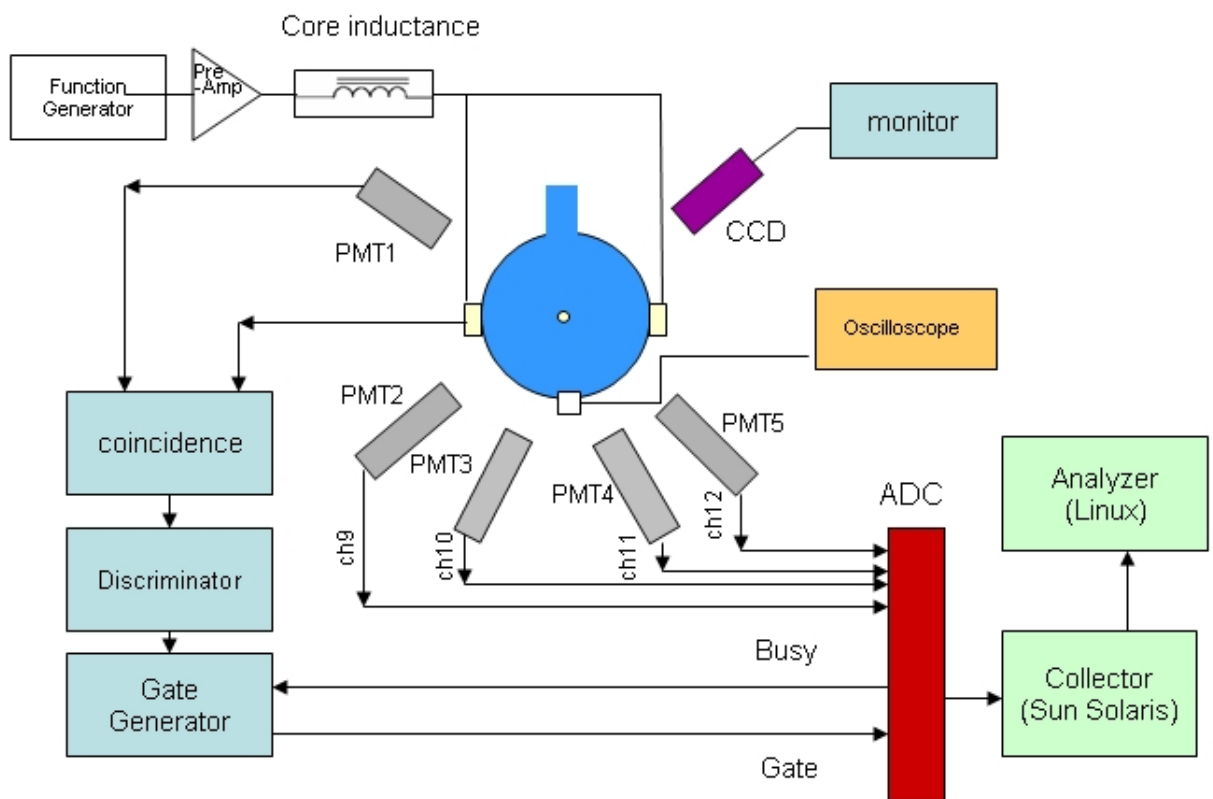


Fig. 2.6 The figure shows the setup and the DAQ system architecture for the SBSL experiment in Academia Sinica.

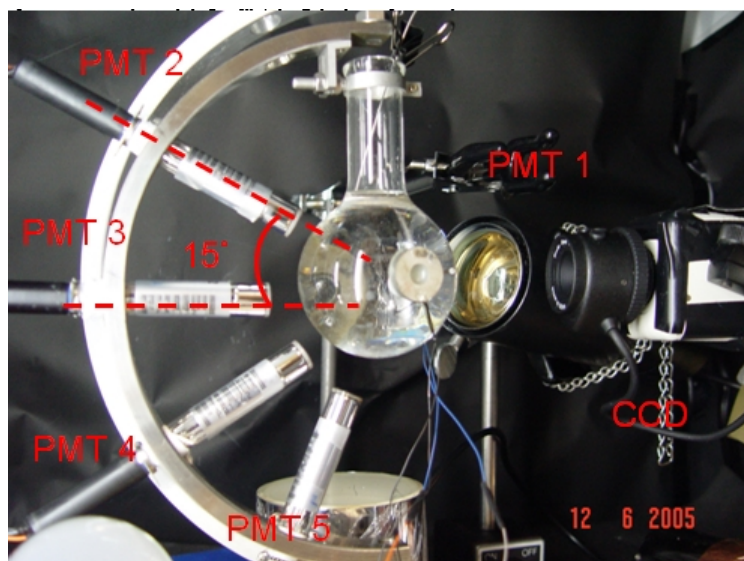


Fig.2.7 The 4 sensitive PMTs placed at an separation angle of 15° in the fridge of 4°C .

2-2.4 Identify the Sonoluminescence

By adjusting the oscillating frequency slightly, we can observe that the bubble move slightly. If the frequency is not in resonant mode (flask plus liquid), the bubble motion looks unstressed. It also moves freely and goes upward to the liquid surface finally. Conversely, if we approach the right frequency, the bubble moves quickly in a small region and trapped by the acoustic sound produced by the piezoelectricity ceramics. Fine turning the frequency, the bubble would stop suddenly and looks bright and stable. Now, you can turn off the light to see whether the CCD monitor have a bright point, if the background is completely dark, that point is the sonoluminescence bubble (Fig 2.8).



Fig.2.8: Photo of SL light by the snapshot of CCD monitor

2-3 Data Acquisition in SBSL experiment

In SBSL experiment, the rate of the input acoustic sound wave from the function generator is at about 25kHz~27KHz but the duration of sonoluminescence emission light is around 50-150ps [9]. The ADC sampling rate is obviously not high enough to trigger such input signal. In order to obtain a high trigger rate of SBSL experiment, we selected the ADCs (CEAN model V965) to settle the problem. The resolution of ADC is 12-bit. The V965 ADC module (Fig.2.9) is a 1-unit wide VME 6U module, it is a commercially ADC module developed by CAEN [10]. It houses 16 Charge to Digital Conversion channels with current integrating negative inputs (50 Ω impedance). For each channel, the input charge is converted by two ADCs in parallel; one ADC is preceded by a x1 gain stage, the other by a x8 gain.

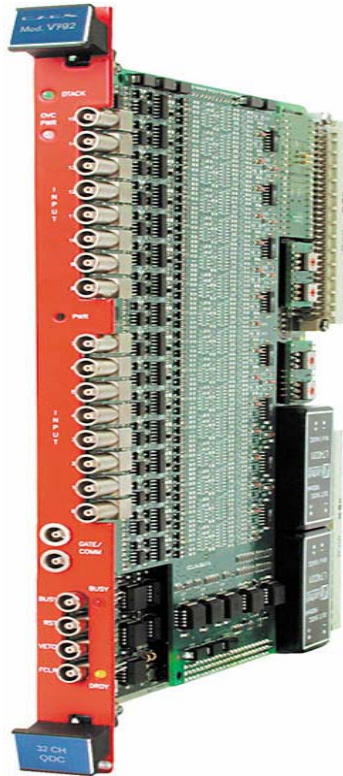


Fig.2.9 Photo of the 16-channel ADC module

2-4 ADC readout scheme

Data from PMT are read out by the ADC (V965) module. The ADC readout digitized the signals with a conversion time at $\sim 6\mu\text{s}$ (Fig2.10). The module hosts 16 QAC (Charge to Amplitude Conversion). The integrator output signal will be amplified and digitized until the charge is injected into the capacitor again. The maximum gate width is designed for $\sim 40\mu\text{s}$ time width. There also is the *Pedestal* value readout when no input signal is present. It's due to the internal current integrated by the capacitor when a GATE is applied.

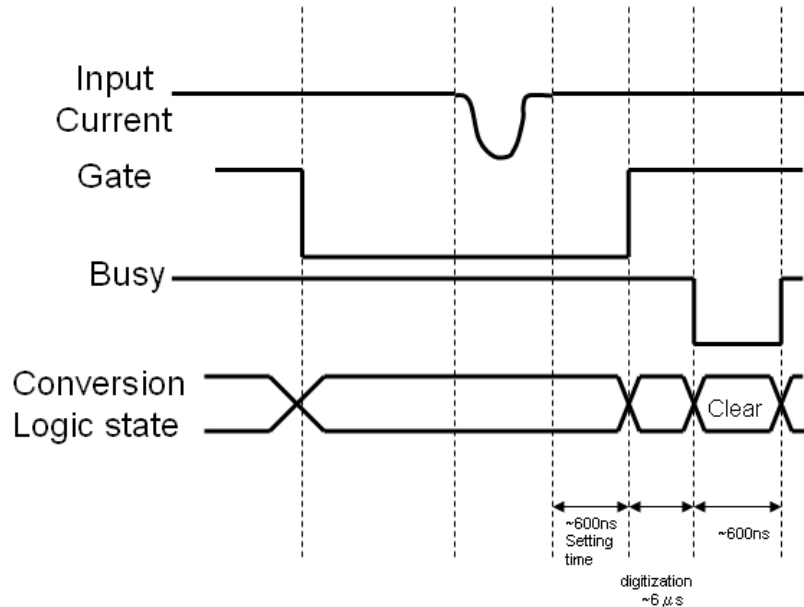


Fig.2.10 Signal conversion timing

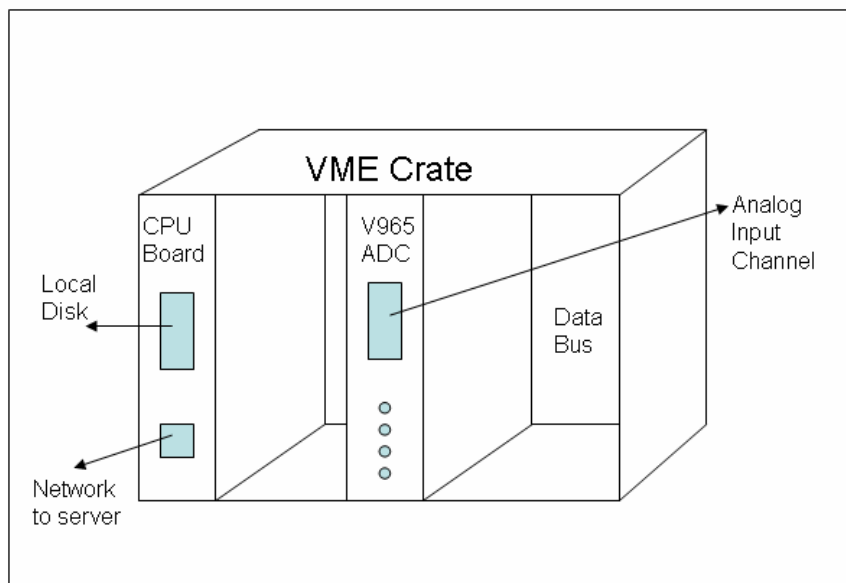


Fig.2.11 A readout setup from the ADC system to the Single Board Computer.

Fig.2.11 shows the readout setup of V965 ADC and VME single board computer. (SBC, UltraSPARC-IIe 64bit SPARC V9, Fig 2.12 manufactured by Force Computer company [11]). With the logic signals from the trigger system, the data in the memory of the ADC are read out by a VME CPU-board of SPARC V9 through a VME-bus.



Fig2.8 Photo of the Single Board Computer, UltraSPARC-IIe 64bit SPARC V9

2-5 VME-bus

VME-bus(*VersaModule Eurocard bus*) , a 32-bit bus developed by Motorola, Signetics, Mostek and Thompson CSF. It is widely used in industrial, commercial and military applications with over 300 manufacturers of VMEbus products worldwide. VME64 is an expanded version that provides 64-bit data transfer and addressing.

2-6 Buffer management

The V965 ADC module has designed the event format in 3 parts: 1.Header, 2.Data Words, 3.End of Block. The converted digital value in the Multiple Event Buffer (MEB) can be read by the following data structure :

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GEO[4:0]				0	1	0	CRATE[7:0]							0	0	CNT[5:0]															

Output buffer: Header

V965																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GEO[4:0]				0	0	0	0	0	0	0	CHANNEL[3:0]			RG	UNOV		ADC[11:0]														

Out buffer: Data Word format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GEO[4:0]				1	0	0	EVENT COUNTER[23:0]																								

Out buffer: End of Block

Fig.2.13A Buffer format in a event of the Multi-Event Buffer (MEB)

Fig.2.13 shows that the data format in ADC buffer. The output buffer contains the output data organized in 32-bit words. The data in the buffer are organized in events. Each event consists of header, data words, End of Block (EOB). The purpose of the event format is for separating the data into different events and channels. The data format for one event would be shorter than that in full 16 input channels. The header here contains the geographical address, the crate number and the number of converted channels. One or more data words contains the geographical address, the number of the channel, the Under-Threshold (UN) bit, the Overflow (OV) bit and the 12-bit converted value. The End of Block (EOB) contains the geographical address and the event counter. The sequence of the data storage in the buffer follows as:

- Header**
- Channel 0 High Range**
- Channel 8 High Range**
- Channel 0 Low Range**
- Channel 8 Low Range**
- Channel 1 High Range**
-**
- Channel 7 Low Range**
- Channel 15 Low Range**
- End of Block**

The data structure is sequentially stored in the ADC Buffer. Data will

be packed in multi-event buffer (MEB) format, event by event filled up to the ADC Buffer. Once the ADC buffer reached the specified value or the number of events is full for the *Interrupt* condition (condition for the number of events being transferred), the ADC data is needed to read out by the Single Board Computer (Sun SPARC V9) and cleared the buffer again. The data output size can be reduced by the threshold of pedestal level cut.

3 UNIDAQ: Software for Unix-Based Data Acquisition

UNIDAQ was a distributed Data Acquisition system for UNIX-flavor platforms. It was designed in 1993 by the UNIDAQ group, which included KEK, LBL, U.Michigan, SSCL and Tokyo Inst. of Tech. The Solenoidal Detector Collaboration (SDC) required the data acquisition capabilities at many locations in reduced costs. But unfortunately, the computer environment on those locations was very different from each other such as SUN under SUN OS and Solaris, DEC under ultrix and Alpha OSF-1, SGI and VxWorks platform. The SDC had to define the base-line so as to provide uniform capabilities, high performance system and portability.

UNIDAQ runs on standard UNIX. It consists of many processes that can run on one or more machines, communicating with each other through *message queues* and *shared memory*. Data is passed from process to process using a buffer manager, NOVA. Critical event variables are saved in the shared memory and stored on disk. The early design of UNIDAQ is shown in Fig.3.1. [12]

Message queue: Connection link for one or more different machines.

Shared memory: Block of memory could be shared for one or more different machines.

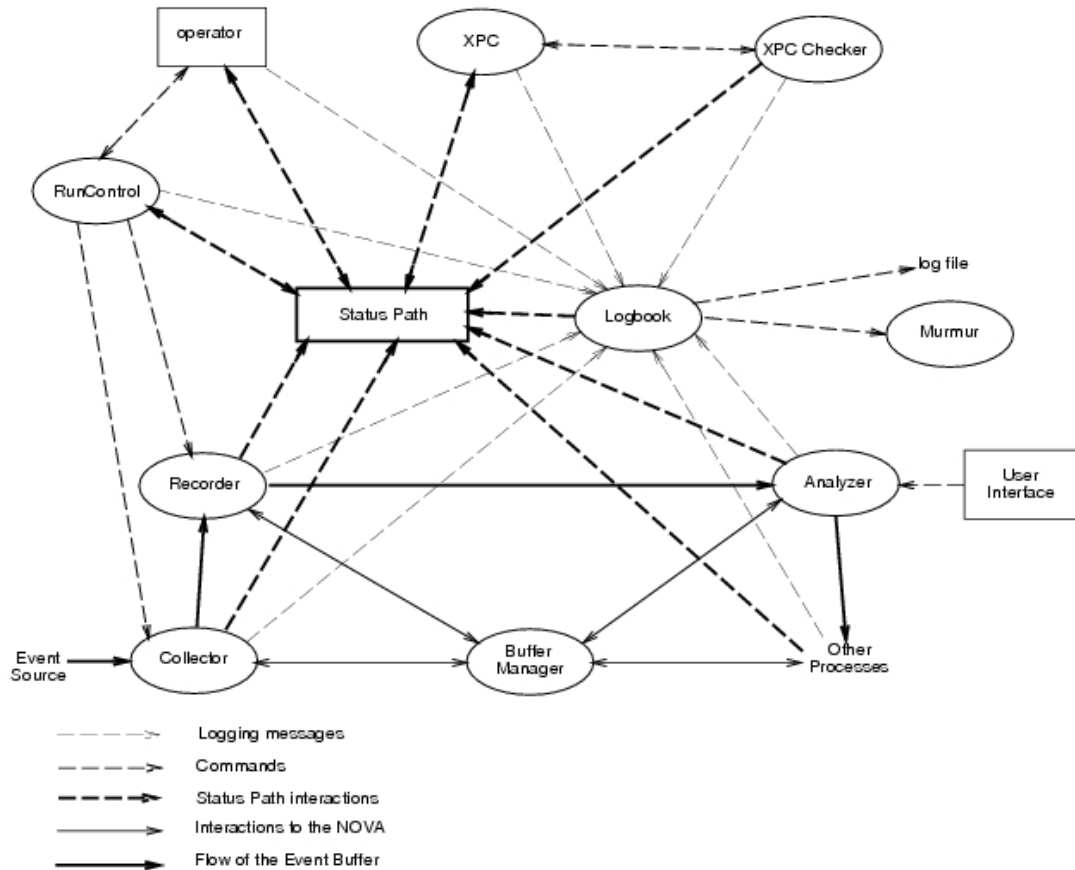


Fig.3.1 The interaction of processes in the early UNIDAQ version

Fig.3.1 shows the overall system software consists of a generic template structure, a buffer manager, and a set of processes such as collector, analyzer, recorder which were developed using the template and the buffer manager. The processes running in UNIDAQ interact with each other by means of three entities: command messages, event data, and process data.

3-1 Data and control flow

The Multi-Event Buffer (MEB) starts to poll data from the ADC module. The collected data are stored in buffers and sent to shared memory on the **Collector** PC (Single Board Computer), Lep5401 (see Fig.3.2), which is managed by the **NOVA** process. The NOVA process passes a pointer to the

Recorder and **NetNOVA** processes as a message with Inter Process Communication (IPC), so that they can transfer the data to each destination. IPC is the techniques for processes to communicate with each other process in UNIX-based system.

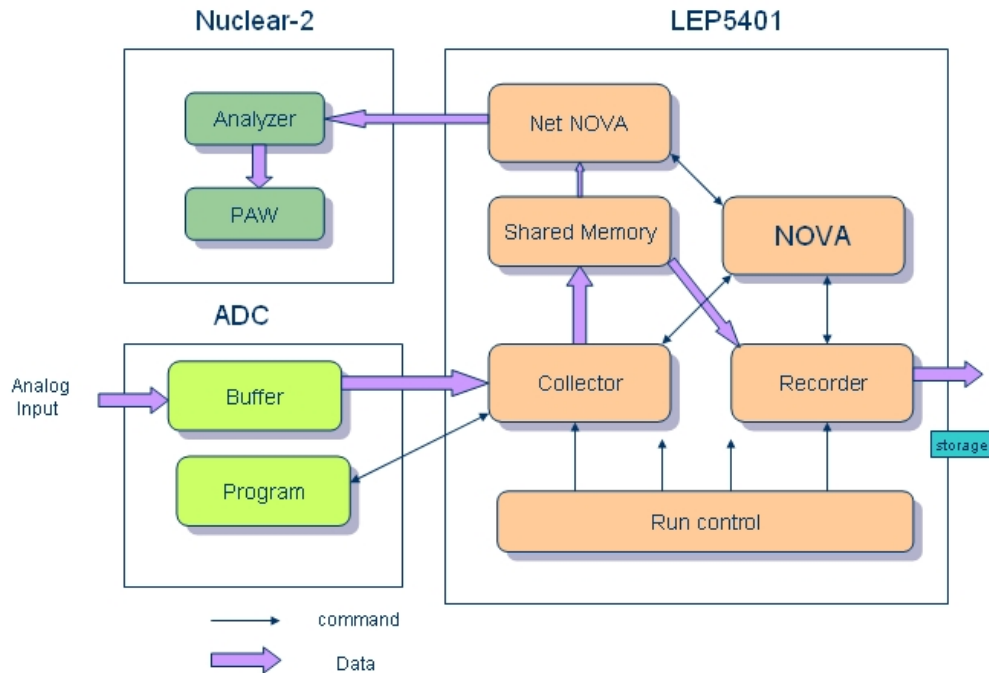


Fig.3.2 Block diagram of the data-acquisition system. The arrows indicate the data and control flow of the system

There are three ways of running UNIDAQ in a single CPU-system. The first option is to use the setup created during the installation of the software. The install-scripts create a nodefile containing only the node that user are running on (for example, nuclear-2 and Lep5401 in SL-DAQ system). The installation creates a Start_node script for the local node, in which all distributed process are started.

3-2 The Recorder Process

The **Recorder** process saves collected data to a hard disk, while the

NetNOVA process transfers a part of the data to the Remote-PC via the Ethernet (Fig.3.2). The real-time data can be stored and monitored with Physics Analysis Workstation (PAW) [13], developed at CERN (European Organization for Nuclear Research) on that remote Linux platform. PAW assists physicists in the analysis and presentation of their data. It provides interactive graphical presentation, statistical and mathematical analysis, working on objects familiar to physicists like histograms, event files, vectors, etc. The stored data provided the offline analysis by decoding the data format described above (see also in 2-6).

3-3 The Collector Process

The **Collector** process communicates with buffer control program. Data **semaphore** is used for access control of the memory buffers. The semaphore is a counter used to provide access to a shared data object for multiple processes to coordinate the use of shared memories [14]. The buffer control program can retrieve the buffer when data semaphore is available, and return the semaphore when the buffer is full or the trigger is received. On the other hand, the Collector process needs a semaphore during data transferring, and returns it while the transfer completed. The Collector process issues a command to the communication area, while the status register generates the reset signal at the end of each data acquisition [15].

3-3.1 Modifying the Collector Process

Fig.3.2 shows that the collector process runs on the Sun Solaris platform

of the single board CPU. There are six routines which can be called in sequence by the collector process in SL-DAQ system:

1. *user_begin* – Called to process a BEGIN command to inform the Single Board Computer that the ADC is ready of data taking.
2. *user_resume* – Called to process a RESUME command to enable the trigger acceptability for the ADC module.
3. *user_event* – Called to process an event for data acquisition. The event length is returned from *user_event* routine, and is the number of words placed in the event buffer.
4. *user_end* – Called to process an END command to inform that the Single Board Computer stop the data taking.
5. *user_init* – Called to set the maximum length which a *user_event* call may fill and initialize the V965 ADC module.
6. *user_exit* – Called to process whenever the user would like at program exit.

All of these routines are called back for UNIDAQ and coded in “*user_event.c*”(refer to Appendix C).

3-4 The Analyzer Process

The analyzer program was early developed in UNIDAQ on SUN Solaris operating system. In our study, we selected the Linux (Red Hat 9.0) as our analyzer platform. In order to install the analyzer in our system, a desirable modify and change is needed. There are 2 analysis tools for user to choose for analyzing the event data in Linux platform:

1. Off-line analyzer: Process is called by decoding the event data from the

stored data and for analyzing by PAW.

2. On-line analyzer: The on-line analyzer would be used to monitor the data transferring while the experiment started. The process is called by decoding the event data from the real time shared memory and for on-line displaying the event data by PAW.

3-5 The Message Template

The processes running in a typical data acquisition system deal with two types of activities : (1) Issuing commands to other processes, and processing the commands received; (2) Waiting for event data, and processing (updating) event data. Commands can be issued to a process by a user (from a user interface such as the “ask” tool). Event data is received in terms of event buffers, which are circulated among process by the buffer manager (NOVA).

The template provides the basic structure for the processing of command messages. It also provides a set of commands, and a set of variables to the user process. These are called system commands and system-level variables, since they are defined in the template kernel. Any user process developed using the template inherits the message passing mechanism, the system-level commands, and the system-level variables from the template. The user processes are able to define their own user-level commands, and variables (Fig 3.3).

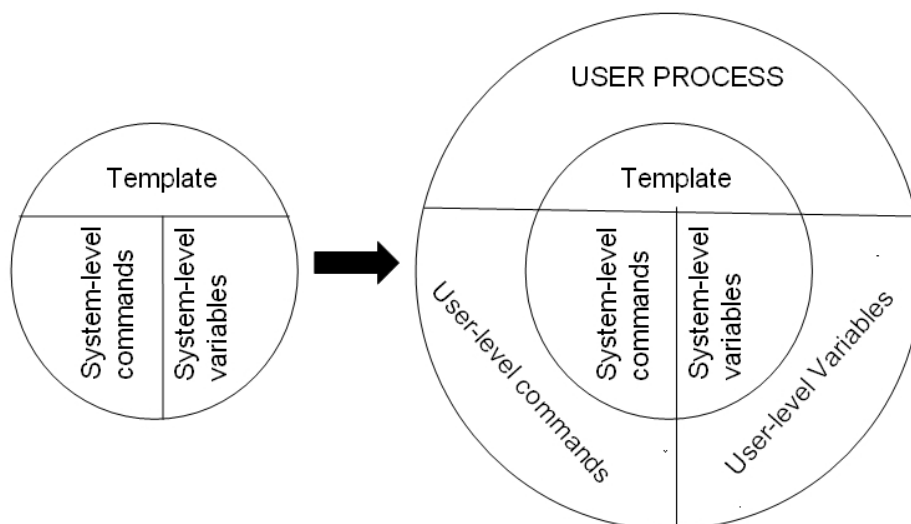


Fig.3.3 Developing user processes using the template

The following system-level commands and variables are provided on Linux version of UNIDAQ:

System-level commands: ASK, EXIT, SET.

System-level variables: process_name, status.

There is a simple example about the operating commands could be seen in **Appendix B1**.

3-6 The Event Buffers

The Collector process run on Lep5401 interacts with the event source. When an event occurs, collector accesses the event source (the ADC buffer), reads the related signals from the channels, and stores the event data into a NOVA buffer.

This data read is by the Collector process is named as an *event_data* record. It is then circulated among other system processes such as recorder and analyzer. Whenever the collector process receives a BEGIN command,

it prepares a *begin_run* record, and ships it to the other processes. Similarly, a *pause_run*, *resume_run* or an *end_run* record is prepared for each PAUSE, RESUME, END command, respectively. Each record type is distinguished from the others by the record format. The system contains 5 types of records: 1. Event data record, 2.Begin run record, 3.Pause run record, 4. Resume run record, 5.End run record (Fig.2.3).

Event Data Record

Buffer Length	Record Type	Run Number	Event Number	Mode	Reserved	Event Data (ADC MEB)
---------------	-------------	------------	--------------	------	----------	----------------------

Event Record Type = 0

Begin, Pause, Resume and End Record

Buffer Length	Record Type	Run Number	Event Number	Mode	Reserved	Record Related Data (time,etc)
---------------	-------------	------------	--------------	------	----------	--------------------------------

Begin Record Type = 1

Pause Record Type = 2

Resume Record Type = 3

Resume Record Type=4

Fig.2.3 Record Type and format

The first of the event data record is for collecting the MEB data from the ADC module described in previous chapter (Fig.2.10).It is written on the portion of Event Data. The other 4 data record types are for recording the information about the event related data of the “Begin, Pause, Resume and End records consists of a character time-stamp and the comments supplied with the equivalent commands to the collector process.

3-6.1 Number and Size of Event Buffers

By default, NOVA and 16kByte event buffers, and UNIDAQ will request records up to `max_length` words from each buffer until no more such can be filled, or a maximum timeout period has expired without an event. This maximum buffer size can be changed using the “-b” command line switch when the process is started. The value of `max_length` is set as collector starts by calling the `user_init` routine, which is coded in `user_event.c` (see in **Appendix C**) of the collector directory.

3-7 Structure of memory and control programs

The Multi Event Buffer (MEB) from the V965 ADC module is mapped as a communication area to the buffer area in Lep5401 while the start command of data acquisition (resume) is dispatched to the MEB. Once the start command is accepted, the MEB gets a buffer and waits for the Collector process check the status register at the beginning of the DAQ (resume) routine.

When the data acquisition is paused or restarted again, a “pause” command is issued which keeps all of the acquired data, scalar information, event log, and run number. They are continuously updated upon restarting. When the data acquisition is stopped, a “stop” command stores all of the acquired data and log files on the hard disk.

3-8 Installation of UNIDAQ on RedHat9.0 platform

In the SBSL experiment, UNIDAQ acts as a main role on the whole data acquisition. As time goes by, platforms such as SUN, SGI or HP become out of date or unpopular. In addition to the cost consideration, LINUX platform would be desirable. The following example is illustrated how UNIDAQ achieved at Red Hat 9.0 platform:

1. Update the Linux kernel version: 2.4.20-8
2. Install the GCC compiler version: 3.2.2-5
3. Download the old UNIDAQ v2.3 package from the High Energy Accelerator Research Organization (KEK) website:
<http://www-online.kek.jp/~online/Unidaq/unidaq2.3/>
4. Unpackaged it and modified the configuration file in the NOVA directory:
Add the parameter “ #define Linux ”
5. Modify the NOVA makefile: OS_name = Linux
6. Add the name “all.Linux” to the document directory
“/NOVA/doc/Makefile”.
7. A more detailed installation sequence of UNIDAQ could be referred in the **Appendix B2.**

Chapter 4 DAQ Test

In this chapter, we will introduce two experiments for testing and debugging the SL-DAQ system: the Co⁶⁰ Experiment and the LED Test Experiment. The purpose of this experiment is to test our DAQ system under a cross UNIX platform system and explain some of the physical features from the data. In this testing study, we need to determine the physical data out of the background noise: the previous background measurement is necessary.

4-1 Gamma ray source experiment

Before the data taking and analyzing by our DAQ system in SBSL experiment, a testing experiment is performed to debug DAQ system. In our study, we selected the Gamma-Ray experiment as our DAQ testing experiment. It is based on the measurement of Co⁶⁰ radiative source by sodium iodide (NaI) detector that is thallium activated (TI).

Most isotopes that are used for gamma measurements also have beta decay in their decay schemes. The typical decay scheme for the isotope will include a beta decay to a particular level followed by gamma emission to the ground state of the final isotope (Fig 4.1).

$$\text{Beta decay : } n \rightarrow p + e^{-} + \nu_e \quad (\text{Eq.4.1})$$

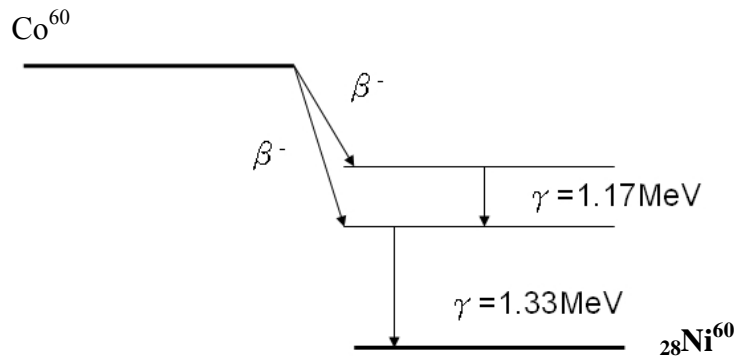


Fig4.1 Successive beta and gamma emission in the decay of ${}_{27}\text{Co}^{60}$ to ${}_{28}\text{Ni}^{60}$

The beta particles will usually be absorbed in the surrounding material and not enter the detector at all. Generally, there are two signals that we would like to investigate about a gamma source. One is the energies of the gammas from the source. The other is the background noise without the gamma source. Set up the electronics and DAQ shown in Fig 4.2.

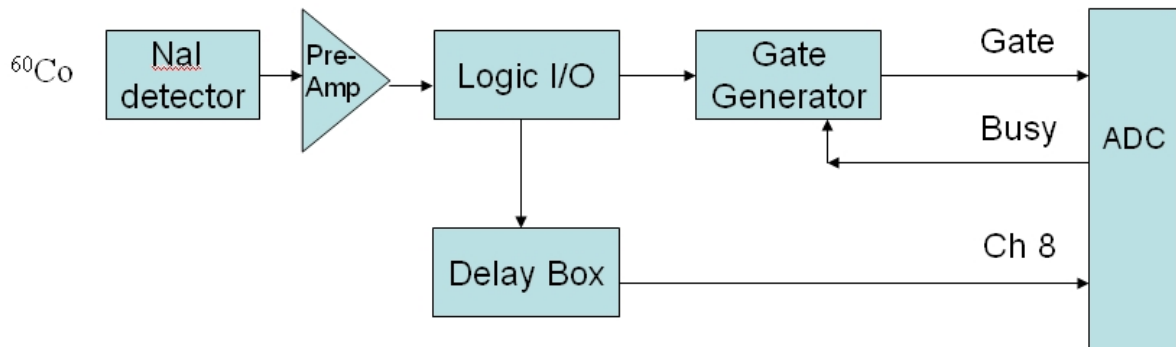


Fig.4.2: Electronic Block Diagram for Gamma-Ray Spectroscopy System with NaI Detector.

Here we placed ${}^{60}\text{Co}$ ($E_\gamma = 1.17$ and 1.33 MeV) source $\sim 15\text{cm}$ in front of the NaI crystal. Adjust the coarse, and fine gain controls of the linear amplifier so that the signal from the PMT would be amplified and clarified by the oscilloscope. The input signal peak falls at around 600mV with 400ns width range. It is just in the allowable input range for ADC (800pC). The data from ADC is processed to the Collector and Analyzer processes for saving and analyzing. Begin the DAQ for data taking and accumulate the

^{60}Co spectrum for a time period long about 15min to determine the peak position by **Off-Line** analyzer (Fig.4.3). Fig.4.3 shows a typical ^{60}Co spectrum that has been plotted.

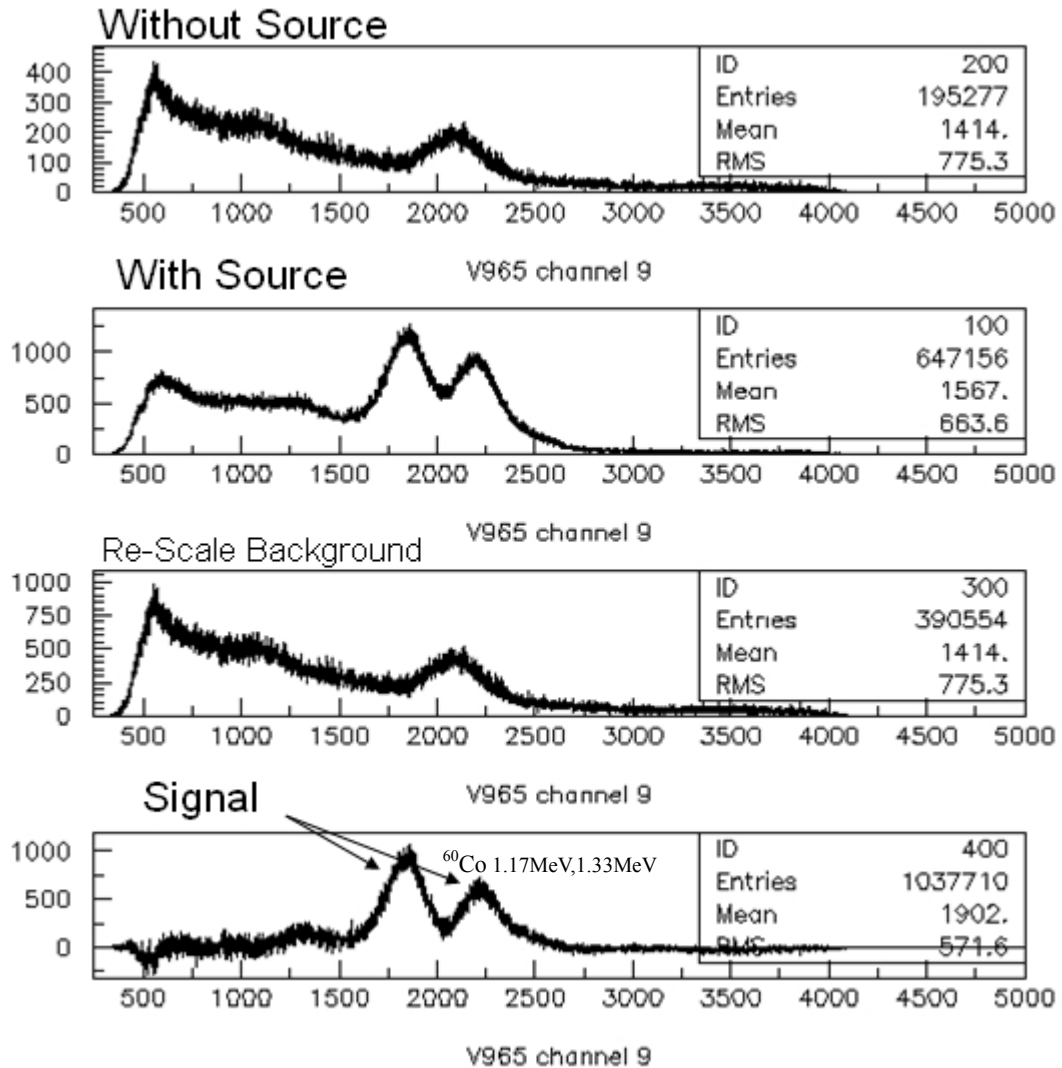


Fig.4.3 The background measurement and the Co^{60} spectrum with background cut-off

4-2 LED Test Experiment

LED (light-emitting diode) light emitted at a p-n junction is proportional to the bias current. Charge-carriers - electrons and holes flow into the junction from electrodes with different voltages. When an electron meets a hole, it falls into a lower energy level, and releases energy in the form of a photon as it does so.

The LED Test is an important job for testing the SL-DAQ system. The configuration of the LED Test setup is shown in Fig.4.4. The LED light is generated by the pulse generator input. By changing the amplitude of the input signal from pulse generator, we can obtain the varying input ADC value in the SL-DAQ system.

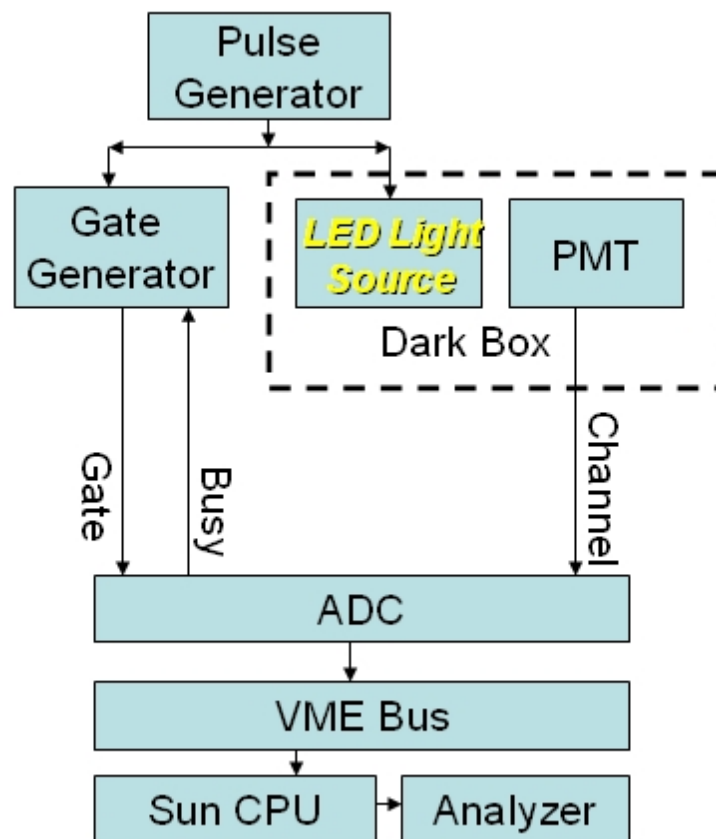


Fig.4-5. Schematic diagram of LED-Test measurement

Fig.4.5 shows the extra charge value in the lower ADC level. The problem here is caused by the busy trigger conditions. To remove the unusual events taken in the overall DAQ processes, we can reduce the trigger rate (the signal rate of the pulse generator) to be shorter than the $6 \mu s$ ADC conversion time and lower the signal threshold level by adjusting the Discriminator.

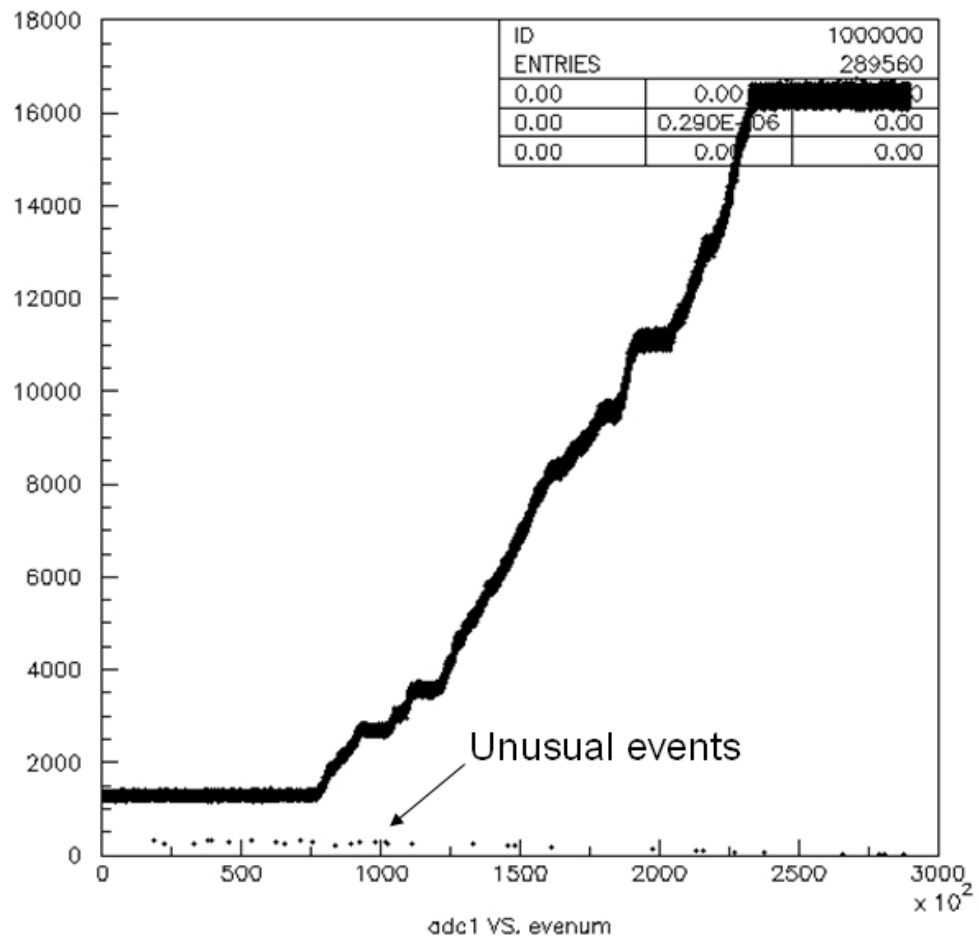


Fig.4.5 The unusual events being debugged in SL-DAQ system

Chapter 5 Results and Discussion

In this Chapter, we will discuss the event data from the SL experiment by the offline analysis. At the beginning, we would like to illustrate briefly about the flow path of the whole experiment (Fig.5.1).

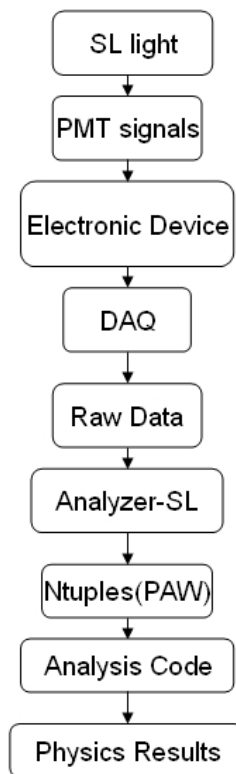


Fig.5.1 Offline analysis flowchart

In Fig.5.1, the SL-light emitted to the PMT detectors and produced the electronic signals. The electronic signal would be gathered by the DAQ system and compacted into the required data format (raw data). The data format recorded all the physical information in the total experiment but in digitized version. Here, the raw data would be stored by the recorder process in the cross-platform system as described in the pervious Chapter (in Linux

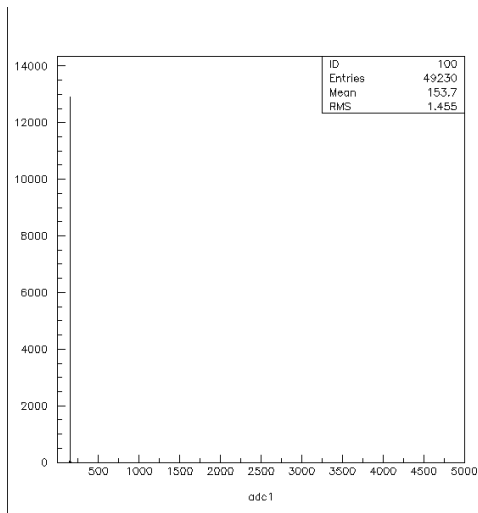
OS). Afterward, we use the Analyzer-SL code for decoding the raw data and then generate the ntpules. By some analysis code for reconstructing the physical events, we can get the “Physical Results” to analyzer and learn more about the Sonoluminescence.

In the coming sections, we will describe about the SL-background and the SL-light signals from the different runs. We will illustrate why we need to understand more about the experimental condition. By comparing the ADC values between 2 PMTs, we can obtain the correlation of light respected to the distribution of the ADC value.

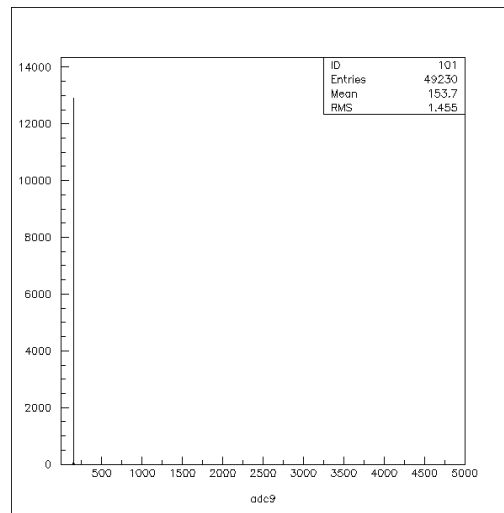
5-1 Background Measurement

The backgrounds were referred to the electronic noise and the background light influence in the SL experiment. In order to obtain the reliable data, it is necessary to recognize the input signal and backgrounds before starting the experiment. Fig.4.1 showed that the backgrounds of 5 PMTs (PMT 1 ~PMT 5) in the configuration setup as Fig.2.4. The PMT signals were connected to the ADC channels (PMT1 to channel 1, PMT 2 to channel 9, PMT3 to channel 10, and so on). The purpose of these figures is to show the number of counts of ADC value taken in the Run of 60 seconds without the SL light. In this measurement, we could know that there are no background signals except for the electronic pedestal on the lower level at the SL trigger conditions (26.2 kHz).

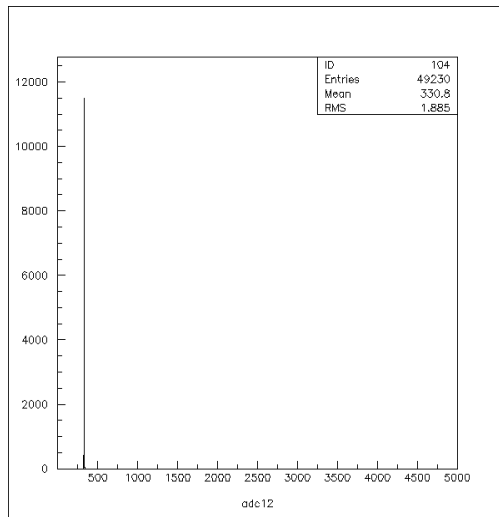
PMT 1



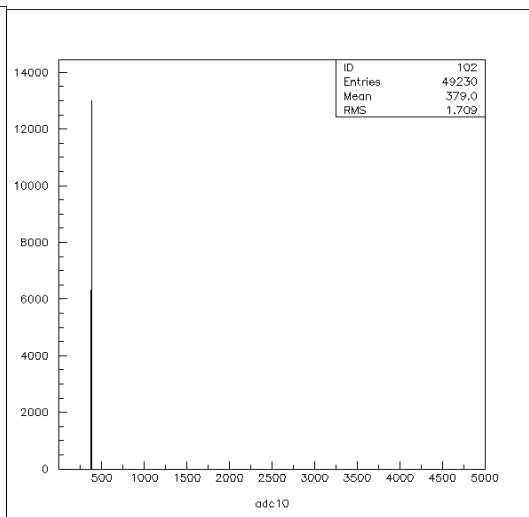
PMT 2



PMT 3



PMT 4



PMT 5

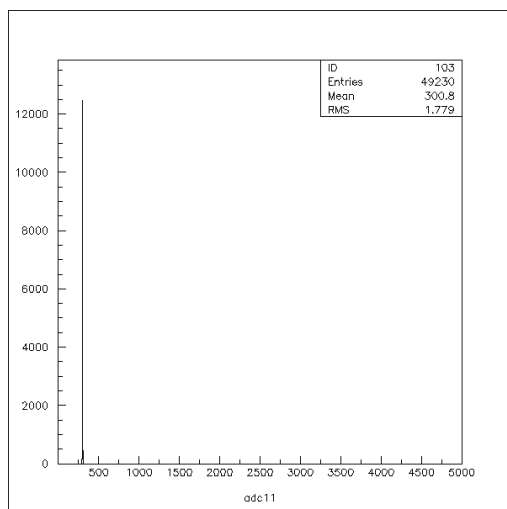


Figure 5.1: The pedestal level for 5 Photomultiplier tubes in SL-experiment

5-2 Sonluminescence Measurement in 1-D histograms

In this section, we will discuss 3 runs for the duration of 60sec, 300sec and 600sec long what we took in the SL experiment.

The Run #1 :

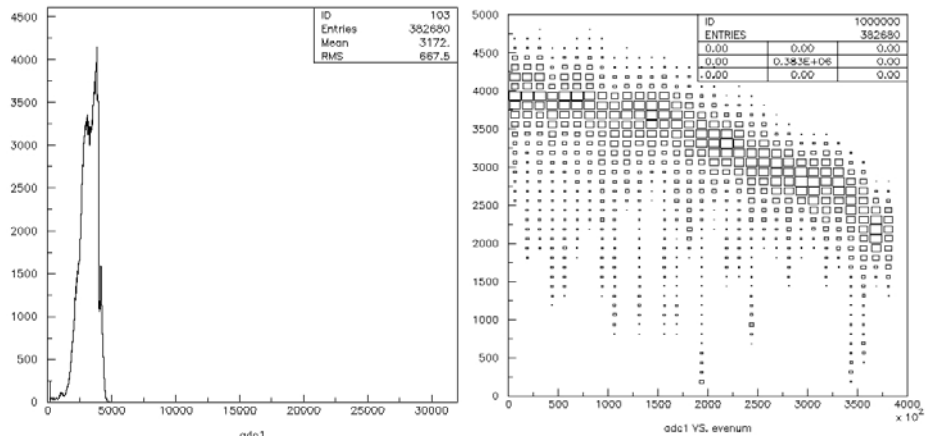


Fig.5.2.ADC distribution for 30 seconds of PMT 1

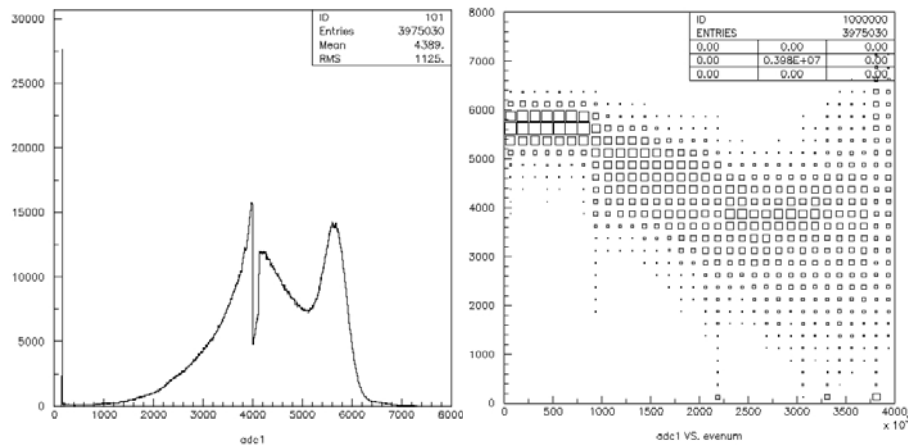


Fig.5.3 ADC distribution counts for 300seconds of PMT 1

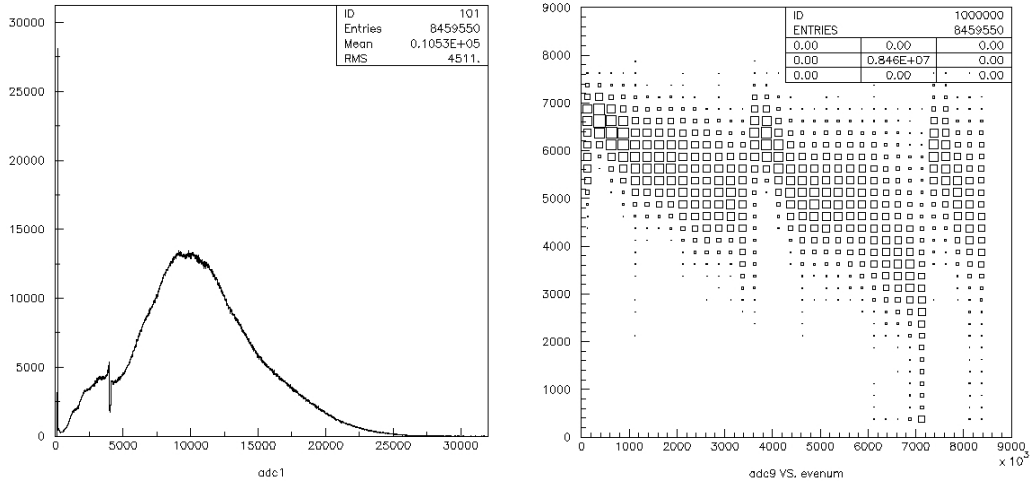


Fig.5.4.ADC distribution for 600 seconds of PMT 1

In the figures above, it is found that the signal is not stable across the whole period of the run and there is some fluctuation on the channel near to 4000 and the overall fluctuation becomes large. There are also the pedestal signals on the all figures compared to background figures (Fig.5.2~Fig.5.4). The reason is that the pedestal level would be too low. We can solve this by adjusting the discriminator.

The Run#2

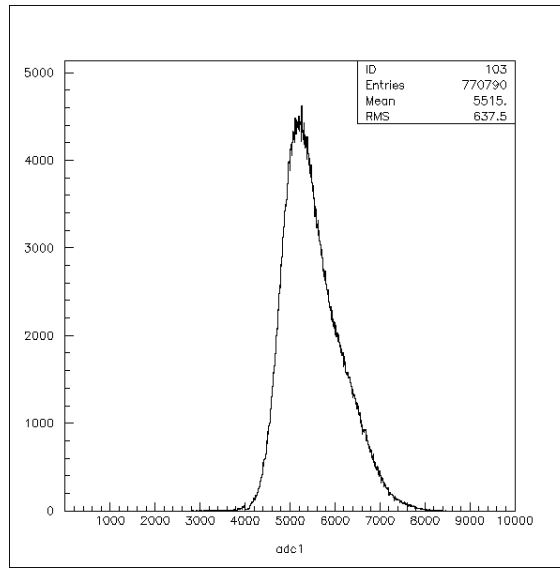


Fig.4.5 The PMT 1 (to ADC channel 1) Q value VS number of counts in 60 sec

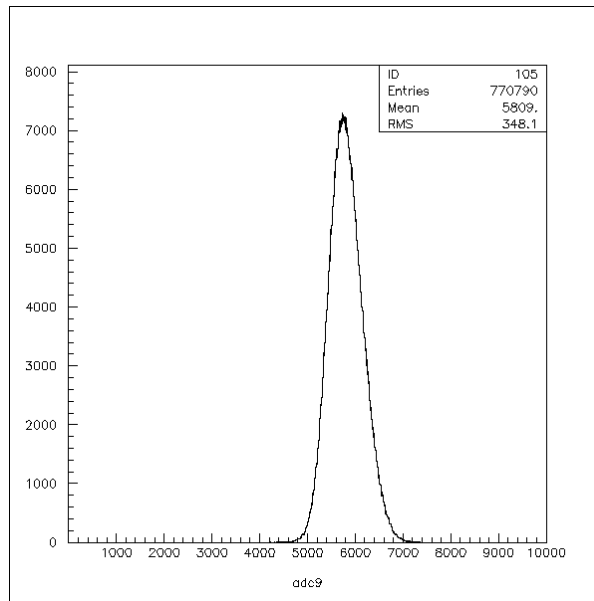


Fig.4.6 The PMT 2 (ADC channel 9) value versus number of counts in 60 sec

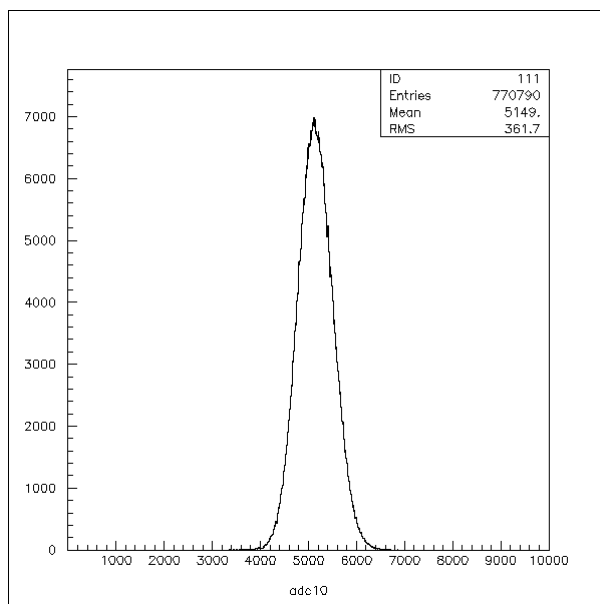


Fig.5.7 The PMT 3 (channel ADC10) Q value versus number of counts in 60 sec

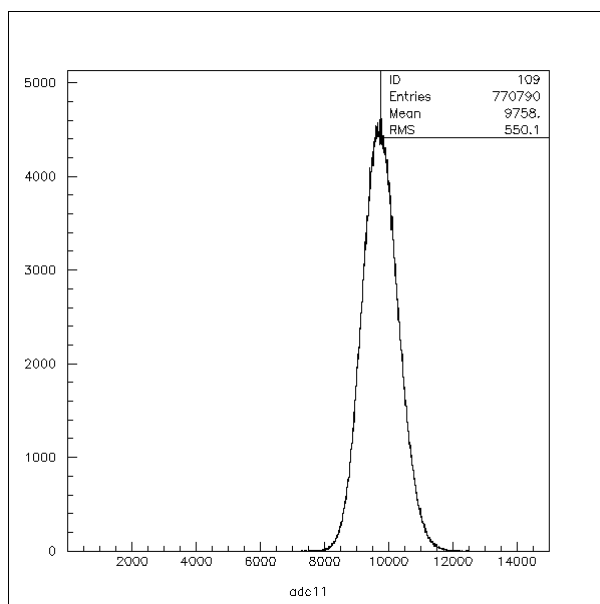


Fig.5.8 PMT 4 (channel ADC 11) Q value versus number of counts in 60 sec

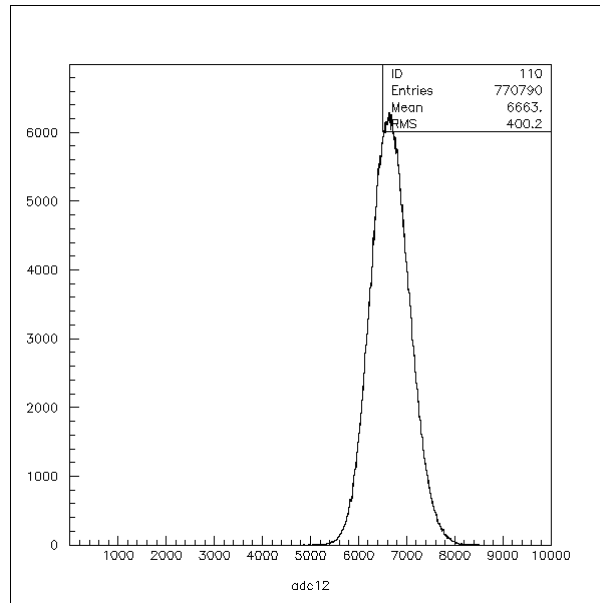


Fig.5.9 PMT 5 (channel ADC12) Q value versus number of counts in 60 sec

From Fig 5.5 to Fig 5.9, we show that the converted charge value from 5 photomultiplier tubes versus to the counts of data taken in 60 seconds. In each figures, we can see the sharp peak obtained and no any background noise appear. It means that the signal is very clear and the environment is completely dark.

5-3 ADC value versus Event number

By comparing the ADC Q value gathered by the 5 PMTs to the event numbers, we can determine the stability of the SL light while starting the DAQ. In figures (Fig.5.6, Fig.5.7), it is found that the SL light is stable in the time for 60 seconds.

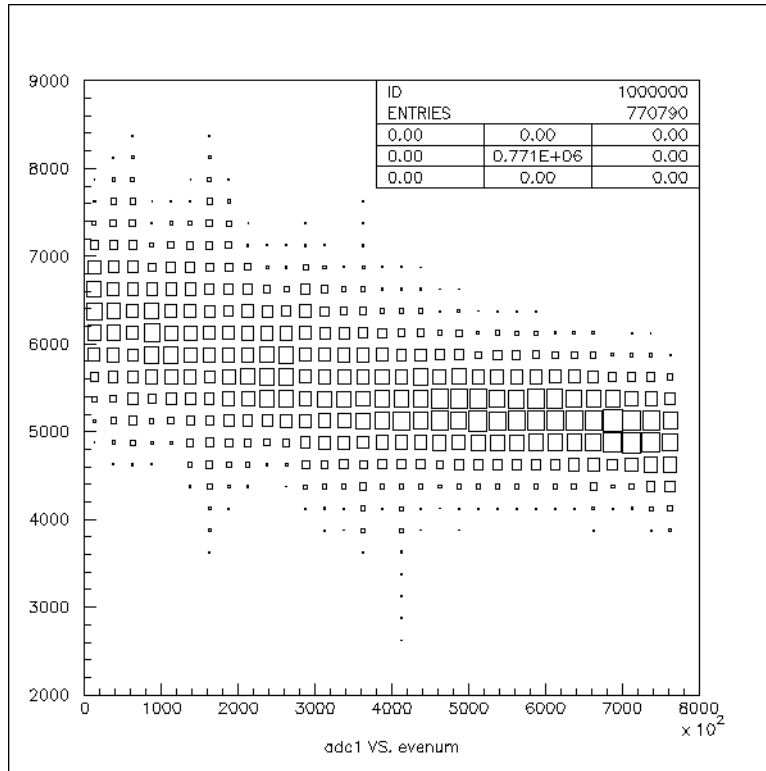


Fig.5.6 Diagram of the PMT 1 versus number of event

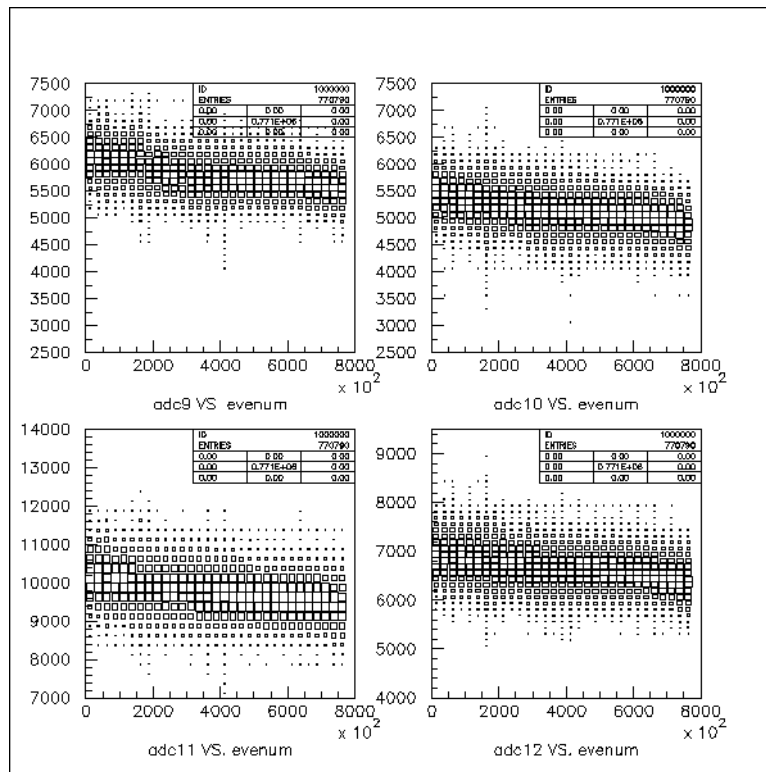


Fig.5.7 Diagram of the PMT 2 (channel 9) to PMT 5 (channel 10) versus number of events

5-4 The comparison of the correlation between 2 PMTs

Here we can compare the ADC value pair by pair of the PMTs to see the intensity correlation between 2 different directions of the SL light. The angular difference is 15° between 2 PMTs. From Fig.5.8 to Fig.5.12, we found that there is perfect correlation to both different PMT directions. It means that the photons which gathered by the PMT in the same duration were 100% correlated. Thus, there is no independent background events could be seen in the “No Event” regions. It means that the HBT effect is maximal when the particles are emitted randomly and the correlation function related to the extending angle θ would be plotted in a straight horizontal line (refer to the Fig.1.2).

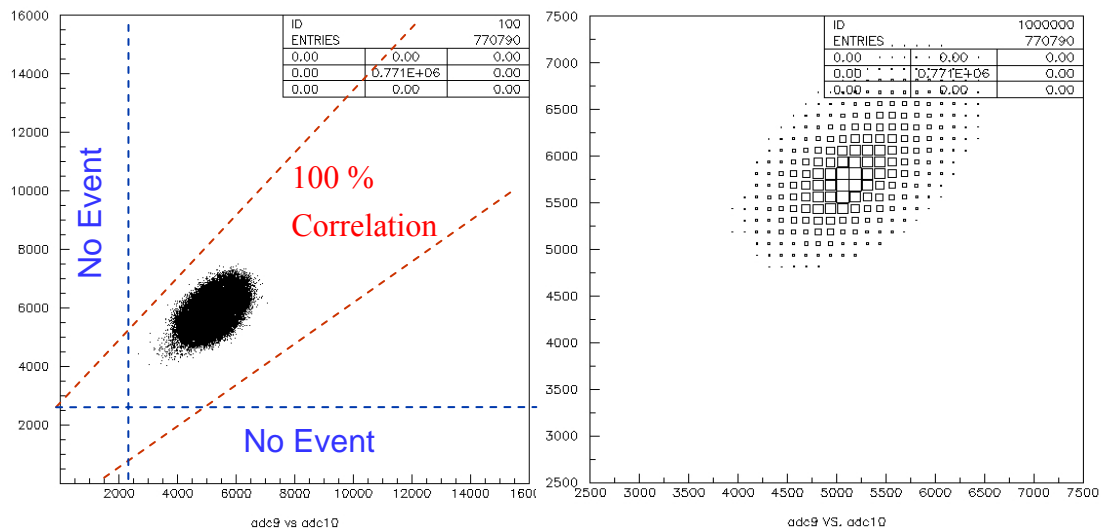


Fig.5.8 Diagram of the SL light correlation between PMT2 (channel 9) and PMT3(channel 10).

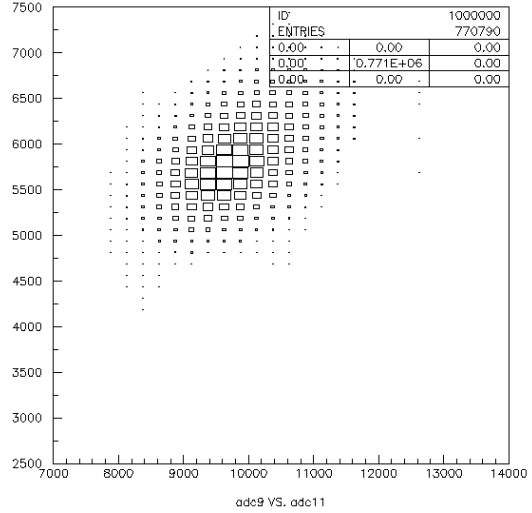
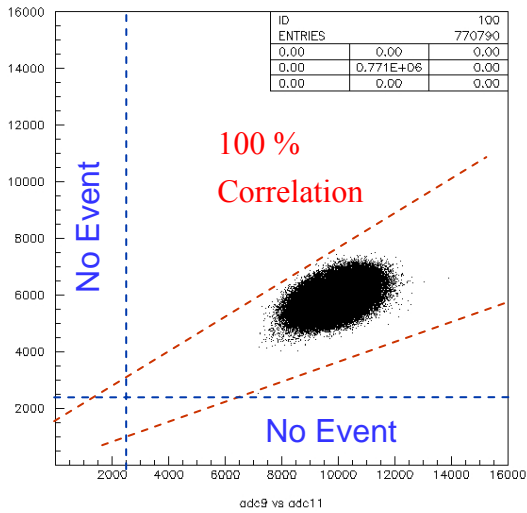


Fig.5.9 Diagram of the SL light correlation between PMT2 (channel 9) and PMT4(channel 11).

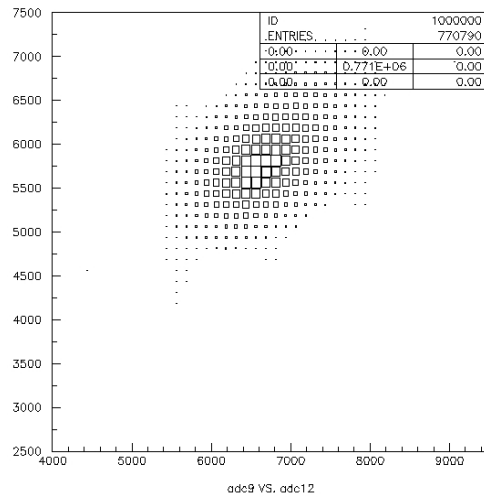
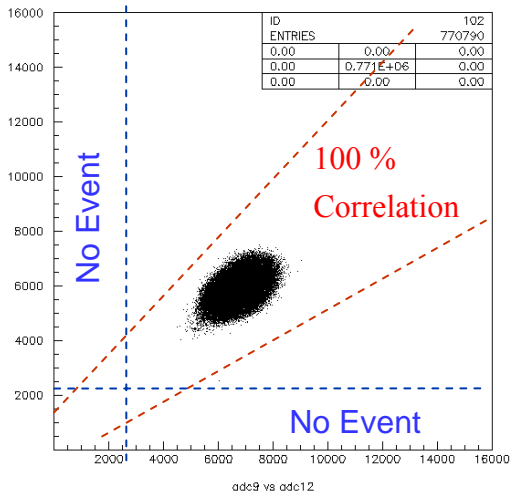


Fig.5.10 Diagram of the SL-light correlation between PMT2 (channel 9) and PMT5(channel 2).

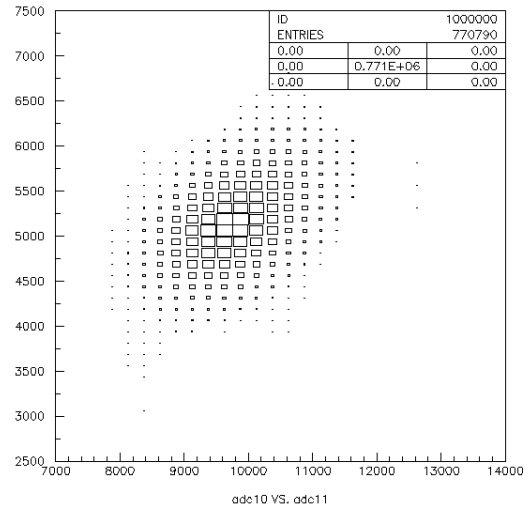
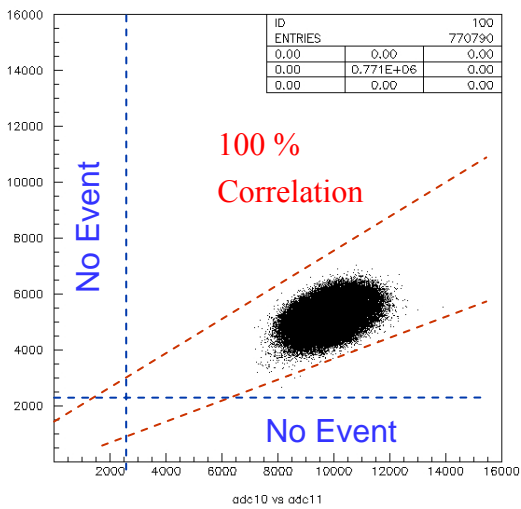


Fig.5.11 Diagram of the correlation between PMT3 (channel 10) and PMT4(channel 11).

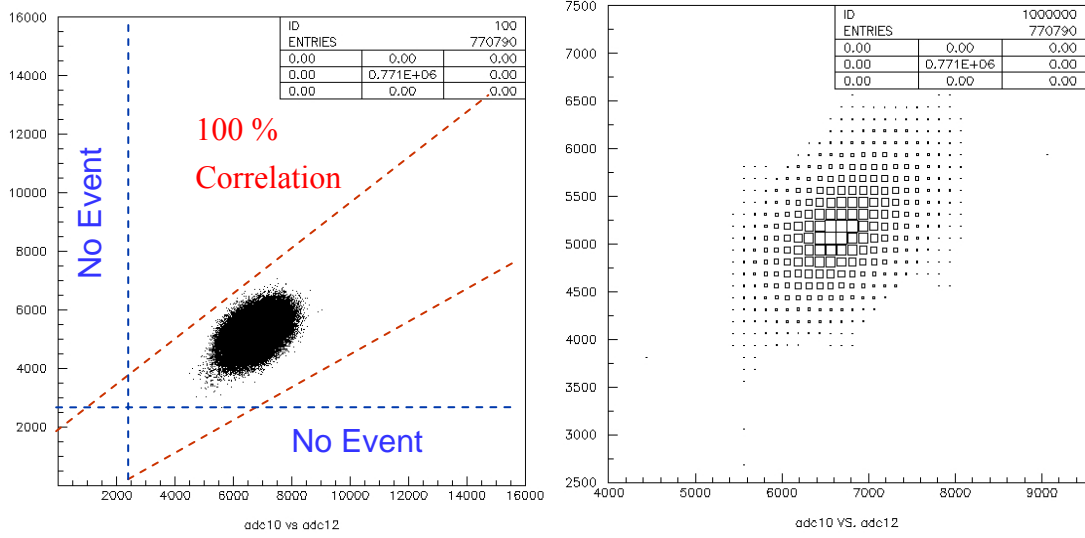


Fig.5.12 Diagram of the SL-light correlation between PMT3 (channel 10) and PMT5(channel 12).

5-5 Angular Correlations in SL-light

The deviations from isotropic emission in SL can be resolved in Fig.5.7. Angular dependence in the intensity of SL would be characterized by a nonzero value of the correlation,

$$\Delta Q_{AB} = \frac{1}{Q_A Q_B} \langle (Q_A(i) - \bar{Q}_A)(Q_B(i) - \bar{Q}_B) \rangle_i \quad (5.1)$$

as a function of the angle θ_{AB} formed by the PMTs A and B . In Eq.(5.1), $Q_A(i)$ is the total charge recorded in PMT A on the i th flash, \bar{Q}_A is the running average of $Q_A(i)$, and $\langle \rangle_i$ is an average over i .

Fig.5.7(a) shows that the mean value of each PMT obtained for a 1 minute run. It is found that the ADC mean values are not completely at the same level. To calibrate the mean ADC values for each PMTs is a meaningful work in HBT study although it is hard to calibrate a moving bubble. By substituting the ADC value to Eq.5.1, we can see the angular

correlation of light intensity between two PMTs at subtending angle of 15° , 30° and 45° . Moreover, the dipole component from the emission light region is also found as the same result as Robert Hiller et al. predicted in 1996 [14]. The relative ratio of dipole moment to the isotropic one is approximately 10^{-5} in our measurement.

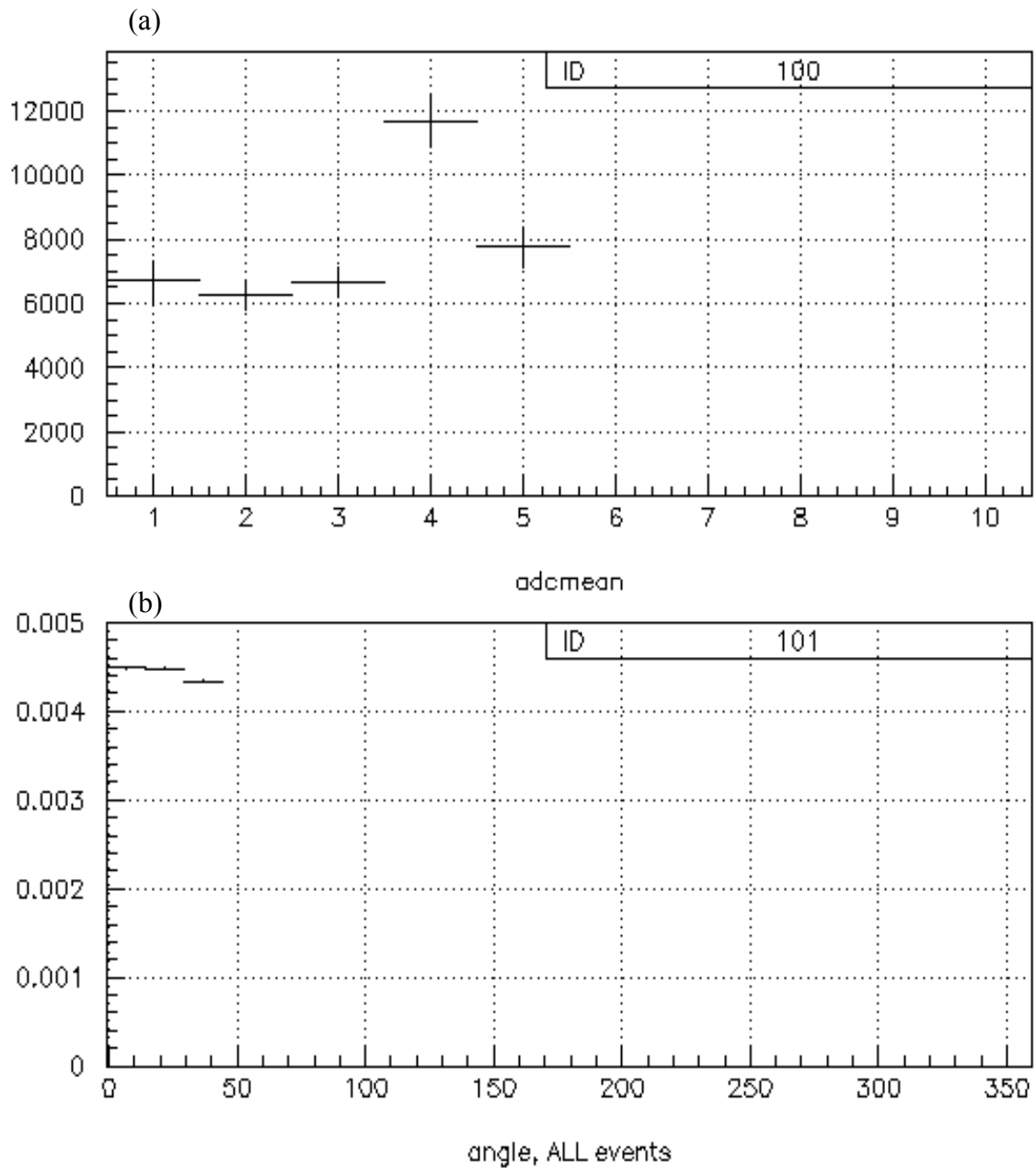


Fig.5.7 Correlation of light intensity between two PMTs subtending an angle θ_{AB} with respect to a sonoluminescing bubble.

Chapter 6 Summary and Outlook

6-1 Summary

1. Construction of the Cross-Platform DAQ System

In SL-DAQ system, we built up on two different platforms which respectively are the UltraSPARC-IIe, 400/500MHz processor-based CPU-54 single board computer (manufactured by Force Computer company) which runs Sun Solaris 2.6 for collecting ADC data and the cross-platform computer which runs Red Hat Linux 9.0 version for analyzing and monitoring the event data. We can achieve the DAQ rate at 12KHz.

2. Modification of the SL-DAQ system

In order to check if our SL-DAQ system does work, we have done 2 experiments for testing: Gamma ray source experiment and the LED test experiment. From the Gamma ray source experiment, we observed the well-known signals from Cobalt beta decay. The acquired spectrum from the event data shows that our DAQ system in addition to the NaI detectors can resolve the real physical results. By the way, it also checks our decoding program called from the off-line analyzer. The second LED testing experiment provides us some unusual event information. It's a sign to check if the trigger is in correct level.

3. Correlation Analysis

By comparing the PMT signals, it concluded that the HBT methods in determining the radius by our current detectors did not offer any meaningful information about source size due to the missing of wavelength filter and narrow angle coverage. However, up to now in IPAS, we have not the good techniques to get over the photon counting experiments. Since event by event is equivalent to the pulse by pulse analysis. The intensity of angular correlation analysis can be done at subtending angle of 15° . From the acquired data we found that the dipole component in SL emission light can be detected in this measurement.

6-2 Outlook

In our study, although we can just obtain some dipole information by comparing the light intensity of SL emission light, we cannot obtain the real radius of the emission region inside the bubble. In order to improve the HBT method, a more sensitive detector (PMT) is possibly needed to solve this problem when a $\sim 10\text{nm}$ band width filter is added on it. In the future, by repeating the measurement again with photon counting (HBT) method, it's able to measure the real radius of the bubble during the light emission occurs. If it is so, therefore, it can offer us more information about why the bubble emits light or the possibility of the fusion inside the "Sono-bubble".

References

- [1] J. Acoust. Soc. Am Suppl. 1 87, S141 (1990).
- [2] R. P. Taleyarkhan et al., Science 295, 1868 (2002).
- [3] H. Metcalf, Science 279, 1322 (1998)
- [4] A commercial sonoluminescence apparatus can be seen at www.teachspin.com.
- [5] David H. Boal, Intensity interferometry in subatomic physics, Rev. of Modern Phys. Vol. 62, No. 3, 553~602 (July 1990).
- [6] S. Trentalange and S. U. Pandey, Journal of the Acoustical Society of America, Vol 99, 2439~2441. (1996)
- [7] Robert Hiller, Seth J. Putterman and Bradley P. Barber, Physical Review E, Vol 54, Number 3, 2205-2208 (1996).
- [8] The Quad Gate/Delay Generator refer to : www.phillipsscientific.com/preview/794pre.htm
- [9] Robert Hiller, Seth J. Putterman and Bradley P. Barber, Phys. Rev. Lett. 69, 1182-1184 (1992)
- [10] A commercial ADC module can be seen at www.caen.it/nuclear/.
- [11] A commercial single board computer can be seen at www.forcecomputers.com.
- [12] R. Ball, Software for UNIX-Based Data Acquisition User's Guide.
- [13] Documentation of PAW can be seen at <http://wwwasd.web.cern.ch/wwwasd/paw/>
- [14] W. Richard Stevens, Advanced Programming in the UNIX Environment.

- [15] R. Ball, Software for UNIX-Based Data Acquisition Technical Guide
- [16] K. S. Budil et al., Phys.Rev.Lett.76,4536(1996).
- [17] M. Kornfeld and L. Suvorov, J. Appl. Phys.15,495(1994)
- [18] R. Fruhwirth, M. Regler, R. K. Bock, H. Grote, D. Notz, Data Analysis Techniques for High-Energy Physics.
- [19] William R. Leo, Techniques for Nuclear and Particle Physics Experiments.
- [20] Henry Smeat, John R. Albright, Introduction to Atomic And Nuclear Physics.

Appendix A

Experimental Verification of Rutherford's Nuclear Theory of the Atom

Rutherford, in 1911, proposed a nuclear theory for the structure of the atom. He was led to this theory by the results of an experiment by Geiger and Marsden on the scattering of alpha particles by matter. He assumed that both the nucleus and the alpha particle behaved as point charges, that Coulomb's law was valid for such small distances, and that Newtonian mechanics was applicable. By considering a nucleus of charge Ze stationary at point C and an alpha particle of mass M and charge Q approaching it along the line AB, as shown in Fig. There will be a force of repulsion between the two charges given by Coulomb's law,

$$F = k Ze Q/r^2 \quad (1)$$

Because of this force of repulsion, the alpha particle will be deflected from its original direction and will move in a hyperbolic path with the nucleus at the focus on the convex side of this branch of the hyperbola. The angle of deflection θ is given by

$$\cot \frac{\theta}{2} = \frac{MV^2}{kZeQ} p \quad (2)$$

where p is the distance from the nucleus at C to the original line of motion AB. In the actual experiments on the scattering of alpha particles a large number of particles were directed against a thin metallic foil. It is therefore necessary to calculate the number of particles scattered through a given angle θ or, what amounts to the same thing, determine the probability that an alpha particle will be deflected through this angle θ .

Appendix B

B-2 Installation of UNIDAQ on Red Hat Linux 9.0 (kernel-2.3.20-8)

(* : means the comments)

reference website : <http://www.phys.sinica.edu.tw/~spring8/UNIDAQ.htm>

1. Update the Linux kernel version: 2.4.20-8
2. Install the GCC compiler version: 3.2.2-5
3. Download the old UNIDAQ v2.3 package (unidaq2.31s5.tar.gz) from the High

Energy Accelerator Research Organization (KEK) website:

<http://www-online.kek.jp/~online/Unidaq/unidaq2.3/>

4. Unpackaged it and modified the configuration file in the NOVA directory:

Add the parameter “ #define Linux ”

5. Change the login shell as “tcsh”.

6. Edit the file (example in nuclear-2): `~/.tcshrc`

`source /data/v2.3/setup.csh`

`set=(($path /data/v2.3/user_UNIDAQ/bin)`

7. Edit the file (example in nuclear-2) : `install.csh`

`TOPUNIDAQ = [/data/v2.3]`

`Install for Linux = 1`

`CERN_ROOT = /usr/local/cern/pro` (*the PAW executable link)

`MURMUR_DIR = /usr/murmur/v1.1`

`MURMUR_SERVER_ADD = 192.168.102.238` (*the nuclear-2 IP)

8. Execute the shell script : `./install.csh`

9. Execute the shell script : `./setup.csh`

10. Enter the NOVA directory : `cd NOVA/include/`

11. Modify the header file : `include/config.h`

```
#define Linux
```

12. Modify the file : NOVA/Makefile

```
(*line 24) OS_NAME = Linux
```

```
(*line 51) make OS_NAME=$(OS_NAME) CC=$(CC) FF=$(FF)
```

```
all.$(OS_NAME));
```

13. Modify the file : /NOVA/doc/Makefile

```
(*line 38) all.Linux:
```

14. ./su (*change to superuser)

15. Install UNIDAQ : make install

16. Change directory to : /data/v2.3/lib (*example in nuclear-2)

```
make
```

17. Change directory to : /data/v2.3/recorder (*example in nuclear-2)

```
make
```

18. Change directory to : /data/v2.3/receiver (*example in nuclear-2)

```
make
```

19. Change directory to : /data/v2.3/dataview (*example in nuclear-2)

```
make
```

20. Install UNIwish :

Modify the “Makefile” under the directory : /data/v2.3/user_UNIDAQ/UNIwish

```
(*line 9) TOPTCLTK = /usr/local
```

21. Install TCL/TK version of tcl70.b3 and tk3.3b3

22. Change the directory to /data/v2.3/src/control and “make” it.

23. Change the directory to /data/v2.3/src/etc and “make” it.

24. Change the directory to /data/v2.3/include and “make” it.

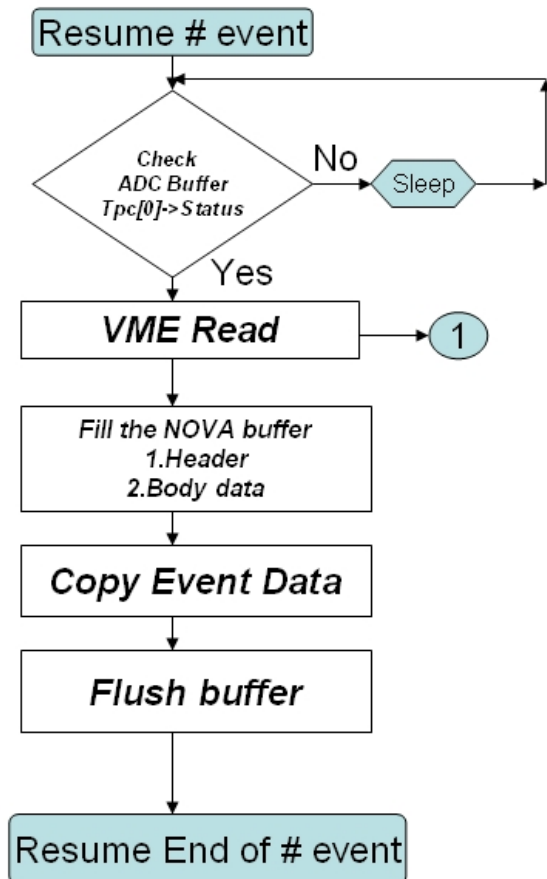
B-1 SL-DAQ operation guide

```
ssh -Xl daq nuclear-3 /* nuclear-3 IP : 192.168.103.249 */
nuclear-3 : daq /*telnet Lep5401 machine */
Lep5401 : cd /export/home/daq/user_UNIDAQ/collector-v965
           novad /tmp/nova & (start once only)
           novand > /dev/null & (start once only)
           collector-tpc-sl -n c1
nuclear-3 :
           cd /data/sl-data
           recorder -n r1 -m atx /* start recorder on nuclear-3 */
Lep5401 : ask c1 begin /*initialize the DAQ */
           ask c1 resume /* Data transfer begin */
           ask c1 end /*DAQ stop */
nuclear-3: cd /data/v2.3/user_UNIDAQ/collector-tpc/bin.sl
           ./evdis-off.sl -p /data/sl-data > tmp.log
           cd /data/sl-data
           paw &
           paw> h/file 1 fadc.hst
           paw>h/pl 1
           paw>h/pl 2
           ... /* display the histogram for the required channel */
```

Appendix C

Sub-routine: `user_event.c`

Source code: <http://www.phys.sinica.edu.tw/~spring8/SL>



Sub-routine : VME_READ

