

# **Creating an EPICS Based Test Stand Development System for a BPM Digitizer of the Linac Coherent Light Source**

Farah Memon  
Office of Science, Science Undergraduate Laboratory Internship Program

San Jose State University

SLAC National Accelerator Laboratory  
Menlo Park, California

August 13, 2010

Prepared in partial fulfillment of the requirement of the Office of Science, Department of Energy's Science Undergraduate Laboratory Internship under the direction of Shantha Condamoor in the Controls Department division at SLAC National Accelerator Center.

Participant:

\_\_\_\_\_  
Signature

Research Advisor:

\_\_\_\_\_  
Signature

## TABLE OF CONTENTS

ABSTRACT	iii
INTRODUCTION	1
MATERIALS AND METHOD	4
GRAPHICAL USER INTERFACE AND PRESENTATION OF RESULTS	9
DISCUSSION AND CONCLUSION	10
ACKNOWLEDGEMENTS	11
REFERENCES	12
FIGURES	13

## ABSTRACT

Creating an Improved Test Stand Development System for a BPM Digitizer of the Linac Coherent Light Source. Farah Memon (San Jose State University, San Jose CA 95192)  
SHANTHA COONDAMOOR (Controls Department, Stanford Linear Accelerator center, Menlo Park, CA 94025)

The Linac Coherent Light Source (LCLS) is required to deliver a high quality electron beam for producing coherent X-rays. As a result, high resolution beam position monitoring is required. The Beam Position Monitor (BPM) digitizer acquires analog signals from the beam line and digitizes them to obtain beam position data. Although Matlab is currently being used to test the BPM digitizer's functions and capability, the Controls Department at SLAC prefers to use Experimental Physics and Industrial Control Systems (EPICS). This paper discusses the transition of providing similar as well as enhanced functionalities, than those offered by Matlab, to test the digitizer. Altogether, the improved test stand development system can perform mathematical and statistical calculations with the waveform signals acquired from the digitizer and compute the fast Fourier transform (FFT) of the signals. Finally, logging of meaningful data into files has been added.

## INTRODUCTION

The Linac Coherent Light Source (LCLS) is a novel invention of SLAC National Accelerator Laboratory which aims to create movies of molecular activities by capturing images using powerful and high frequency x-ray pulses. Since the x-rays have a wavelength which is smaller than the dimensions of chemical molecules, 1.5 Angstroms to be exact, they are able to successfully provide insight regarding the structure of molecules and the properties of chemical reactions [1]. Not only are the x-ray pulses short in wavelength, giving them substantial energy, but they are also flashed once every 100 femtoseconds with energy ranging from 540eV to 9.0keV [2]. These features of the x-ray pulses make them an ideal source for offering accurate depictions about the movement, structure, and the arrangement of particles at the atomic level. Due to these advantageous properties, the LCLS can be used to provide monumental advances in the fields of medicine, chemistry, and materials science.

One device used to monitor the LCLS instrument is the Beam Position Monitor (BPM) digitizer. The Controls Department utilizes the BPM digitizer to scrutinize the trajectory of the electron beam in the LCLS due to the digitizer's capability of providing a method to derive the location and the length of the electron bunches in the undulators. Four unique analog signals are captured, one from each side of the beam as shown in Figure 1, and converted to low intermediate frequency (IF) signals [3]. These radio frequency signals are transformed into digital signals using analog-to-digital converters (ADCs). Data is acquired by the ADCs at each rising edge of the clock signal, which may be either external with a frequency of 119MHz or internal with a frequency of 125MHz. Once digitized, each analog signals is loaded into one of the four 16K by 18 bit buffers, depending on its input channel. The analog signals are comprised of multiple samples where each sample is 18 bits long, with the first 16 bits enclosing the raw

data. The input clock signals of the buffers (Clk1, Clk2, Clk3, and Clk4), generated by the four individual ADCs, trigger the release of the samples into the buffers. Apart from the raw sample data and the four clock signals, the buffers also need the acquisition gate signal, which defines the number of samples obtained in each cycle. The acquisition gate signal is generated by any of the two methods: internal triggering or external triggering. In the former case, the internal trigger generates the acquisition gate signal, while in the latter case, one of the two external triggers (Trig1 or Trig2) is responsible. This process is clearly demonstrated in Figure 2. Since the buffers have 16K locations, a maximum of 16384 samples can be passed to each buffer in one cycle [4]. In the case of this research project, for the purpose of limiting the scope of testing, 4096 samples can be passed altogether, corresponding to 1024 samples per channel. Once transferred to the buffers, the analog signals are digitized and in a format that allows mathematical calculations and digital processing to be computed.

The main components used to control the devices located on the periphery of the LCLS, such as the BPM Digitizer, are the Input and Output Controllers (IOCs) and operator interfaces (OPIs), which are both are connected via a local area network (LAN). While the IOCs are servers that are used to communicate to the device, perform mathematical calculations on its data, and monitor its internal registers, the OPIs are client computers utilized to view the device's information gathered from the IOC in a user-friendly way. As depicted in Figure 3, most of the IOC is devoted to running the Experimental Physics and Industrial Control System (EPICS) Core and the Application Database. The EPICS Core is responsible for transferring data between the IOC and the OPI while the Application Database is essential for keeping an accurate track of the captured data in an organized database fashion [5]. In the case of the BPM digitizer, the IOC is a VME 64x crate with a Motorola microprocessor VME6100 that uses the Real-Time Executive

for Multiprocessor Systems (RTEMS) operating system (OS) while the OPI is a Dell computer with Linux as its OS.

The Application Database is a collection of records, or macros, necessary to obtain data from the device or perform operations on data which has already been collected. Records contain a list of subroutines that are activated each time the record is required to process and obtain useful information. There exist multiple fields in the records, called Process Variables (PVs). While some fields are inputs and defined by the user, other fields are outputs of the record [5]. On the other hand, the EPICS Core consists of Channel Access which is fundamentally a path of communication between the IOC, also known as the server, and the OPI, also called the client. Through its path of communication, Channel Access successfully provides access to the contents of PVs, located in the IOC, to the OPI, thus permitting the user to observe the contents of the record fields [6].

Currently, Matlab is being utilized to test the BPM digitizer. The Matlab scripts provide results to the computer through Matlab Guide. However, it is preferable to use EPICS, as an integrated test tool as well, due to its reliability, community support, and tractability. Many national laboratories are in the process of using EPICS, which is a set of software libraries and applications, for controlling large scientific instruments. This paper thoroughly explains the process of providing the required functionalities for testing the digitizer using EPICS.

## **MATERIALS AND METHOD**

### *Equipments and Project Problem*

In order to provide an effective test stand development system that accurately tests the BPM digitizer before it goes into the production stage, several equipments were utilized. These include the following:

1. Cisco Systems Catalyst 3750 Series Switch
2. Digi Port Server TS 16 serial port
3. Power-One Hybricon VME 64x crate
4. HP Signal Generator (Model 8648C)
5. Stanford Research Systems Digital Delay Generator (Model DG645).
6. The VME 64100 microcontroller in the VME 64x crate's Slot 1.
7. BPM Digitizer (Model 114-045-1) in the VME 64x crate's Slot 4.

The primary goal of the digitizer, as seen earlier, is to provide a digital representation of the four different analog signals acquired from the electron's trajectory in the LCLS. The driver for the BPM digitizer was derived from the driver that is part of the production BPM Application Code.

The Matlab scripts, which currently test the digitizer, compute the DC offset, minimum, maximum, peak-to-peak, and the fast Fourier transform (FFT) of the four individual signals. As Figure 4 illustrates, the results of all the mathematical operations are clearly displayed on Matlab Guide graphical user interface (GUI). Although the Matlab scripts offer a suitable test stand configuration system, similar and even advanced functionalities need to be provided through EPICS to accurately examine the digitizer. Currently the EPICS driver code only permits the

user to precisely view the four waveform signals as one whole signal on the EDM panel, which is the GUI for EPICS.

### *Configuration and Setup of the Devices*

The first step to develop the test stand development system using EPICS is to configure the IOC properly with the OPI. To accomplish this task, the VME 64x crate containing the microprocessor and the digitizer was connected to the Linux computer over a local area network (LAN) using a Cisco Systems Catalyst 3750 Series Switch. A Digi Port Server TS 16 serial port is also used for local or remote debugging of the IOC, as displayed in Figure 5. The HP Signal Generator, shown in Figure 6, provides the input waveform signals, 40MHz at 0dB on first two channels, which are close to the actual analog signals which the digitizer may acquire. To obtain and process the data, the digitizer needs to be clocked and triggered, both of which can be either external or internal. For testing purposes, the internal clock with a frequency of 125MHz is employed. A Digital Delay Generator, illustrated in Figure 7 is used to externally trigger the signal at a rate of 10Hz although the input waveform signals can also be triggered internally through software.

After configuring the hardware devices, the IOC application for the BPM digitizer was built using EPICS environment, tools, and packages. Once the IOC application is built using the make command in Linux, the following directories are created in the IOC Application folder: bin, db, dbd, and, lib. The application is built against RTEMS 4.9.1 RTOS and the output object file is placed in the bin directory as the result of the 'make' command. The dbd directory consists of the definitions of all the records the user can generate while the db directory contains the list of all the records which have been created in the specific IOC application. Additionally, the

configure directory sets up the environment variables needed to process the IOC. Finally, there is a startup script for setting up the IOC appropriately after booting.

The next step consisted of running the latest driver code. A subset of R5-3-0 of BPM Application Code was used for this project. The simulation related portions of the driver code were disabled as this had dependencies on other code which were not needed for this project. All the test application software was built on top of this driver code to test the basic functionalities on the IOC. The rudimentary functionalities of the driver code allow the four waveform signals to be properly obtained from the digitizer and be accurately displayed on the EDM panel as one cumulative signal. Additionally, the driver code provides some low-level functions for exposing specific parameters of the digitizer, such as serial, firmware, and crate number, which the test application software presented in the EPICS GUI. The iocConsole is another tool used to control the IOC by providing a unique pathway of communication between the IOC and the Linux computer. Due to its ability to directly communicate, via serial communication, with the IOC, it is used to issue commands and debug. One of the useful commands that is issued from the iocConsole software, includes `bsp_reset()`, which allows the IOC to reset during during development.

### *Implementing the Advanced Functionalities*

The test application extended the driver to support new features. These advanced operations consist of illustration of all four input signals as independent waveforms, mathematical calculations, fast Fourier transforms, and logging of data.

### Illustration of the Input Signals

In order to separate the combined input signal into four different waveforms, the `genSub` record was used. The `genSub` record, when processed, invokes a subroutine in the source code,

which is essentially a C function. This specific record can access up to 21 PVs as inputs, whether they are in scalar or array format, and perform complex calculations using the input parameters. The genSub record is passed as a pointer of a structure (\*psub) to the subroutine and the different fields of the genSub record, such as the input and the output fields, are all elements of the structure and accessed using the pointer [7]. For the separation of the waveform signal, a genSub record called *showWaveform* was created. This record accepted the combined signal as well as the four distinct signals to the four input channels. When invoked, it divided the combined signal, holding 4096 elements, into four waveforms, each comprising 1024 samples. After executing the division, the subroutine uses a database access routine, *dbPutField*, to change the contents of the four input fields that correspond to the waveforms of the four channels. These four waveform signals are displayed in the original EDM panel.

### Waveform Analysis

To perform the calculations on the four waveform signals, the waveProc module, which is a set of functions that interfaces with the existing source code, was used. This specific module is available from EPICS and serves as a tool that interacts with waveform records and performs mathematical and statistical operations on them [8]. After incorporating the waveProc module, four instances of the waveAnl record were created. Each record pertains to the waveform on each channel. The waveAnl record accepts the raw waveform as its input, performs calculations, and outputs the answers of the calculations in different PVs which are presented in an EDM panel. To effectively expose all the data, separate EDM screens for each channel and the combined waveform were created that clearly displays all the meaningful and useful data for that particular channel in a well structured manner. These five new EDM screens, called the Waveform Analysis Record Panels, are linked to the original EDM screen via a button.

### FFT Processing

The FFT is computed on the waveform signals of the four channels using Matlab *labCA*. This test program allows a Matlab client to interface via Channel Access to the IOC and have access to all the published PVs. As a result, Matlab script was written that captured the digitized waveform signals, stored them as variables, performed FFT, computed the power spectrum, and calculated the fundamental frequency. The power spectrum and the fundamental frequencies of the four channels were linked to separate PVs, which were exposed on the FFT EDM panel. This specific panel is linked to the main EDM panel via the 'FFT Button'.

### Logging of Data

The application added a new feature to store the results of a test along with the data collected from the waveform signals and the digitizer into a log file. The log files are created on demand and stored with file names that have time stamp of the test time. All of the log files are saved in the nfs directory specified by the following path:

`/nfs/slac/g/lcls/epics/ioc/data/ioc-b34-cd16/output`

Six different types of log files are created: four log files corresponding to the four input waveforms, one log file for the combined waveform, and one generic log file for hardware-specific information, i.e, serial, firmware, and hardware revision number. To create the five log files that record all the mathematical data relating to the waveform signals, a function was implemented in the waveAnl record. This function was called upon the click of a button in each wave Analysis Record EDM Panel. Moreover, to produce the generic log file, a subroutine record named *genericLog* was created. This subroutine captured the hardware specific parameters and when processed upon the click of a button on the main EDM panel, created a log file that adequately and neatly stored all the information relevant to the digitizer.

## **GRAPHICAL USER INTERFACE AND PRESENTATION OF RESULTS**

After compiling the code and linking all the EDM panels, the test development system prototype was built. As portrayed in Figure 8, the main EDM panel displays the four waveforms, each corresponding to a different channel. The graphs possess the zoom-in capability such that the user may be able to concentrate and carefully observe a particular area of a waveform. The combined signal can be viewed by clicking upon the button, 'VMEADC03-All'. This button brings up an EDM panel that demonstrates a graph of the combined waveform signal, as illustrated in Figure 9. Besides the four graphs, five buttons are on the main panel that link to the Wave Analysis Record EDM panels. Four of the five buttons correspond to panels of the four individual waveforms ('VMEADC03-1', 'VMEADC03-2', 'VMEADC03-3', 'VMEADC03-4') and one of the buttons links to the wave analysis record panel of the combined signal ('VMEADC03').

Each of the Waveform Analysis Record panel exhibits important parameters of the waveAnl record. Displayed in Figure 10, these include the maximum (MAX), minimum (MIN), peak-to-peak (PEAK), variance (VAR), standard deviation (STDEV), and mean (MEAN). The maximum is the maximum sample point in the waveform signal, the minimum is the minimum sample point in the signal, and the peak-to-peak is the difference between the maximum and the minimum. The variance and the standard deviation are a measure of the spread of the sample points in the waveform. Lastly, the mean corresponds to the dc offset pertaining to the input signal. Apart from these mathematical values, the panel depicts a more diminutive version of the graph. Most importantly, the panel has a button called 'Process'. Corresponding to a PV of the waveAnl record, upon clicking this button, the record processes, and the values on the panel are updated. Another significant button includes 'Logger'; this button triggers the creation of a log

file that stores the mathematical values of the signal in a method displayed in Figure 11. The name of the log file is in the following format: *chNlog\_MM-DD-YY\_HH:MM:SS.dat*. The letter N corresponds to the channel number. It spans from 0 to 4 where 0 indicates the combined signal and 1-4 are relevant to the signals of Channels 1 to Channel 4. On the main EDM panel exists a button labeled ‘Generic Log’, used to activate the *genericLog* subroutine. Upon clicking this button, the generic log file is created with the name *generic.dat*. This file, as displayed in Figure 12, stores hardware-specific parameters of the digitizer.

Lastly, the power spectrum and the fundamental frequencies of the four waveforms are displayed on the FFT EDM panel, as shown in Figure 13. This specific EDM panel is linked to the main EDM panel via the ‘FFT’ button. In order to dynamically view the power spectrum, the Matlab script needs to be running in the background and can be stopped by pressing the ‘Stop FFT’ button on the FFT EDM panel.

## **DISCUSSION AND CONCLUSION**

Currently, the improved test stand development system is running effectively to monitor the BPM digitizer. It can divide the combined waveform acquired from the device into four separate waveforms, with one corresponding to each channel. Additionally, mathematical and statistical values relevant to the waveforms are computed using the *waveAnl* record from the *waveProc* module. The FFT and the logging functionalities have also been incorporated.

To further improve the test stand development system, a few additional functionalities need to be inserted. These include adding buttons on the main EDM panel that open up PDF documents discussing ways to configure the digitizer. This feature will assist the user in setting up the test stand system if he or she is not familiar with the steps. Furthermore, the capability of changing engineering units needs to be provided; the user should be able to change between the

units of samples and volts with a click of a button. Changing units should change the scale of the graphs and the values of the mathematical operations. In conclusion, the test stand system created as a result of this project, can be used as a guide to generate improved EPICS based test stand systems for other digitizers including the Phase and Amplitude (PAD) digitizer.

### **ACKNOWLEDGEMENTS**

I would like to thank the Department of Energy (DOE) for providing a beneficial and rewarding research experience through the Summer Undergraduate Laboratory Internship (SULI) program. The research was effectively held at the Controls Department of the Stanford Linear Accelerator Center (SLAC). Most importantly, I would like to thank my mentor, Shantha Condamoor, for guiding me through my internship experience and providing constructive feedback for my project. Lastly, I am also appreciative towards Sonya Hoobler, Matt Boyes, David Earl Anderson, and Ernest Williams for their assistance during my internship program.

## REFERENCES

- [1] J. Frisch et al. *Beam Position in LCLS* [Online]. Available: <http://www.als.lbl.gov/biw08/papers-final/MOIoTIO02.pdf>
- [2] *LCLS FAQ* [Online]. Available: [https://slacportal.slac.stanford.edu/sites/lclscore\\_public/Lists/LCLS\\_FAQ/FAQ.aspx](https://slacportal.slac.stanford.edu/sites/lclscore_public/Lists/LCLS_FAQ/FAQ.aspx)
- [3] R. Lill et al. *Design and Performance of the LCLS Cavity BPM System*[Online]. Available: <http://accelconf.web.cern.ch/accelconf/p07/PAPERS/FRPMN111.PDF>
- [4] W. Ross, “SLAC BPM Digitizing Module 144-045-1 Programming Module”
- [5] (2008, March 10). *EPICS I*[Online]. Available: <https://confluence.slac.stanford.edu/download/attachments/67503315/LCLS-EPICS-Intro.pdf?version=1&modificationDate=1257093082000>
- [6] B. Dalesio. (1999). *Channel Access Concepts* [Online]. Available: [http://www.slac.stanford.edu/comp/unix/package/epics/training/documents/02\\_CA\\_Concepts.pdf](http://www.slac.stanford.edu/comp/unix/package/epics/training/documents/02_CA_Concepts.pdf)
- [7] A. Foster. (2003, March 12). *The EPICS genSub Record Reference Manual* [Online]. Available: <http://www.slac.stanford.edu/grp/ssrl/spear/epics/site/genSub/genSubManual.pdf>
- [8] E. Norum. *WaveProc* [Online]. Available: <http://www.aps.anl.gov/epics/modules/soft/waveProc/index.html>
- [9] E. A. Medvedko et al. *LCLS Stripline BPM System Commissioning* [Online]. Available: <http://trshare.triumf.ca/~pac09proc/Proceedings/papers/th6rep036.pdf>

## FIGURES

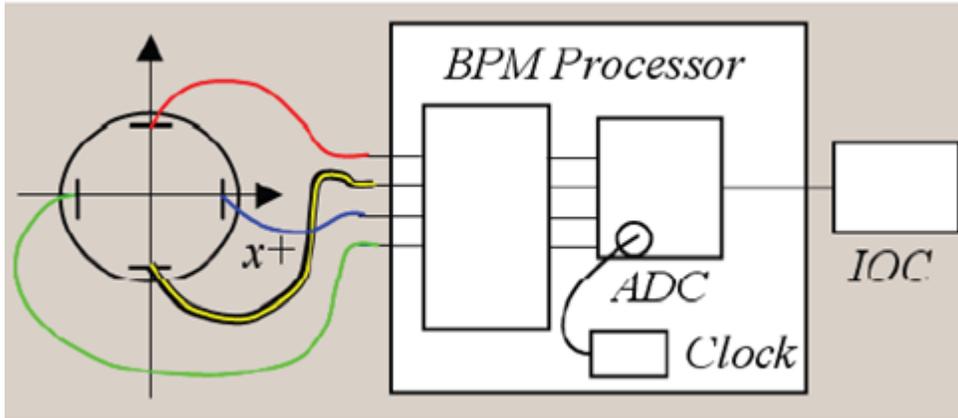


Figure 1: The BPM digitizer acquires four analog signals from the four different edges of the beam line near the undulators [9].

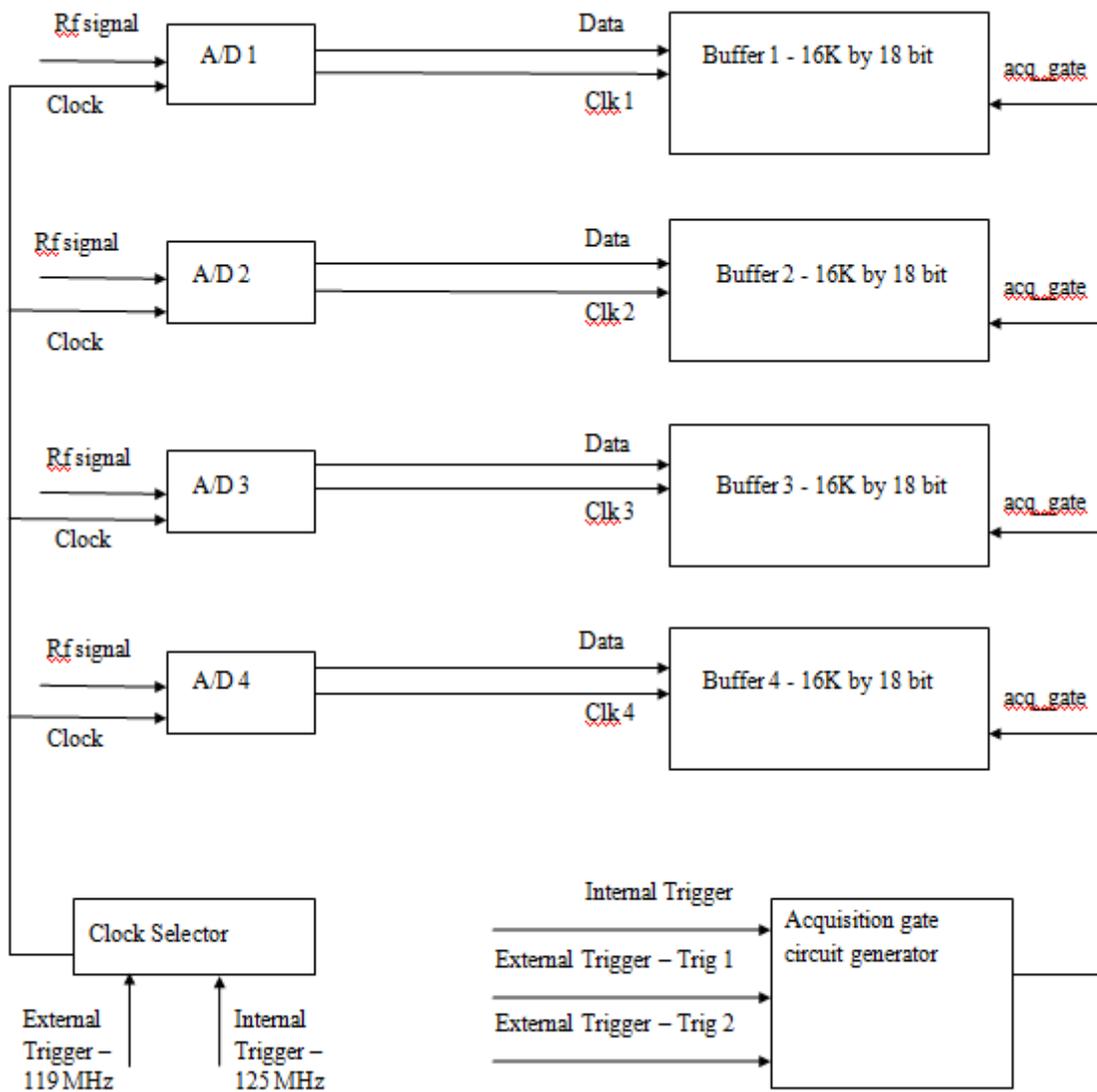


Figure 2: The process of converting the RF signals into digitized samples and storing them into the buffers.

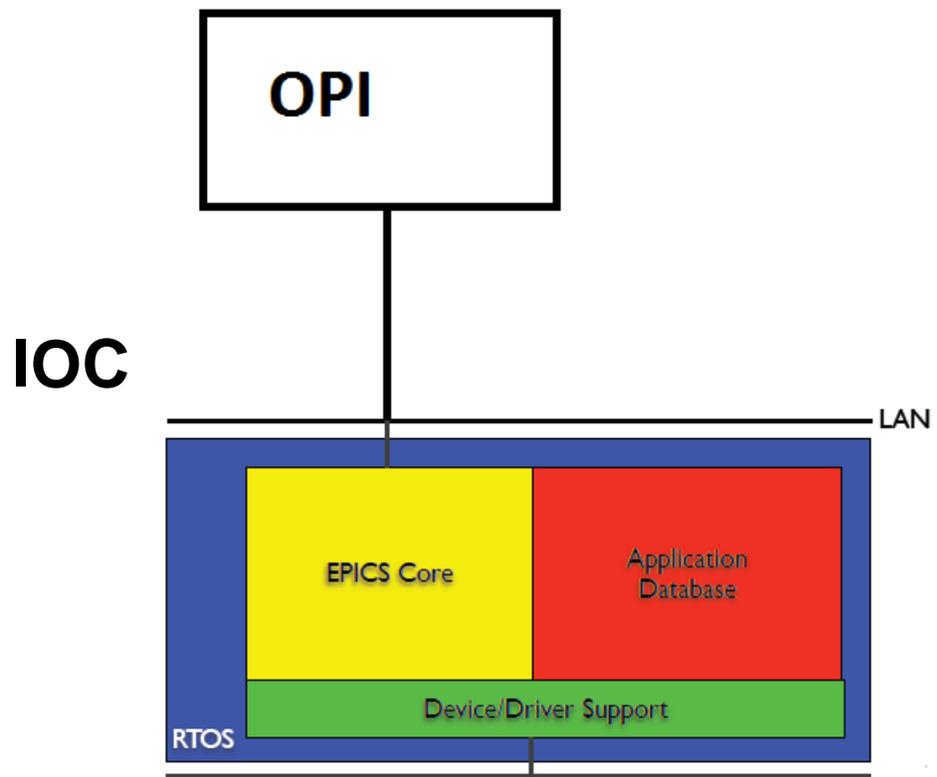


Figure 3: The IOC, consisting of the EPICS Core, the Application Database, and the Device/Driver Support is connected to the OPI through LAN.

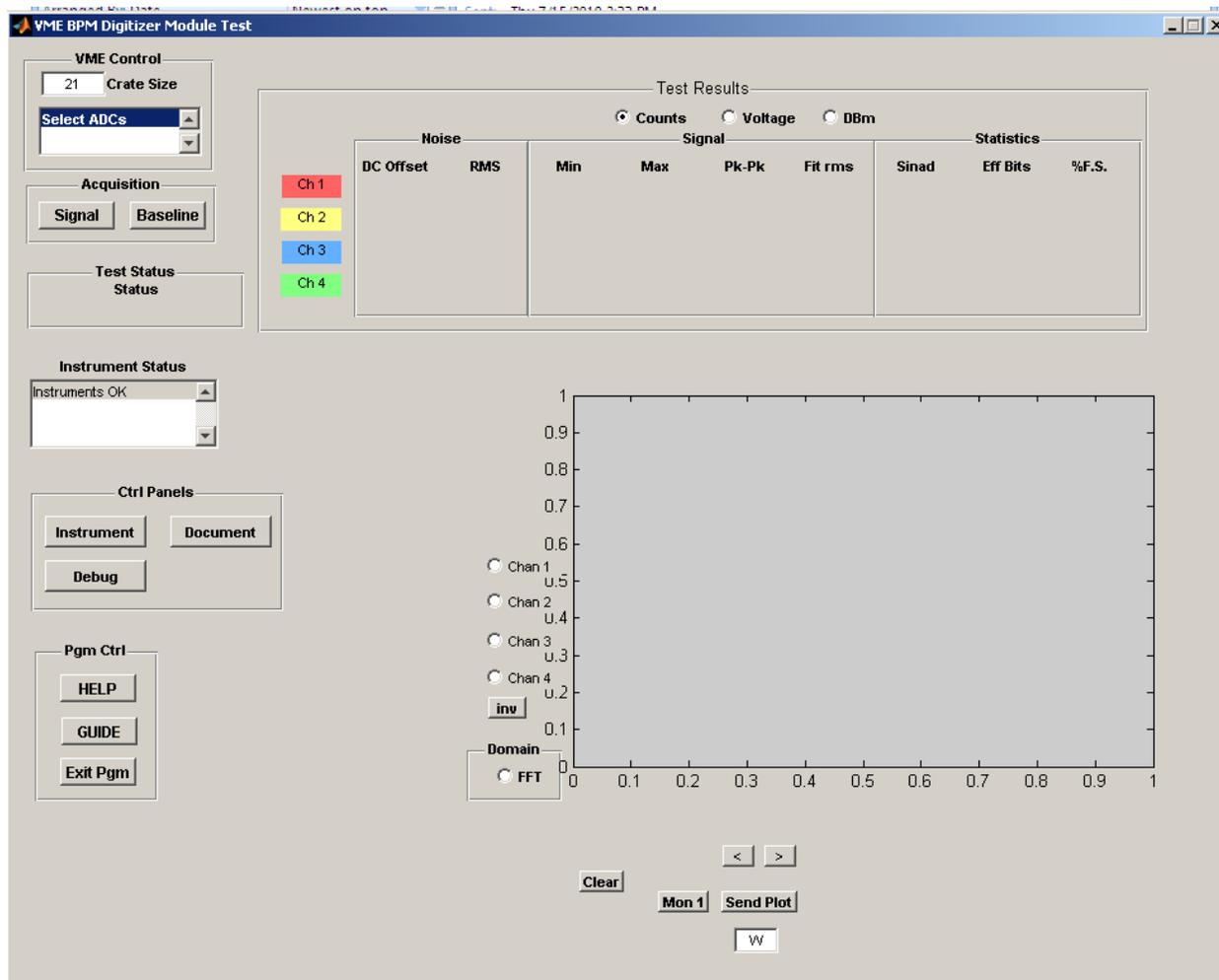


Figure 4: The existing functionalities provided by the Matlab Scripts on Matlab Guide.

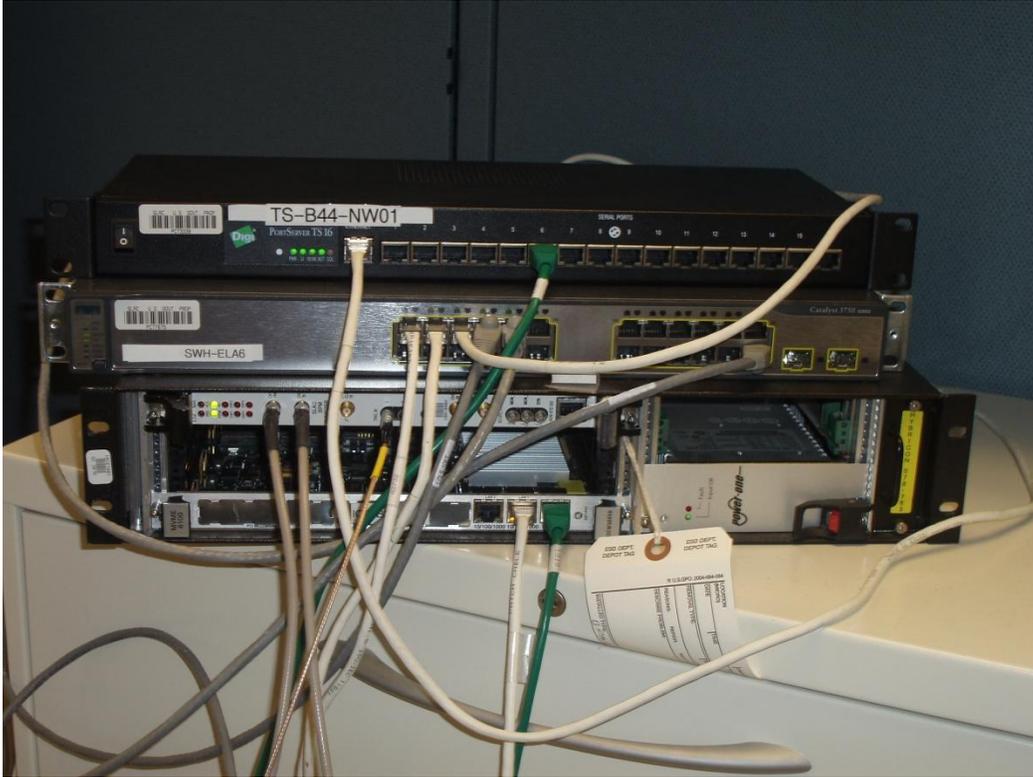


Figure 5: (From Top to Bottom) Digi Port Server TS 16 serial port, Cisco Systems Catalyst 3750 Series Switch, and Power-One Hybricon VME 64x crate.



Figure 6: An HP Signal Generator (Model 8648C)



Figure 7: Stanford Research Systems Digital Delay Generator (Model DG645).

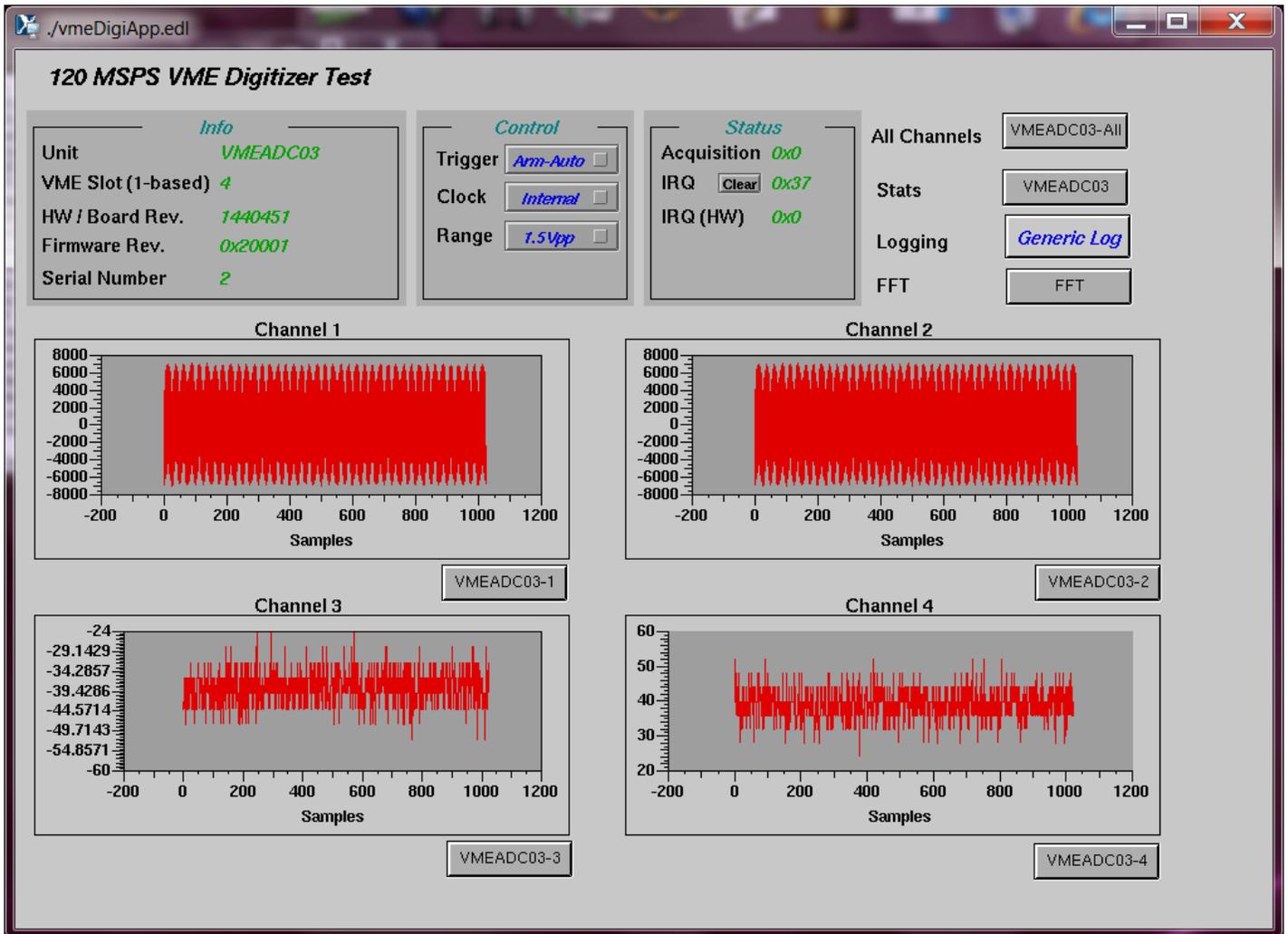


Figure 8: The main EDM panel with the four waveform signals along with the buttons ('VMEADC03', 'VMEADC03-1', 'VMEADC03-2', 'VMEADC03-3', 'VMEADC03-4') to display each Waveform Analysis Record panel. The combined waveform signal is displayed via 'VMEADC03-All' button while the generic log file is created through 'Generic Log' button.

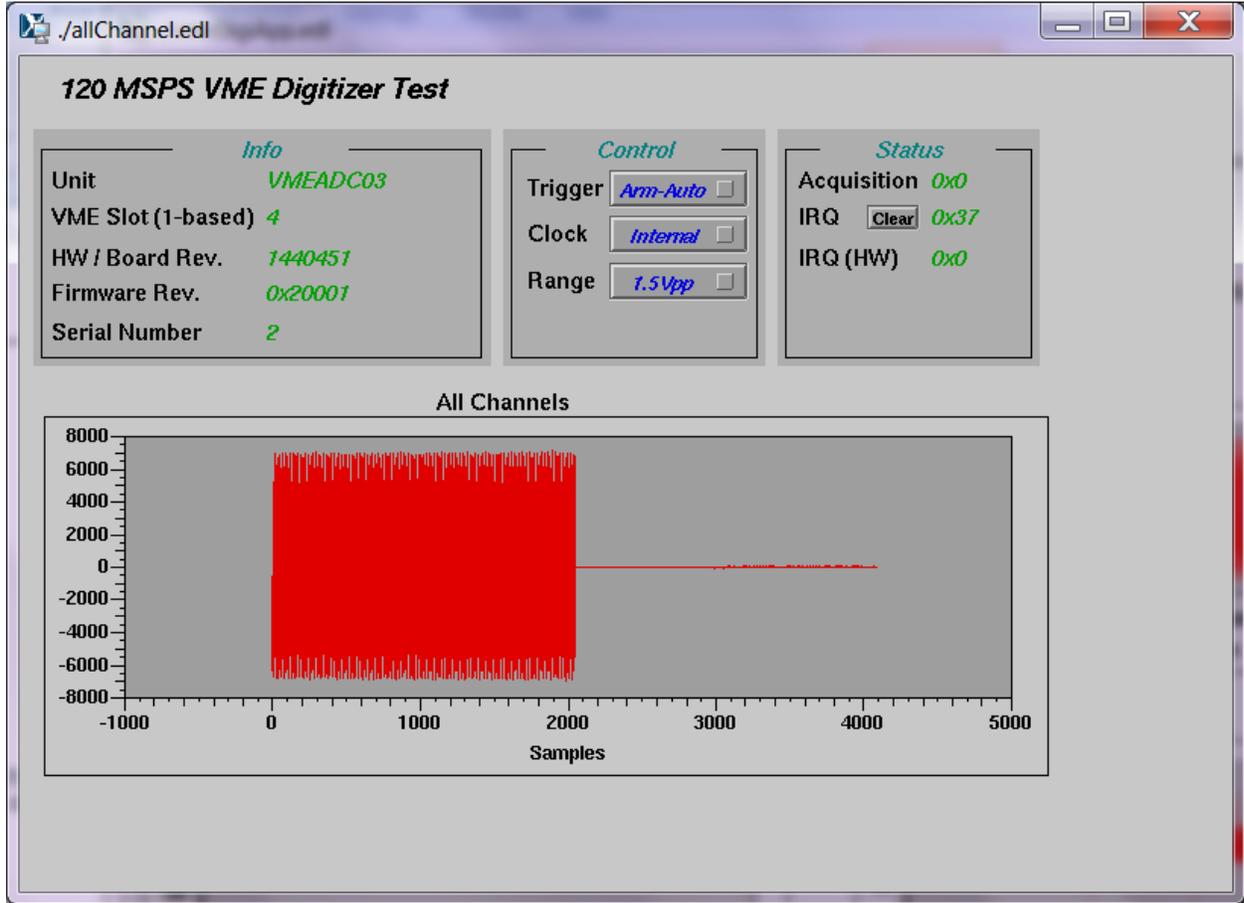


Figure 9: The EDM panel displaying the combined waveform signal. It is brought up upon the click of the 'VMEADC03-All' button on the main EDM panel.

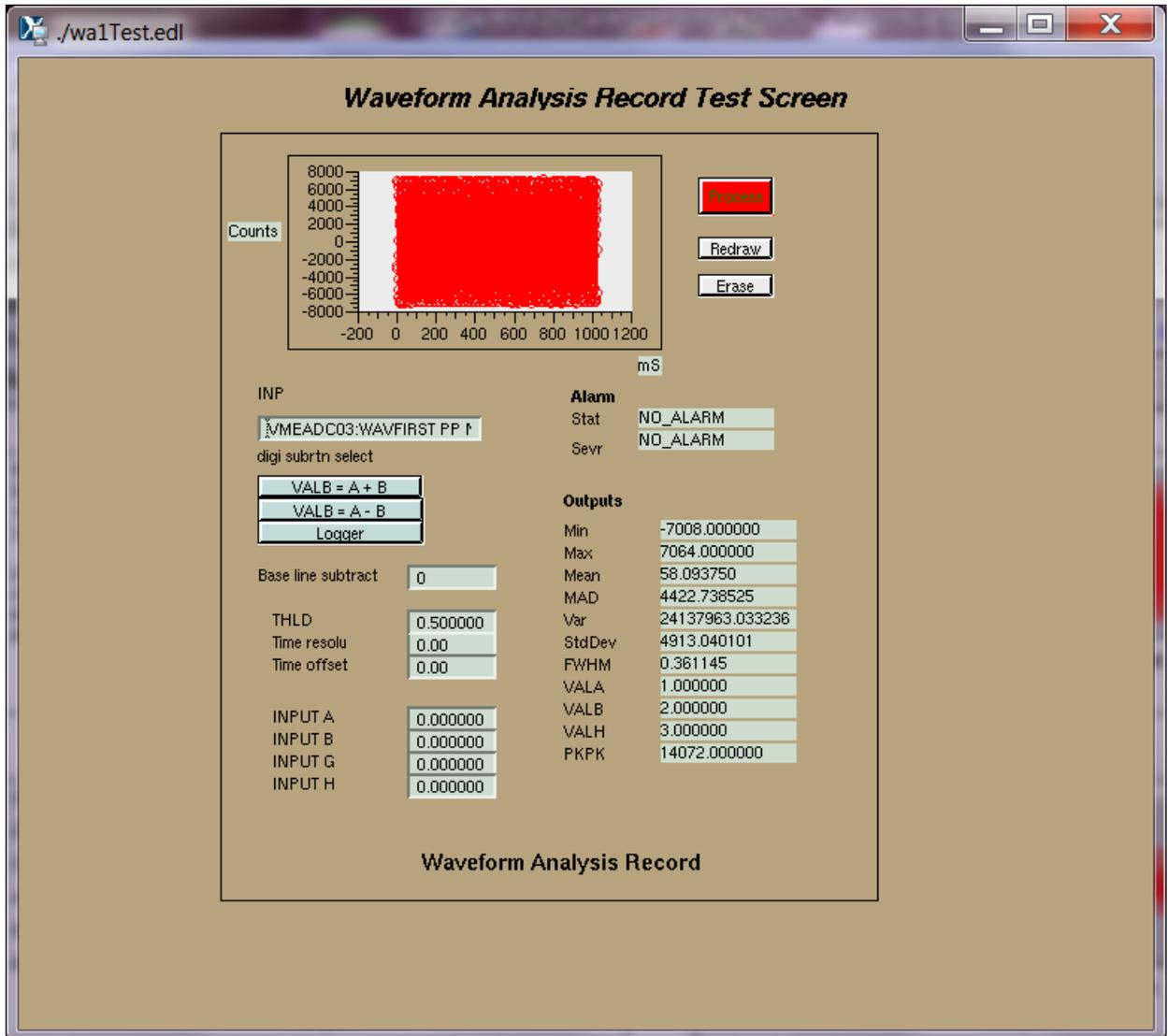


Figure 10: The Waveform Analysis Record EDM panel displays the waveform and the mathematical and statistical data relating to the waveform. All the elements on the EDM panel are process variables of the waveAnl record.

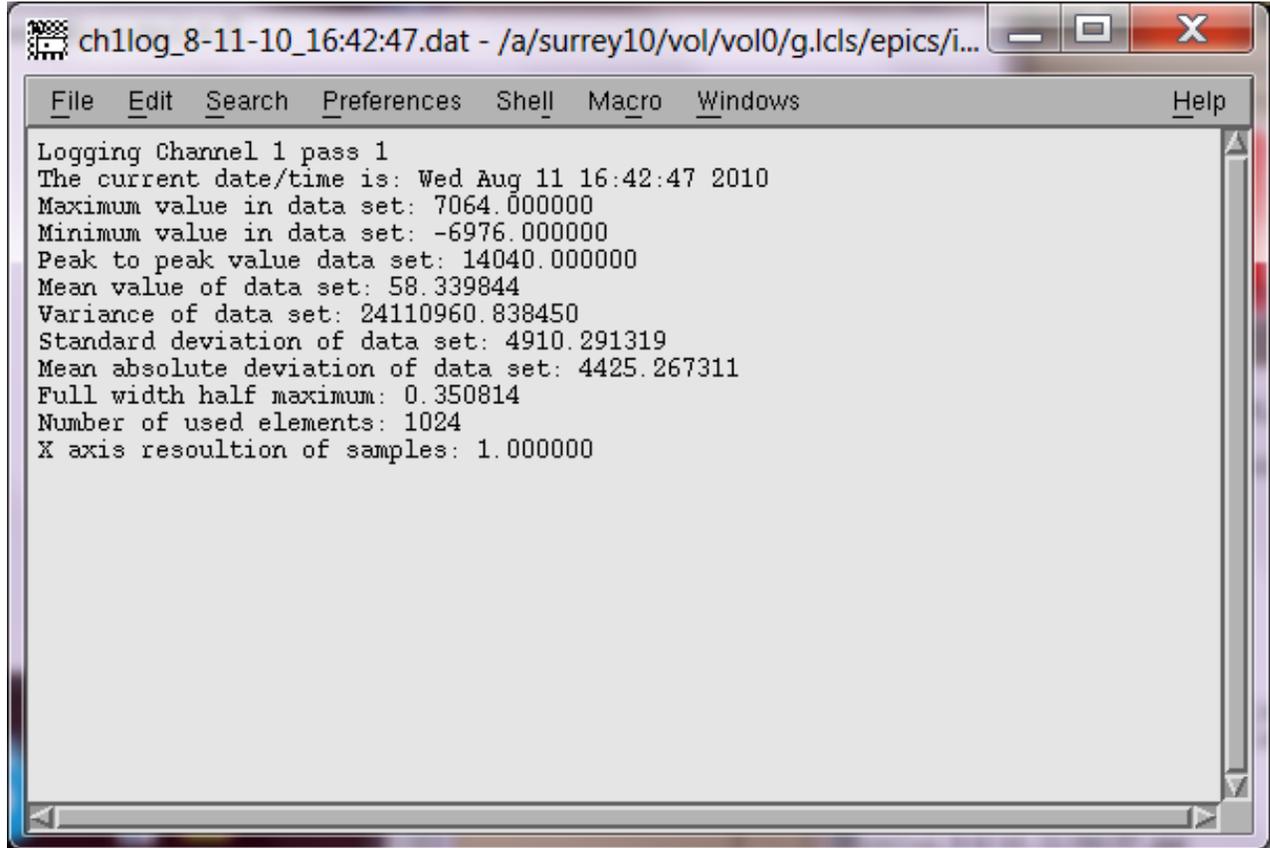


Figure 11: The channel log file consisting of the mathematical data pertaining to the waveform. It is created by pressing the “Logger” and the “Process” buttons on the Waveform Analysis Record panels.

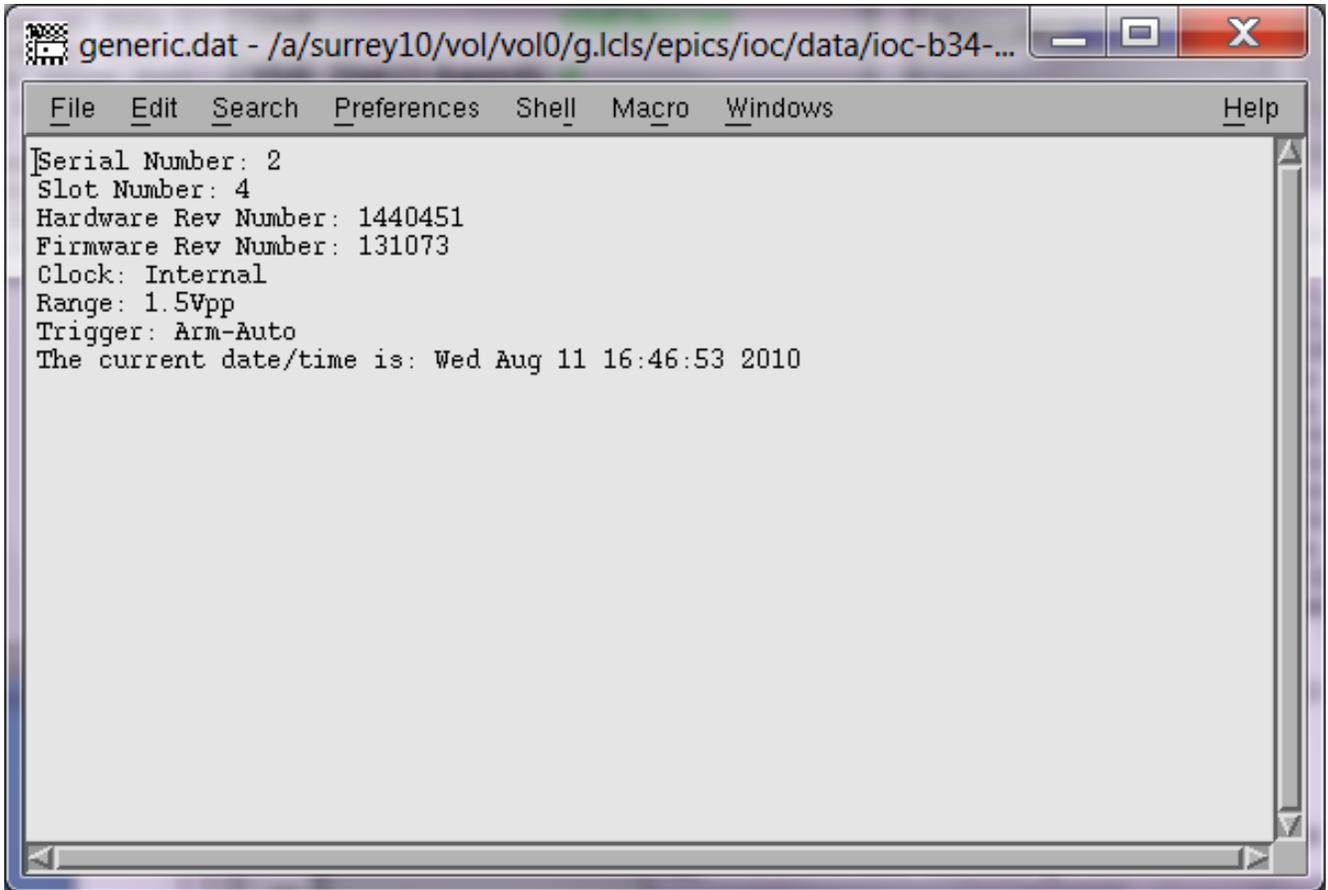


Figure 12: The generic log file carries the parameters that are specific to the digitizer. It is created upon the click of the “Generic Log” button on the main EDM panel.



Figure 13: The FFT EDM Panel displays the power spectrum and the fundamental frequencies for the four RF signals. It is invoked upon clicking the 'FFT' button on the main EDM panel.