Fermilab FERMILAB-Pub-00/244-E November 2000

# The Development of Software to Characterize the Fermilab Pixel Readout Chip, FPIX1, for the BTeV Experiment.[**]

M. A.Vargas [a,b,c,*], M. Sheaff [d], S. Vergara [a,b,c,*]

[a] *Fermi National Accelerator Laboratory, PO Box 500, Batavia, lL, 60510, USA*
[b] *Faculty of mathematical and physical sciences-BUAP, Apdo. Postal 62-570, Av. San Claudio y 14 Sur, 72570, Puebla Pue, Mexico*
[d] *Physics Department, University of Wisconsin, Madison, WI, 53706, USA*

**Abstract**

Fermilab has designed and assembled a pixel-readout-chip sub-assembly containing five FPIX1 chips with flexible cable interconnections with which to address the technical issues involved in system integration for the proposed BTeV pixel detector. The module contains a total of 14400 pixel cells that need to be characterized in order to test the entire module. Software has been developed within the LabVIEW framework to control a set of instruments to perform threshold and noise tests on all five readout chips. These tests take only a few hours to run.
PACS: 07.05.-t; 07.05.Hd; 07.05.Kf

## 1. Introduction

We have developed a test stand for use in characterizing the FPIX1 front-end electronics chips that are one version in the FPIX sequence of VLSI chips being developed at Fermilab [1] for readout of the BTeV [2] pixel detector. This detector will provide high-resolution space points near the interaction region for use in reconstructing tracks and vertices. The information the detector provides will be used in the first level trigger to select events that have a high probability to contain secondary decay vertices. This means that all of the hit information from every beam crossing must be made available to the trigger processors. A beam crossing occurs every 132 ns.

In order to achieve the required resolution, $\leq 9\mu m$, the pixel unit cells must not only be very small, $50\mu m$ by $400\mu m$, but the charge deposited in each must also be digitized and read out. The FPIX chips contain front-end electronics cells with the same dimensions as the pixels on the sensors. The number of FPIX chips which will be bump-bonded to each sensor will depend on the number of cells on each FPIX chip. The FPIX1 version of the readout chips contains 2880 cells. Future iterations are expected to contain an even larger number.

If the readout is to be accomplished in the short time between crossings, the information must be sparsified so that only valid hit data is presented to the trigger processors. Which cells are read out is determined by a discriminator in each cell. If a signal above threshold is detected in the cell, then it is read out. The threshold setting for all cells on a single FPIX chip is the same. On FPIX1 it is set by a voltage input, called

[**] *Work supported by the U.S. Department of Energy under contract No. DE-AC02-76CH03000.*
[c] *Scholarship holder of CONACyT, MEXICO.*
[*] *Corresponding author. Tel.: (630) 840 2261; fax: (630) 840 8208. E-mail address: mavargas@fnal.gov.*

Vth0, using an external supply. On the next iteration, the threshold will be set by means of a digital to analog converter. The test stand was built for the purpose of testing that the measured threshold (charge required to record a hit) and noise levels for the 2880 discriminators on the FPIX1 chips were uniform enough that a single Vth0 for all of the cells would suffice. If there are significant cell to cell differences, then, since Vth0 must be set to a value well above the highest noise level measured, the charge information needed for accurate position measurements might be below the resulting charge threshold for some cells and would not be read out.

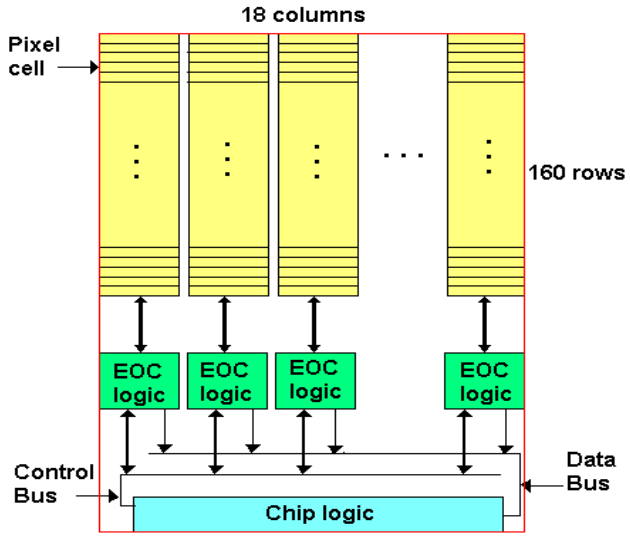
To gain experience on the technical issues involved in system integration, Fermilab has made a pixel-readout-chip sub-assembly containing five FPIX1 chips called the Fermilab pixel multi-chip module (MCM). Five FPIX1 chips have been integrated into a single module connected together by a flexible circuit, or High-Density Interconnect (HDI), made by Fujitsu Computer Packaging Technologies [3]. The tests described here were performed to characterize the 14,400 pixel cells contained on the five FPIX1 chips in the MCM. The tests were performed to characterize the chips alone, i.e., without bump-bonding them to a pixel sensor.

The entire BTeV pixel detector will contain nearly 30 million pixel unit cells each bump-bonded to its associated front-end electronics cell. Since the total number of components is so large, they will not be tested at a single facility, but will be tested at three or four different collaborating institutions. For this reason, the test stand has been developed using test equipment that is commercially available and in common use. Also, the software has been developed utilizing the Laboratory Virtual Instrument Engineering Workbench (LabVIEW) [4], which is widely used among the collaboration and which is familiar to both physicists and engineers. LabVIEW is a development environment based on the graphical programming language G. It is integrated fully for communication with hardware such as GPIB, VXI, PXI, RS-232, RS-485, and plug-in data acquisition boards by means of PCI interfaces. It also provides an extensive set of commands for file manipulation, including Ethernet transfers. This programming language is both very easy to understand and flexible. It also incorporates a number of sophisticated analysis techniques. We were thus able to develop the complex software needed for these tests entirely within this framework.

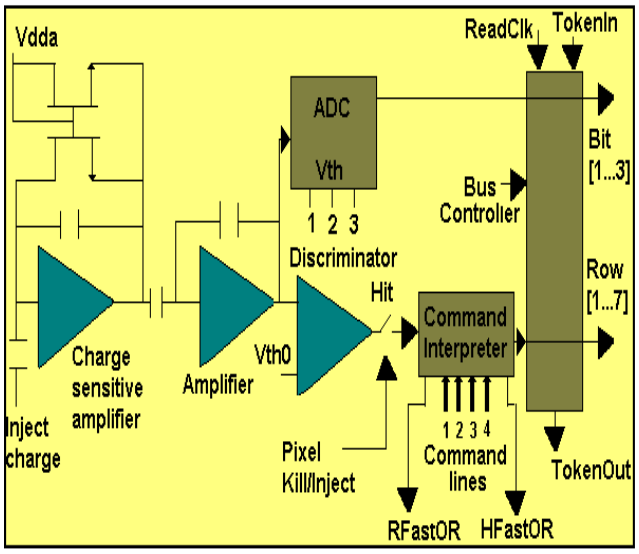
## 2. The FPIX1 chip

FPIX1 is the second integrated circuit in the sequence of iterations aimed at developing a chip that meets the BTeV requirements [1]. It is the first to contain much of the proposed column-based fast readout architecture. The chip is arranged in 18 columns, each with 160 rows of cells containing pixel front-end electronics, making a total of 2880 cells per chip.

The FPIX1 chip is composed of three mutually dependent parts, the pixel unit cells, the End-of-Column (EOC) Readout Logic, and the Chip Control Logic, as shown in the block diagram labeled Fig. 1a) [1]. The pixel cells each contain the front-end electronics for the sensor to which they will be bonded and the digital interface to the EOC logic as shown in Fig. 1b). The front end contains a charge-sensitive amplifier and a second amplification stage; the output of the second stage connects to a two-bit flash ADC and a discriminator. The discriminator output is asserted when the signal at the input of the discriminator is higher than the threshold (Vth0).




a)




b)

Figure 1. a) FPIX1 chip; b) Front-End of the FPIX1 chip.

The digital interface has two major components, the command interpreter and the pixel token and bus controller. The command interpreter has four inputs corresponding to the four EOC command sets that are part of the chip control logic. Four independent command sets are provided for de-randomization. These commands allow the front ends to record hits from the current beam crossing (BCO) while hits from previous beam crossings are being read out. Commands are presented by the EOC logic simultaneously to all pixel cell interpreters in a column. When an interpreter is executing the input command and the hit output from the discriminator is asserted, the interpreter associates itself with the particular EOC set, and simultaneously it alerts the EOC logic to the presence of a hit

via the wire-or'ed Hit Fast Or (HfastOR) signal. After that, the information is stored in the cell until the same EOC set issues an output or reset command. When the command is an output command, the interpreter issues a bus request and asserts the wire-or'ed Read Fast Or (RfastOR) signal.

The readout of hits then proceeds synchronously with the Read Clock (ReadClk). The EOC logic provides a column token; the token quickly passes pixel cells with no hit information until it reaches a cell that is requesting the bus. The cell is then read out. The data is composed of the ADC count bits [3:1] and the row address radd [7:0]. As the hit pixel is read out it automatically resets itself and withdraws its assertion of the RfastOR. This signal returns to its inactivated state while the remaining hit pixels are being read out [1].

The FPIX1 chip has 87 pads for external control and data readout. There are eight internal scan paths formed by shift registers with different depths, which we need to fill with information in order to program the chip. Two scan paths are the mask register path, which is used to set the programmable reset, and the mode path which contains the chip mode (triggered or continuous), the chip ID and the BCO Time Stamp. The last two scan paths are the Kill/Inject paths, which select the pixel cell or cells to be characterized. These last two paths go through all pixel cells in the module and are filled by 2880 times 5 shifts of data.

**3. Test Procedures**

a) Overview

The software we have developed takes advantage of the very flexible environment provided by LabVIEW as well as the advanced data analysis features it provides. While the use of LabVIEW and the relatively slow General Purpose Interface Bus (GPIB) for data acquisition meant that the tests were potentially time-consuming, several time-saving measures were taken that increased testing speed considerably.

First, two personal computers are employed, operating in parallel, both running LabVIEW programs under the Windows NT operating system. The first, PC1, is programmed for data acquisition. The second, PC2, performs the data analysis. The use of a logic analyzer with a Local Area Network (LAN) connection to receive the raw data and pass it on to PC2 makes it possible for the two computers to work simultaneously during the tests.

Second, two data generators are used to send commands to the chips and to the various instruments. This means that only small changes are needed on each cycle and thus a small amount of new information needs to be downloaded per data read cycle. Because of the many cycles involved, this results in a very large reduction in time. The data acquisition program running in PC1 synchronizes the action of the two.

We have demonstrated that an adequate characterization of each chip, by which we mean measurements of discriminator threshold uniformity and electronic noise dispersion, can be achieved by charge injecting 10% of the cells in a grid that covers the entire chip uniformly. The threshold and noise results vary randomly within the errors over the chip and do not appear to depend on the location of the cell being tested. We performed an initial test using a very low threshold on all cells to see which cells were working. This test revealed that the last 60 rows in each column did not record hits. This is a feature of the layout that will be corrected on the next submission. Furthermore, the first column on these prototype chips was made to have outputs that could be visualized on an oscilloscope and could not be read out by the EOC logic. Thus there are actually only 1700 cells to be characterized on each FPIX1 chip (17 columns times 100 rows) in the MCM. Ten percent of this is a total of 170 cells to be charge injected on each of the

five chips. Again, with time constraints in mind, we inject these 17 at a time, making sure that the ones we are injecting at the same time are separated by sufficient distance (20 cells) to avoid crosstalk.

Figure 2 contains photographs of the components of the FPIX1 test stand. Clockwise, starting at the upper left, we show the arrangement of the various instruments, an individual FPIX1 chip, the module with five chips mounted on it, and the final module assembly mounted on the test board that was designed to allow signals to pass to and from the module for communication with the test instruments.

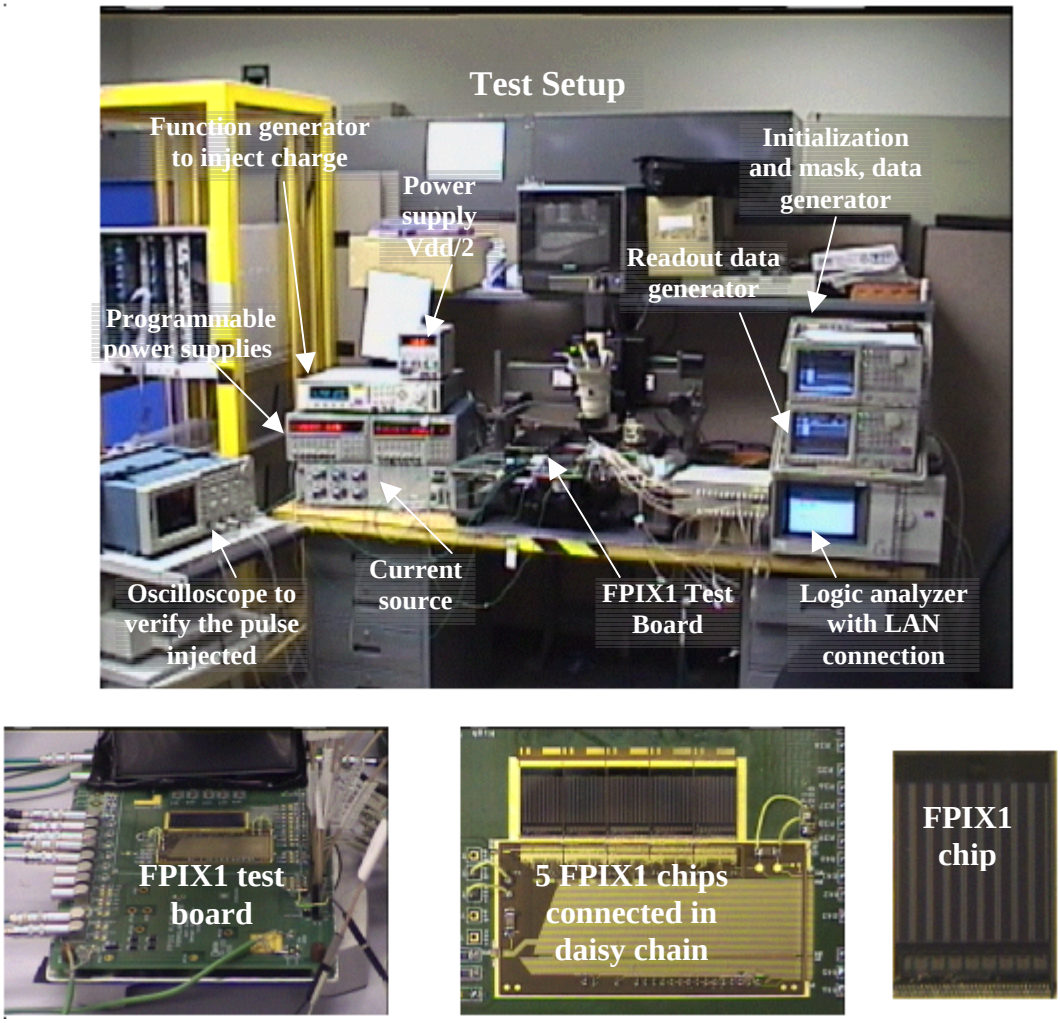



Figure 2. Test stand used to characterize the Fermilab pixel module.

b) Data Acquisition

PC1 contains an AT/TNT-GPIB interface to control the following instruments: Two Tektronix PS2520G programmable power supplies, a Hewlett Packard HP16500C logic analyzer system, a Stanford Research Systems DS345 function generator, two SONY/Tektronix DG2020A data generators and two SONY/Tektronix P3420 programmable outputs.

The test proceeds as follows: The program running in PC1 first initializes all the instruments used in the test and sets them to their nominal values. The two programmable power supplies provide power to the digital part of the FPIX1 chips, the analog part of the FPIX1 chips, and to all five Vth0 terminals of the chips on the module. These represent one of the inputs to the discriminators in the front ends. There are also two other instruments that are set manually. These are a regulated power supply, LEADER 718-SD, which provides power to the vdd/2 input of the FPIX1 to enable single-ended readout, and a current supply made at CPPM, Marseilles, to provide the feedback and Master Bias currents.

PC1 then initializes the five FPIX1 chips using the first of the two data generators. First, it configures the chips in continuous mode, and, since we have several chips connected in daisy chain, it gives each of the chips a unique ID number (ChipID). The FPIX1 chips deliver their information in two words, the data word and the control word. The data word contains the ADC value, the column and the row; the control word contains the ChipID number and the BCO number, which for these tests was always zero. PC1 then programs the five FPIX1s to kill (disable) all the pixels cells except 17 by means of the first data generator. The same data generator is then used to enable charge injection to the 85 cells (17 times 5) it has selected. Next PC1 sends commands to the logic analyzer to initialize it and to tell it to wait for a trigger that is based on bits in the data format that the chips deliver.

Following this, PC1 uses the second data generator to send a trigger to the function generator DS345, which injects charge to each of the enabled cells. Any cell in which this charge produces a signal at the output of the second amplification stage that is greater than Vth0 will have a hit in the output of its discriminator. The cell then asserts the wire or'ed HfastOR and stores the hit information until the EOC set issues an output command. When this occurs the command interpreter issues a bus request, asserts the wire or'ed RfastOR signal, and presents its data on the bus.

The second data generator has also been programmed to send the commands to perform the read out of the FPIX1 chips a fixed time after the charge is injected. Upon receipt of these commands, the FPIX1s send the data generated in response to the injected charge pulses to the logic analyzer in the order in which the token passes through them. The logic analyzer acquires the information in binary format and transmits it by means of a LAN connection to the other computer (PC2), which processes the data. For each set of 85 pixel cells PC1 repeats the injection of the same charge 500 times and does this for 60 different values of the charge it injects. When all the levels of charge have been scanned, the program changes the selection of the pixel cells and repeats the whole procedure until all 170 pixel cells on each of the five FPIX1 chips have been scanned.

The levels of charge must be chosen to bracket the voltage threshold. To do this we make a preliminary scan over the chip to determine the optimal range for the test. For each test we fix Vth0 at a particular value. We have carried out these threshold uniformity and noise tests in four different regions of operation.

c) Data Analysis

PC2 processes the information collected to obtain the mean threshold with its corresponding standard deviation (threshold dispersion) and the mean noise with its corresponding standard deviation for each FPIX1 chip on the MCM. When PC1 begins to acquire data, PC2 waits until the transfer of information from the first 85 pixel cells finishes, which takes approximately 30 minutes. After this, both programs run at the same time. The data acquisition wait time is 35 minutes (adding an extra 5 minutes more to avoid any potential conflicts). The data processing program needs to wait for this time

only once, because after the first transfer of data both programs can run simultaneously. The information processing takes exactly 30 minutes, so that the times are well matched. Thus, neither PC sits idle for an appreciable time during the test. While PC1 is acquiring the data from the current 85 pixel cells, PC2 is processing the information from the previous 85 pixel cells.

Information processing proceeds as follows: First, PC2 translates the binary code that the data acquisition system has acquired into a file that contains the ChipID number, the column number, the row number, the count of hits recorded above threshold and the voltage at each step, as well as the count of total hits (charge injections). Then it calculates efficiency for each pixel cell by dividing the number of counts recorded over the total number of hits. The discriminator output in the front-end of the FPIX1 chip is characterized by measuring its response as a function of the charge injected. Although only a single Vth0 is input to all 2880 cells on a chip, the value of the charge that triggers the discriminator may vary from cell to cell. This represents the threshold dispersion. The electrical charge injected can be calculated using the relationship between voltage and charge produced on the input capacitor. Since we have a capacitive load, C, of 6fF, the charge, $Q_{inject}$, is

$$Q_{inject} = VppC = Vpp \times 6 \times 10^{-15}\, coulombs, \tag{1}$$

where *Vpp* represents the voltage peak-peak delivered by the SRS DS345 function generator. The number of electrons that correspond to 1mV can be obtained by means of the following relationship:

$$(Q_{inject})e = \left(\frac{VppC}{e^-}\right)e = \left(\frac{Vpp \times 6 \times 10^{-15}}{1.6 \times 10^{-19}}\right)e, \tag{2}$$

with the result that

$$1mV \rightarrow 37.5e. \tag{3}$$

Because of the inherent noise of the electronics, the efficiency as a function of *Vpp* has the properties of an error function. For each pixel cell, a fit is performed to obtain the error function that best describes the experimental points. LabVIEW has a library of functions including one that performs a fit using the nonlinear Levenberg-Marquardt (Lev-Mar) method to determine a nonlinear set of coefficients which minimize chi-square [5]. First, PC2 plots the measured efficiency versus voltage *Vpp* (charge injected). The initial values of the threshold and noise are obtained using the data. The threshold value, $Vth_{\exp}$, is chosen to be the first value of the voltage that has greater than 50% hit efficiency. For the noise value, PC2 finds the first voltage for which the hit efficiency is greater than 81.5% and subtracts the value for 50% hit efficiency, which yields $\sigma_{\exp}$. These experimental values are used as the initial guess coefficients for the nonlinear Lev-Mar Fit technique, since the fits will converge more quickly the closer the initial values are to the solutions.

The nonlinear function needs to be specified, i.e., the relationship that describes the error function for each cell as a function of *Vpp*. For each point, *Vpp*, along the curve:

$$errf = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{Vpp} e^{-\left[\frac{(V-Vth)^2}{2\sigma^2}\right]} dV \tag{4}$$

This technique provides two ways to calculate the Jacobian; i.e., the partial derivatives with respect to the coefficients, needed in the algorithm. These methods are:

Numerical calculation: Uses a numerical approximation to compute the Jacobian.

Formula calculation: Uses a formula to compute the Jacobian. For this one needs to specify the Jacobian functions, which are the partial derivatives of the error function with respect to $\sigma$ and with respect to *Vth*. These functions are:

$$\frac{\partial errf}{\partial \sigma} = \left( \frac{(V-Vth)^2 - \sigma^2}{\sqrt{2\pi}\sigma^4} \right) \int_{-\infty}^{Vpp} e^{-\left[\frac{(V-Vth)^2}{2\sigma^2}\right]} dV \tag{5}$$

$$\frac{\partial errf}{\partial Vth} = \frac{(V-Vth)}{\sqrt{2\pi}\sigma^3} \int_{-\infty}^{Vpp} e^{-\left[\frac{(V-Vth)^2}{2\sigma^2}\right]} dV \tag{6}$$

The program repeats the above procedure for all of the pixel cells being tested. When the information for all the pixel cells has been analyzed, the results are output in five files, each of which contains the final information for all cells on one chip in the MCM, i.e., the column number, the row number, the best fit *Vth* in units of volts and electrons, and the best fit $\sigma$ (noise) in units of volts and electrons. We then run a second program to histogram these quantities for the 170 cells on each of the five chips and to obtain the Gaussian curves that best fit these histograms for each file. We thus obtain the mean threshold, the sigma threshold (threshold dispersion), the mean noise and also the noise dispersion, all in units of volts and electrons for the 170 cells on each FPIX1 chip. These results represent the characterization of the chips on the module. A flow diagram of both programs, i.e., the data acquisition program and the data processing program is shown as Fig. 3

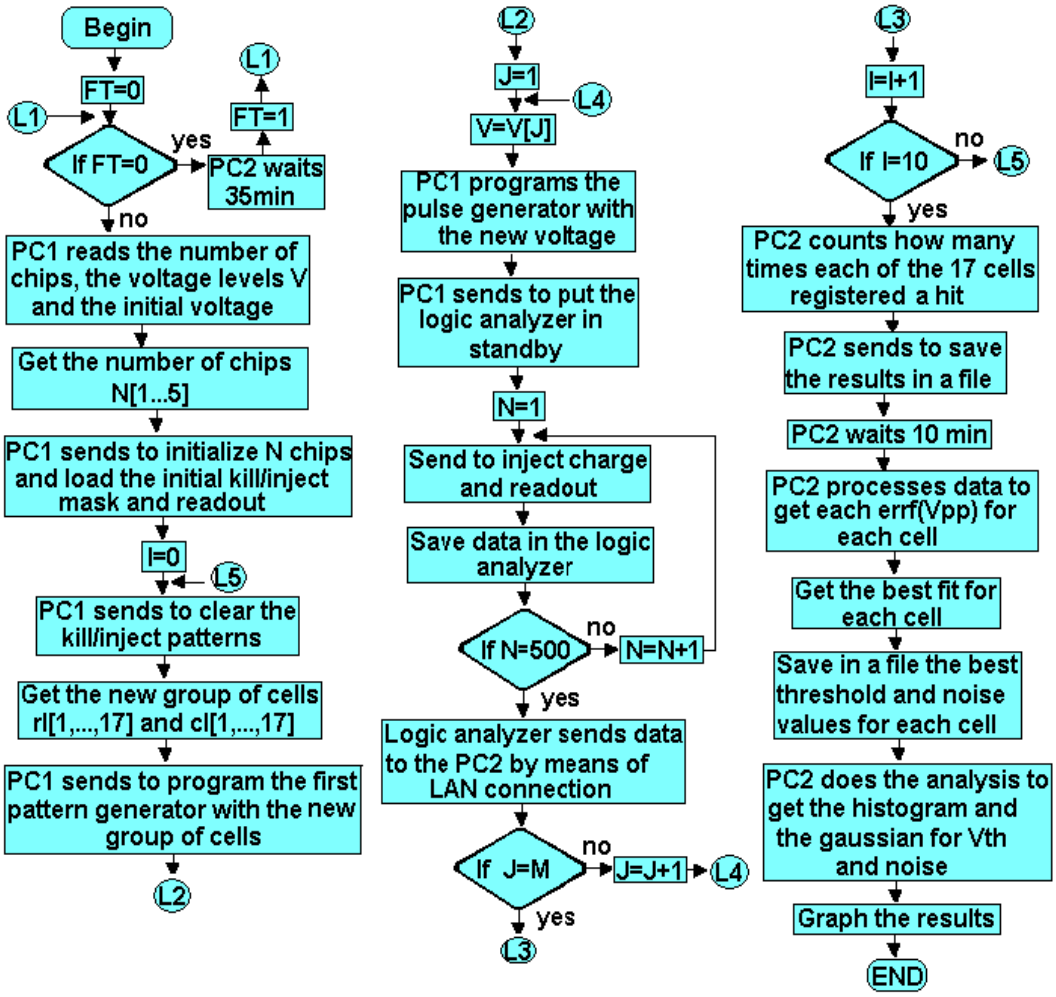



Figure 3. Flow diagram of the software designed.

**3. Results**

The software designed to characterize the MCM yields the threshold uniformity and noise dispersion for the five FPIX1 chips on the module in 5 hours of running with 500 hits (charge injections) per cell at each of 60 values of *Vpp*. This time could be reduced by as much as a factor of five by using PCs with faster processing speeds that are already available in the marketplace. The time required for data acquisition and analysis has been considerably decreased by taking advantage of the flexibility of the LabVIEW programming environment and by using the LAN connection to transfer the data from the logic analyzer to a second computer for analysis.

PC2 processes a very large amount of information. The raw data for 850 pixel cells (170 cells times five chips) is organized into 10 files, each containing the information for a group of 85 pixel cells. An individual file contains 931 KB of data. In a particular characterization we have to scan 60 levels of voltage in order to cover the full range. The total information acquired is obtained by multiplying the size of each file by the total number of files acquired, which in this case is 600 files. Thus, the program processes a

total of approximately 550 MB of information to carry out the characterization in each range of operation.

Fig. 4 shows the efficiency versus *Vpp* for a single pixel cell and the error function that results from the fit to the data. Figures 5 and 6 show the histograms of the threshold and noise dispersion and curves corresponding to the Gaussian fits for one chip mounted in the MCM. Table 1 and Table 2 contain the results of the characterization for the five FPIX1 chips for four regions of operation.

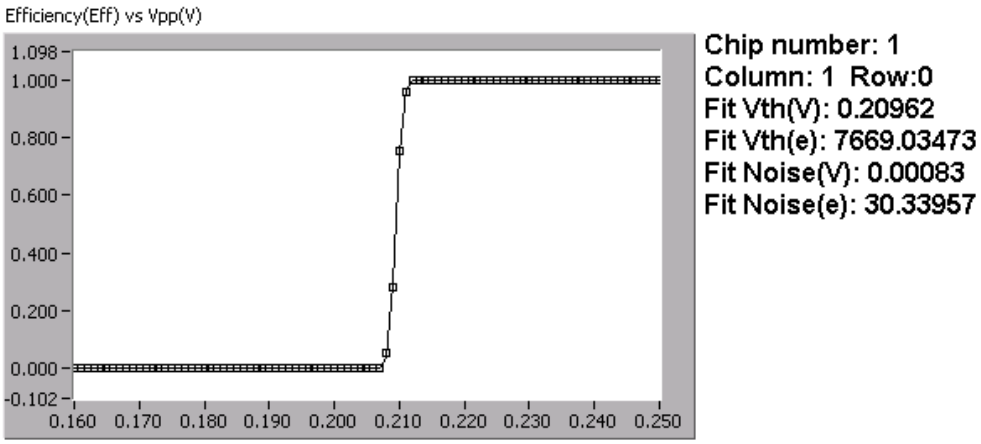



Figure 4. Error function from one pixel cell on one FPIX1 chip.

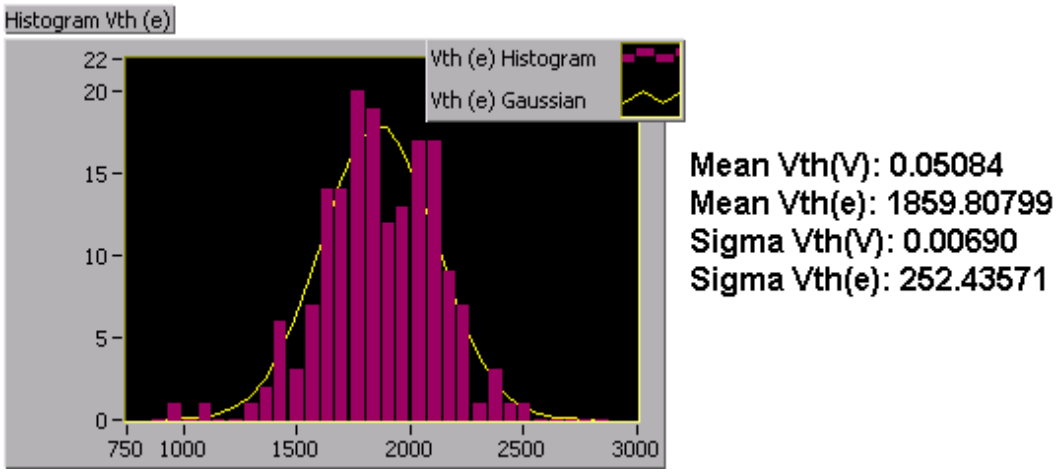



Figure 5. Voltage Threshold distribution in the chip number 1.

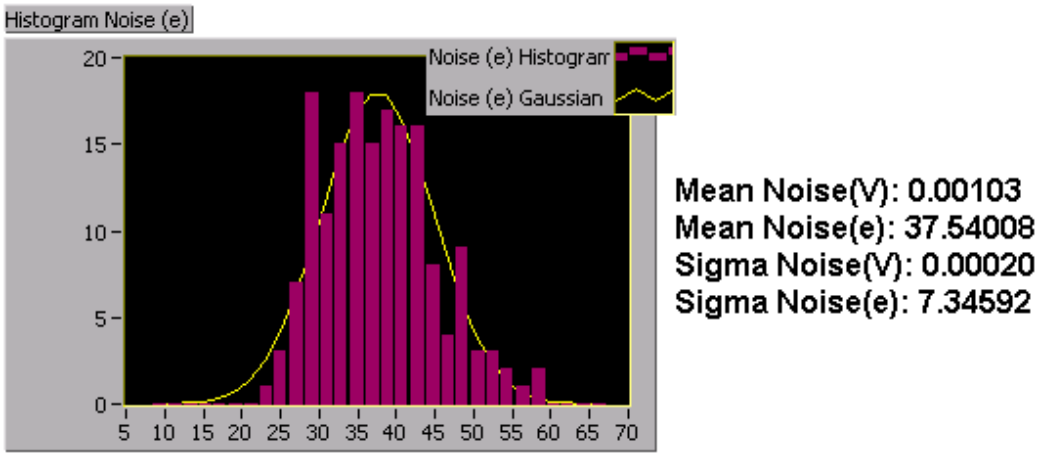



Figure 6. Noise distribution in the chip number 1.

| | Mean threshold (e-) | | | | Sigma threshold (e-) | | | |
|---|---|---|---|---|---|---|---|---|
| **Vth0 (V)** | **1.95** | **2.05** | **2.15** | **2.25** | **1.95** | **2.05** | **2.15** | **2.25** |
| Chip 1 | 7321.1 | 5437.2 | 3660.4 | 1859.8 | 221.7 | 240.2 | 247.2 | 252.4 |
| Chip 2 | 7400.3 | 5506.4 | 3720.7 | 1919.5 | 189.9 | 208.8 | 213.6 | 224.9 |
| Chip 3 | 7289.9 | 5402.2 | 3623.3 | 1830.8 | 209.1 | 219.1 | 227.0 | 238.4 |
| Chip 4 | 7355.3 | 5472.7 | 3691.4 | 1895.2 | 187.5 | 180.9 | 186.3 | 197.1 |
| Chip 5 | 7354.3 | 5430.9 | 3616.7 | 1770.7 | 217.6 | 235.1 | 246.6 | 264.3 |

Table 1. Final results of the characterization for threshold uniformity on the MCM for four regions of operation.

| | Mean Noise (e-) | | | | Sigma Noise (e-) | | | |
|---|---|---|---|---|---|---|---|---|
| **Vth0 (V)** | **1.95** | **2.05** | **2.15** | **2.25** | **1.95** | **2.05** | **2.15** | **2.25** |
| Chip 1 | 39.3 | 37.8 | 37.9 | 37.5 | 8.7 | 7.8 | 7.4 | 7.3 |
| Chip 2 | 36.5 | 36.3 | 36.7 | 36.6 | 8.8 | 7.7 | 7.9 | 7.2 |
| Chip 3 | 38.0 | 38.2 | 37.7 | 37.7 | 9.8 | 9.1 | 8.9 | 9.2 |
| Chip 4 | 32.2 | 32.0 | 31.6 | 31.1 | 8.9 | 7.9 | 7.9 | 7.2 |
| Chip 5 | 35.3 | 35.6 | 36.0 | 37.6 | 10.2 | 9.0 | 9.8 | 9.9 |

Table 2. Final results of the characterization for noise on the MCM for four regions of operation.

**Acknowledgements**

The authors wish to thank S. Zimmermann for valuable advice during the course of this project and G. Chiodini and G. Cardoso for technical assistance. We also want to express our appreciation to the Fermilab staff and management, with special thanks to J. Appel, S. Kwan, and E. Barsotti. This work would not have been possible without the support and encouragement of A. Cordero and A. Fernandez of FCFM/BUAP. The support of the U.S. DOE and CONACyT, Mexico, is also gratefully acknowledged.

**References**

[1] G. Cancelo, D. Christian, J. Hoff, A. Mekkaoui, R. Yarema, S. Zimmermann, *High readout speed pixel chip development at Fermilab*, Fourth Workshop on Electronics for LHC Experiments, 528-532, 1998.

[2] J. Butler, S. Stone, et al., *Proposal for an experiment to measure mixing, CP violation, and rare decays in Charm and Beauty particle decays at the Fermilab collider- BTeV*, submitted to the Fermilab Program Advisory Committee, May 2000. Stage I approval given June 27, 2000.

[3] B. Chou, et al., *Ultra-High-Density Interconnect Flex Substrates*, HDI Magazine, December 1998.

[4] National Instruments Corporation, LabVIEW Users Manual, 1998.

[5] National Instruments Corporation, *LabVIEW Function Reference Manual,* 1998