A MONITORING SYSTEM FOR A GAS-SHEET BEAM PROFILE MONITOR ON LINUX WITH EPICS

Y. Yuasa*, J. Odagiri, N. Yamamoto, Y. Hashimoto, S. Machida,
Y. Mori, T. Morimoto, S. Muto, T. Toyama, KEK, Tsukuba, Japan
T. Fujisawa, NIRS, Chiba, Japan

Abstract

A gas sheet beam profile monitor (SBPM) will be used to investigate a proton beam in a 50-GeV main ring of J-PARC. The control system of the ring and also the monitor employs EPICS. The beam profile is monitored by a camera, which detects a luminescence light generated from a molecular gas-sheet target by interacting with a proton beam. The camera is connected to a frame grabber board (image capture board), which is operated by a Linux-PC and use the PCI bus. Input Output Controller (IOC) on Linux with EPICS 3.14 is built to implement a control of frame grabber board.

We will report the implementation and performance of the monitoring system based on EPICS running on a Linux-PC, the frame grabber board of which system handles huge amount of data in a rapid cycle.

INTRODUCTION

A SBPM is developed to apply for an investigation of a proton beam in a 50-GeV main ring of J-PARC [1]. It is a non-destructive beam profile monitor using a neutral gas target like a sheet across a synchrotron orbit. A camera observes a luminescence light from the gas target yielded by collision with the synchrotron beam.

The frame grabber board acquires an image data from the frame grabber board synchronously with the beam period of the accelerator, 0.3 Hz. This clock is used to a trigger pulse of the frame grabber board to start acquiring a signal from the camera. The repetition rate of data acquisition of the frame grabber board is limited not only by its process but also by signal transportation from a camera. Its standard is NTSC for instance, on which the frame rate is about 33msec/frame. The frame size is VGA and the resolution of Analogue to Digital conversion (ADC) is 10 bits per pixel. Thus the frame buffer size is about 620 Kbytes and the average speed of a raw buffer process requires 19 Mbytes/sec. The schematic view of a monitoring system is shown in Figure 1.

We need to control the frame grabber board with acquiring a synchronous, rapid cycle and huge buffer in IOC running on a Linux-PC. An observer needs not only an image data but also any analysed data, which are X- and Y-projection data and their characteristic parameters, for instance a peak position, peak height, width and area within the region of interest which an observer sets. In IOC, a new type of record is created to process a raw data and to store a raw data and also any analysed data.

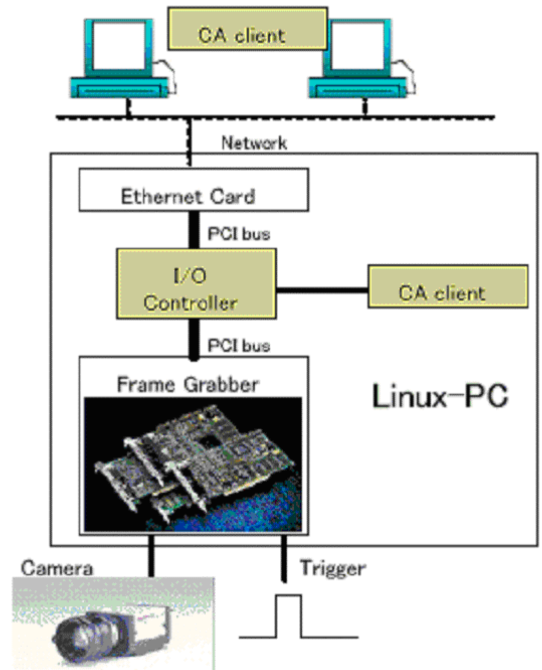

Figure 1: Scheme of a monitoring system.

In this paper, details of the device support are described in Section 2. In Section 3, details of the record support are described. After that, in Section 4, the performance of the monitoring system is described.

EQUIPMENTS

This section describes details of the hardware of the monitoring system. The specification is listed in Table 1.

Table 1: A specification of Hardware

CPU	Intel Xeon 2.8GHz (Dual)
Motherboard	SUPERMICRO P4DCE+
Memory	RIMM PC800-40 512MB (x4)
Graphics card	nVIDIA GeForce4 Ti 4600
Hard disk drive	MAXTOR 6Y120L0
Frame grabber board	MATRIX-Vision mvTITAN-G1

A computer for data acquisition of SBPM needs a high performance of graphics processing. Dual high speed CPUs are installed for parallel processing of multiple threads in EPICS/IOC. We adopted Redhat Linux 7.3, which supports dual CPU hardware architecture and parallel processing, as a platform of EPICS 3.14.

The graphics data is saved to and read from a hard disk drive (HDD) with IDE-bus (ATA133). The data transfer maximum speed of the HDD is 133Mbytes/sec. It is

* Present Address : Kawasaki Heavy Industries LTD.

enough to meet the requirement of an average speed of data acquisition in the monitoring system of SBPM.

The graphics card has high performance to display a graphics data with X-window system at high repetition rate. Graphics cards commonly have 8bit resolution of mono color. To display higher resolution data in a TV-monitor, a pseudo colour image data or a compressed colour image data should be created in the process routine of a record support module.

The requirements of a frame grabber board are the followings:

- Higher bit resolution of ADC
- External trigger input
- Acceptable for non-standard source/signal
- Real time processing more than 33msec/frame
- Linux device driver and library

We adopted mvTITAN-G1 produced by MATRIX Vision GmbH [2], which meets the above requirements.

Owing to a recent development of the frame grabber board, the resolution of ADC is upgraded from 8 bits per pixel to 10 bits per pixel. By reconstructing a profile data, we are able to measure beam profile parameters precisely.

In order to interface the frame grabber board with EPICS, we needed to develop a device support module for the use of the board. Since the vender of the board supplies a Linux device driver and a library with the board, the development of the EPICS device support did not involve any kernel-level programming.

DEVICE SUPPORT

This section describes the design and technical details of the device support module.

The initialisation routine in the device support module spawns following four threads;

- **Titan server** acquires a graphics data from a frame grabber board when an external trigger pulse triggers it, and then moves the data to a record on memory.
- **Writing server** saves the graphics data taken by the "Titan server" into a file on a HDD.
- **Reading server** reads the graphics data saved by the "Writing server" from the file, and then writes it to the record on memory.
- **X dump server** dumps the graphics data taken by "Titan server" or "Reading server" to display a reconstructed image on an X-window terminal with checking a status of the data acquisition.

The synchronization of the four threads is shown in Table 2. The scan field of an EPICS record is used to switch the execution between two operation modes, a "real mode" that acquire data from a frame grabber board ("Titan sever") and a "simulation mode" that takes data from a file on a HDD ("Reading server").

Table 2: Synchronisation of the threads in device support module

mode	real mode	simulation mode
Scan Field	IO Intr.	Periodic
Signal flow	Titan Server ← Wait for a trigger on a Frame Grabber; Signal to proceed; Signal to proceed; Writing Server; X dump Server	Reading Server; Signal to proceed; X dump Server

If the scan field has a value of "I/O Intr", the "Titan server" starts processing when graphics data arrives. After this process finishes, an event signal is sent to both the "Writing server" and the "X dump Server" in order to make them to work. Because the "Writing server" thread must save every event, the "Titan server" needs to wait if a buffer in the "Writing server" is full. However, the "X dump server" does not wait even if its buffer is full because rare misses are acceptable for a monitor.

If the scan field has a value of "Periodic", "Reading server" starts processing. Every time the process finishes, an event signal is sent in order to make "X dump server" display an acquired graphics data on the terminal.

RECORD SUPPORT AND DATABASE

A new record type named as "graphicsRecord" is created in order to handle many different types of data related to a beam profile, such as raw image data, background data, corrected image data, horizontal/vertical projection data and others. Having all the data in a single record allows the users to create an application program, i.e. an instance of EPICS run-time database easily.

The fields specific to "graphicsRecord" are listed in Table 3. In the record support, some fields connected to physical parameters of a beam profile are calculated from the data taken with the device support mentioned in the previous section. The sequence of the calculation is as follows:

- Background process
- Reconstruction of projection data
- Analysis of projection data
- Creation of image data set for a display

A background process consists of a process of background collection and background subtraction to correct an acquiring data. These two processes are switched by a value of a field for that purpose.

To speed up the processing of the graphics data, the execution should not loop many times scanning over the huge data. We carefully programmed the code for the reconstruction and the analysis of the projection data in such a manner that much of the tasks are done in a scan.

A thread spawned in "graphicsRecord" creates a rough resolution graphics data and a special format graphics data, which are used to display image data with a Cartesian plot of DM2K and others.

Table 3: List of the record type "graphicsRecord"

Contents	Size
:: Control flags ::	
Run/Stop control of frame grabber board	Char
Switch of background collection	Char
Switch of background subtraction	Char
Switch of invoking a record process	Char
:: Graphics data ::	
Raw graphics data	640*480*long
8bit graphics data	640*480*short
Rough graphics data for plot in DM2K	640*480*short
Background data	640*480*long
Correction data	640*480*long
Region of interest(ROI) in horizontal/vertical axis for area	short
Horizontal projection data	640*long
Vertical projection data	480*long
:: Analysed data ::	
Peak position of Hor./Ver. profile	short
Mean position of Hor./Ver. profile	double
Peak height of Hor./Ver. Profile	long
Height at mean pos. of Hor./Ver. profile	double
Sigma of Hor./Ver. Profile	double
FWHM of Hor./Ver. Profile	double
Area of Hor./Ver. Profile	double
ROI in Hor./Ver. profile	short

PERFORMANCE

Dual CPUs effectively process some threads in IOC by a parallel processing. A process speed of data acquisition keeps high.

The total processing time required for acquisition, processing and displaying remained is around 20 ms for an event which is below 30 ms for an event, which is a cycle speed of NTSC standards, with DM2K running on the same PC.

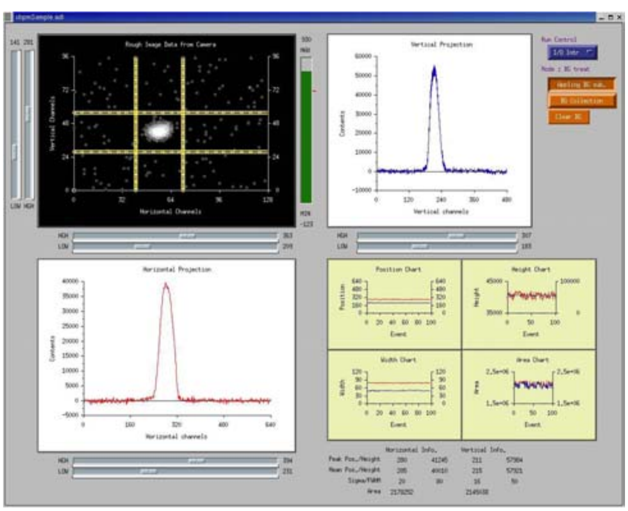

Figure 2: GUI control panel of SBPM with DM2K

A CA application of the monitoring system on SBPM is created with DM2K, shown in Figure 2. The camera takes an image of diode light. The presented data in Figure 2 uses corrected one with background data. A shape (left-top), a histogram of vertical projection (right-top) and of horizontal projection (left-down) of diode light is clearly found in TV-monitor. The right-down picture in Figure 2 shows a chart of physical parameters taken from horizontal and vertical projection data.

CONCLUSION

The monitoring system of SBPM was developed with EPICS 3.14 on a Linux-PC as a cost-effective solution.

A design of a device support that uses multiple-threads enables four tasks of acquiring the graphics data from the frame grabber board, writing it to a file on a HDD, reading it into memory from a file on a HDD and dumping it to a X-terminal, being processed concurrently.

New record type, "graphicsRecord", was created with various fields, which were assigned to a set of graphics data and physical parameters of a projection data, in order to reduce the burden in creating EPICS database instances.

The system showed a performance high enough to be used in practical operations. The total processing time required for acquisition, processing and displaying remained well below 30 ms for an event, with DM2K running on the same PC.

REFERENCES

[1] Accelerator Group JAERI/KEK Joint Project Team, "Accelerator Technical Design Report For J-PARC", KEK Report 2002-13.
[2] http:// www.matrix-vision.de/