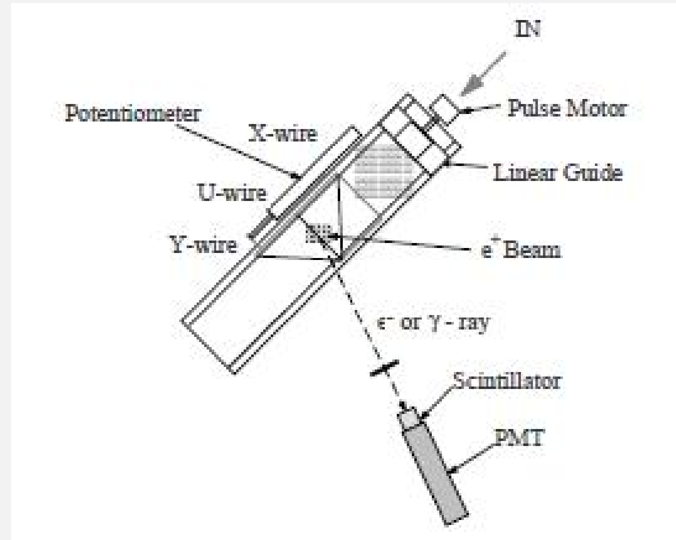
Development of fast controls for Beam Wire Scanner at SuperKEKB

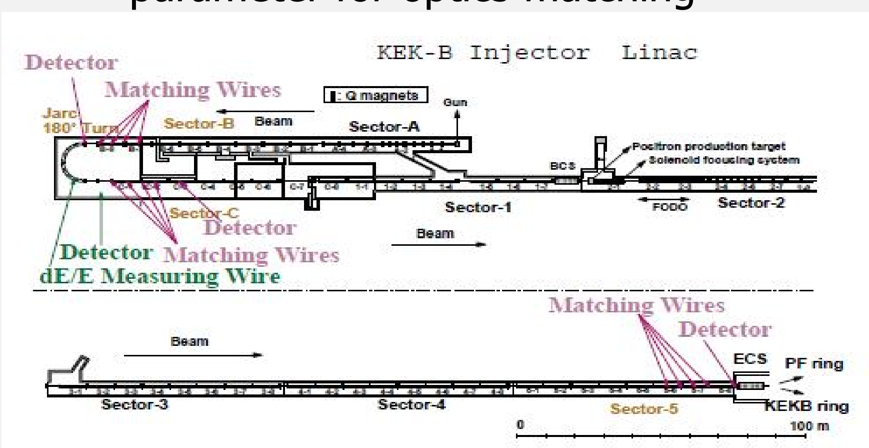
Anindya Roy

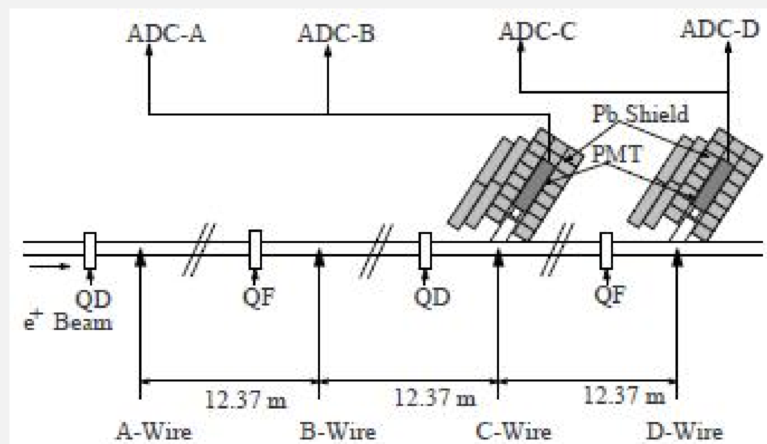
Variable Energy Cyclotron Centre
Dept. of Atomic Energy, India

What is Wire-scanner?

- Used for Non-destructive monitoring of beam profile
- A frame holding 100μm Tungsten wire forming X, Y & U wires perpendicular to beam
- A stepper motor drive to move the frame inside beam line
- A PMT with Plastic Scintillator to detect Bremsstrahlung emitted due to interaction of wire with Beam
- A control & data acquisition system
- A set of three (atleast) wire-scanner for measuring beam emmittance & Twiss parameter for optics matching



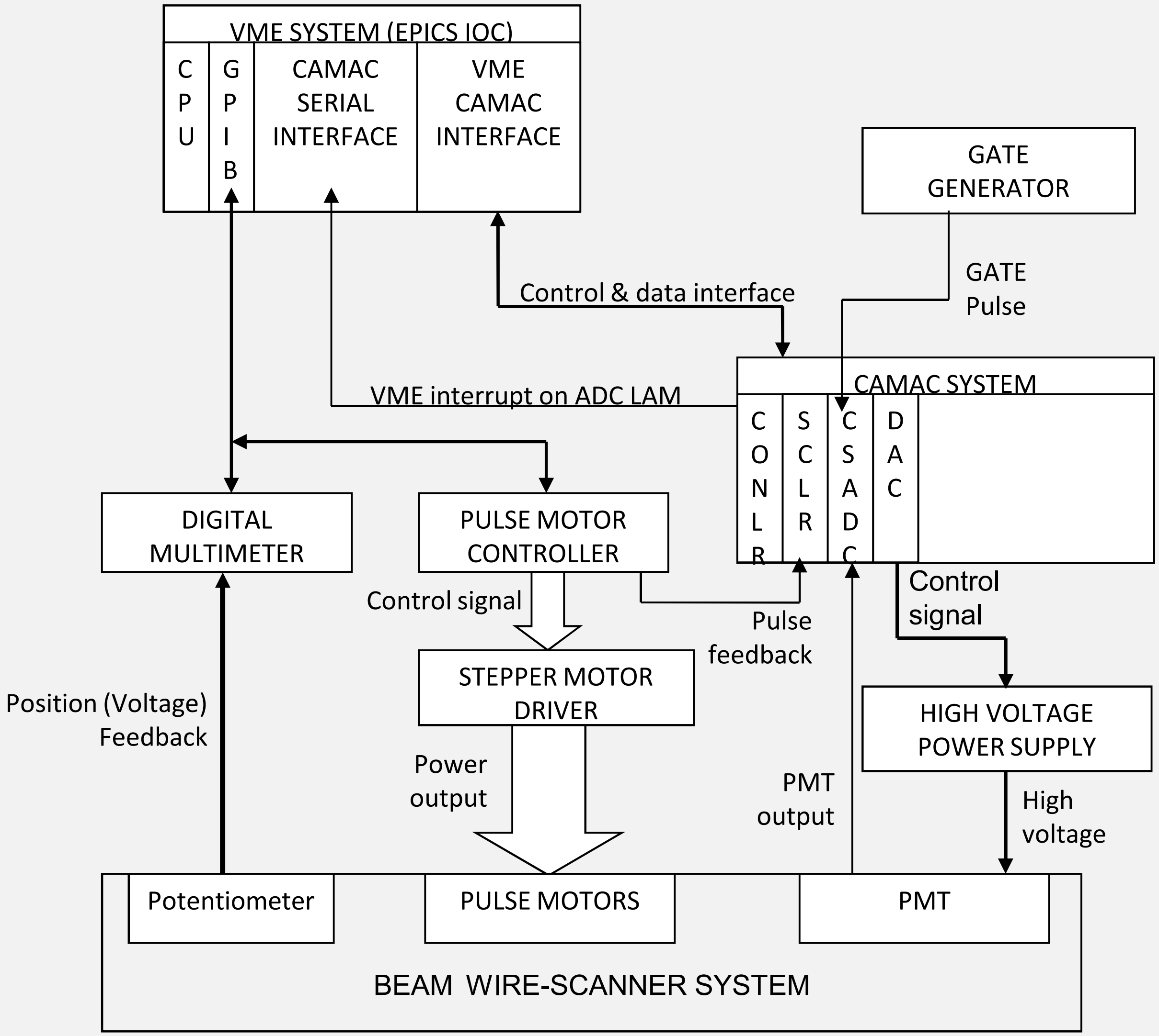




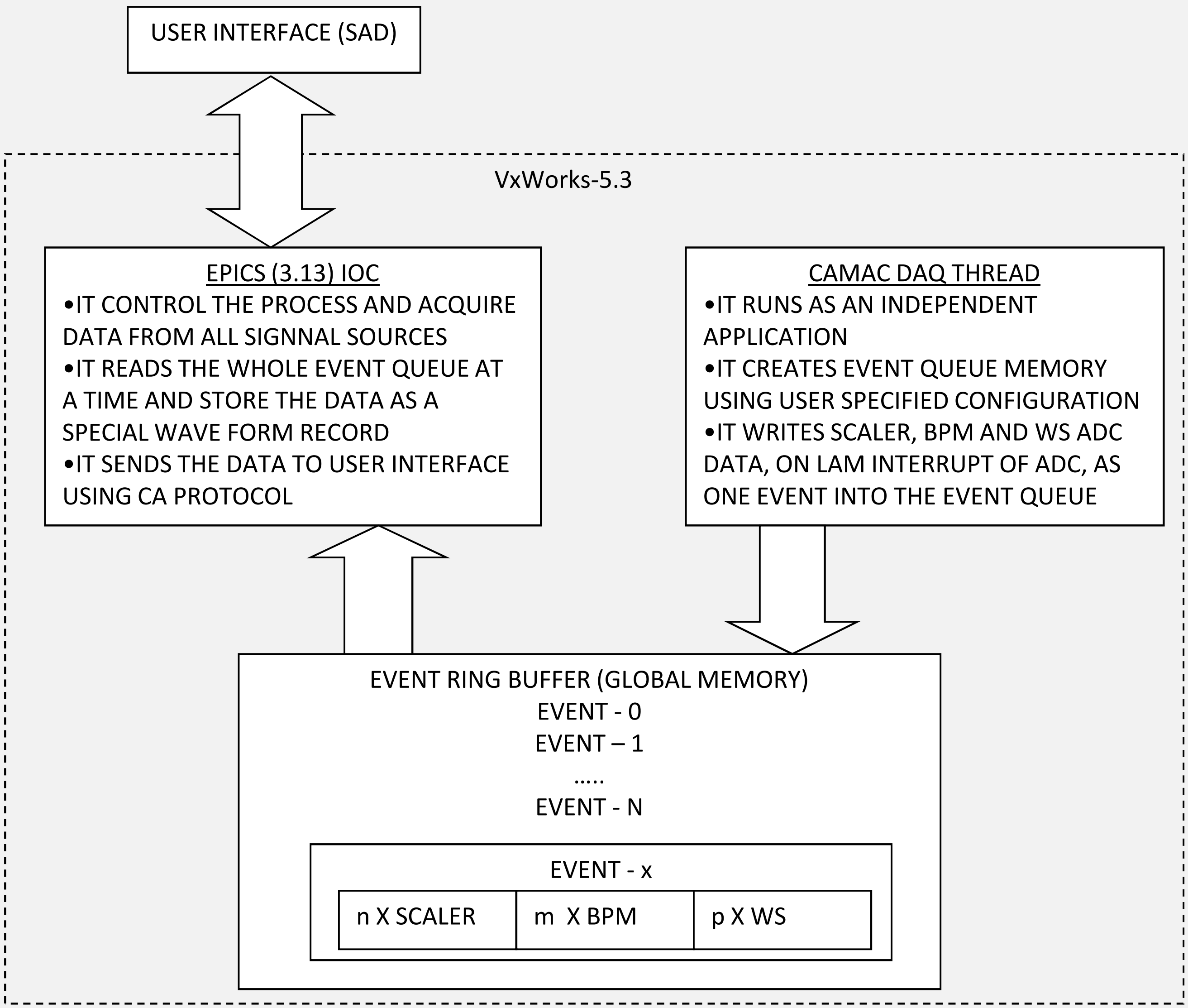
Existing system

- VME/CAMAC system
- EPICS Base-3.13 and Vx-Works 5.3
- PowerCore 6750 CPU card
- 12 ch Lecroy CAMAC ADC
- VME GPIB controller to communicate with Pulse motor controller & Digital Multimeter
- ADC Gate signal generated from Beam timing signal, no information about beam mode
- Beam mode identification using dedicated ADC channel for each beam mode
- IOC independent process for acquiring Beam data and wire position signal and saving in buffer
- Uniform wire speed while scanning the beam
- Not suitable for acquiring multiple beam mode data

Existing system (hardware architecture)



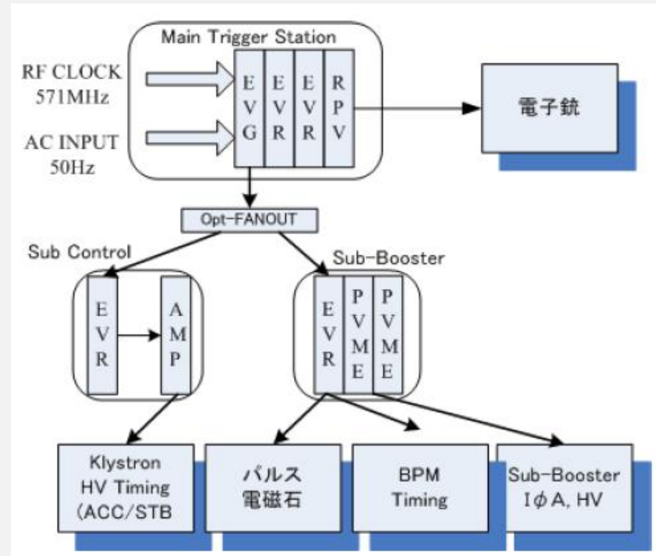
Existing system (software architecture)



Why is the new controls for Wire-scanner?

- 8 Gev LINAC, simultaneously, injects of e^- and e^+ beam of different characteristics into
  - KEKB high-energy ring (HER) – 8.0 GeV, 2nC
  - KEKB low-energy ring (LER) – 3.5 GeV, 2nC
  - Photon Factory (PF) – 2.5 GeV, 0.1nC
  - PF/AR – 3.0 GeV, 0.1nC
- The operation of all equipments along BT are synchronized by LINAC Timing system for simultaneous injection
- A wire-scanner data acquisition system, synchronized with Timing system, can measure beam profiles of multiple beam modes simultaneously in a single run
- Identification of beam modes can be done using timing information available from timimg system
- Simultaneous measurement reduces measurement time and hence improves the overall efficiency of the transport line
- May help in PF and PF/AR beam studies without degradation of luminosity at KEKB!

LINAC Timing system



- MRF's series-230 Event Generator/Receiver
- 114.24 MHz event rate
- Multi/Single mode fiber connectivity
- Timing precision < 10ps
- 20msec Beam switching
- Control approx. 1000 devices in LINAC
- Every pulse (20msec) corresponds a beam mode
- 10 different defined beam modes
- One beam pulse contains several event codes
  - Atleast one Main event code and a preparation code
  - 50 defined event codes
- Main event code and preparation event code are in sequence
  - Main event triggers timing signal
  - Preparation event trigger software to exchange analog & delay parameters
- Total number of receiver: 17 (+ 1)

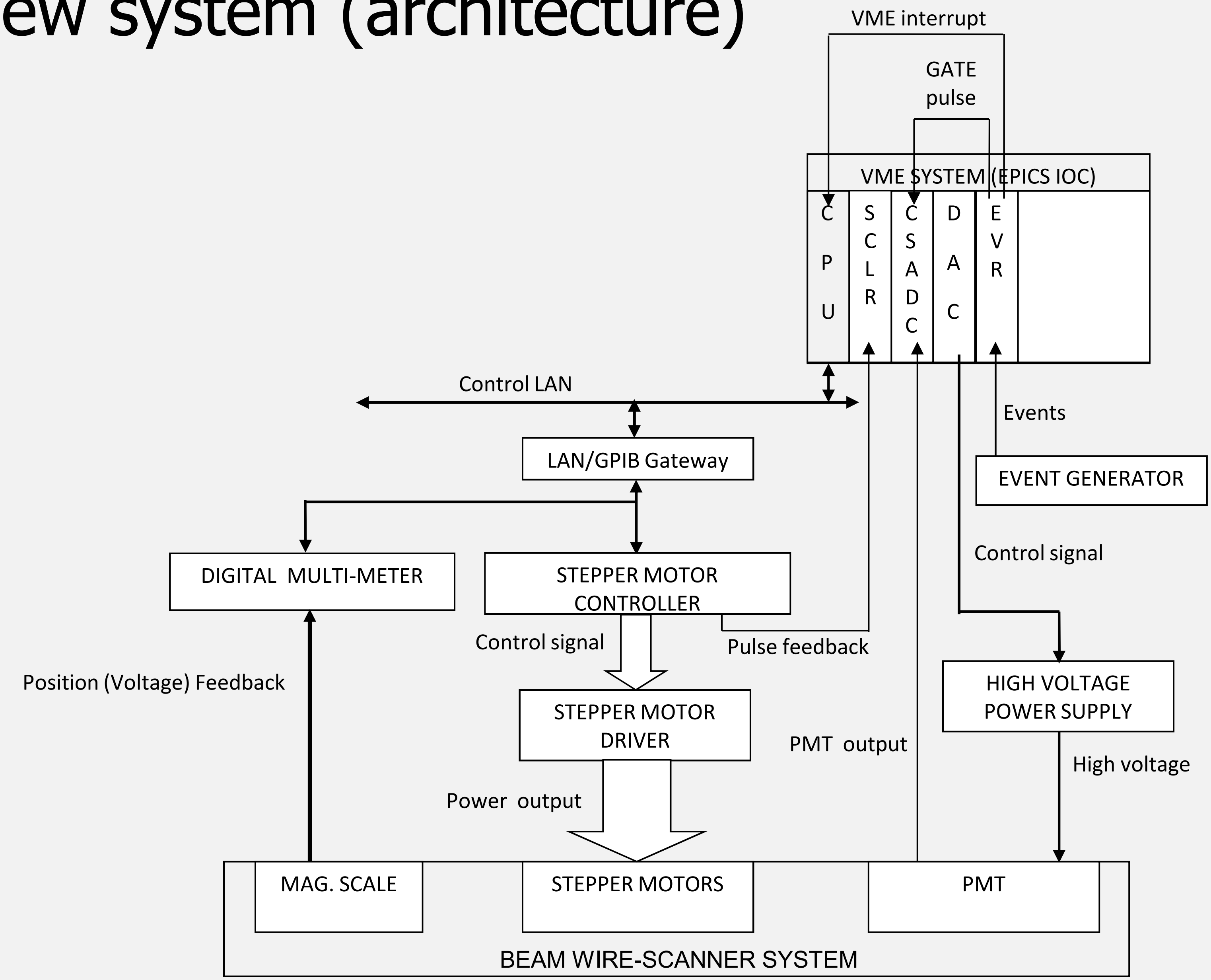
Reference:

"Timing system towards SuperKEKB controls " – Kazuro Furukawa, EPICS Collaboration Meeting, NSRRC, June 2011

New system

- Emerson MVME5500 CPU card, EPICS Base-3.14.12.1 and Vx-works 6.8 combination
- VME based ADC (14bit, 15μsec conversion time), Scaler and DAC hardware.
- MRF's event receiver module (VME-EVR-230RF) - synchronizing the data acquisition process with LINAC timing system
- A LAN/GPIB converter to communicate with Pulse motor controller (PMC) and Digital Multimeter (DMM) for control and data acquisition.
- ADC Gate generated by event receiver – less complicated hardware setup
- Multispeed wire movement to minimize time of scan and maximize useful data
- Options to user for selecting beam modes for data acquisition
- Application specific EPICS record to retain data format for SAD user interface
- Installed at Sector 5 of LINAC BT

New system (architecture)



Event Receiver

- Basic feature
  - Manufacturer – Micro Research Finland, Model - VME-EVR-230RF
  - Bit rate 1.0 to 2.5 Gbps, event clock rate 50 MHz to 125 MHz
  - Four programmable front panel TTL outputs
  - Two front panel TTL inputs
  - Three differential CML pattern outputs capable of RF recovery
  - Two universal I/O slots
  - Rear I/O
  - Jitter typically < 15 ps rms for TTL outputs, < 5 ps rms for CML outputs
  - Support VME64x CR/CSR addressing mode
- EPICS device support
  - ✓ Already available at EPICS Hardware support inventory (mrfioc2-2.0.0.tar.gz) (http://sourceforge.net/projects/epics/files/mrfioc2/)
  - ✓ As current EPICS version does not support CR/CSR addressing mode in OS independent manner, hence devLib2 module (devlib2-2.2.tar.gz) (http://sourceforge.net/projects/epics/files/devlib2/)
  - ✓ EPICS MSI tool to build the above modules (msi1-5.tar.gz) (http://www.aps.anl.gov/epics/extensions/msi/index.php)

Event Receiver (configuration)

- EPICS device driver configuration (Different experience from usual VME device configuration!)
  - Setting up module (IOC initialisation)

    mrmEvrSetupVME(NAME, SlotNo, MapAddr, IntrLevel, IntrVectorAddr)

    NOT Identified by SLOT No, but by NAME - different from usual EPICS VME device support style!
  - Each feature (register) is accessed by "Module name:Feature name" (OBJ) and property name (PROP) fields

    e.g. field(OUT , "@OBJ=EVR1:Pul0, PROP=Delay") => writing into register

    field(INP , "@OBJ=EVR1:Pul0, PROP=Delay") => reading from register

    Little difficult to trace NAME of each register, hence better to copy and modify sample records!!
  - Four record types – ai, ao, longout & longin
  - Four device types (DTYP)

    "Obj Prop uint32" – for longin & longout records

    "Obj Prop double" – for ai & ao records

    "EVR Pulser Mapping" – for mapping Event codes to pulse generator (longout record)

    "EVR Event" – for mapping Event codes to EPICS Event (longout record)
- Important properties to configure
  - Enabling the module - *field(OUT , "@OBJ=EVR1, PROP=Enable")*
  - Setting up Clock - *field(OUT , "@OBJ=EVR1, PROP=Clock")*
  - Setting up Time stamp source - *field(OUT , "@OBJ=EVR1, PROP=Timestamp Clock")*
  - Mapping front panel output to pulse generator - *field(OUT , "@OBJ=EVR1:FrontOut0, PROP=Map")*
  - Enabling & Configuring pulse generator (i.e. delay, width, polarity) - *field(OUT , "@OBJ=EVR1:Pul0, PROP=Polarity")*
  - Mapping timing event to pulse generator - *field(OUT , "@OBJ=EVR1:Pul0, Func=$(F=Trig)")*
  - Mapping timing event to EPICS Event - *field(OUT , "@OBJ=EVR1,Code=31")*

Event Receiver (tuning)

- Synchronization of reference clock with incoming events from event generator
- Clock Referenece generated internally
  - Micrel SY87739L Protocol Transparent Franctional-N synthesizer, reference clock of 24 MHz
- LINAC event rate is 114.24 MHz
- Relation between event rate & reference clock for Micrel SY87739L is

Event rate (MHz) = [(M/N) x {P - ($Q_{(p-1)}$ / (Q_p + $Q_{(p-1)}$)} x Fref] / PostDivSel

where

Fref = 24.0 MHz

PostDivSel = 6

M = 14, N = 14, therefore M/N = 1

P = Mod[(Event rate X postDivSel)/Fref] = 29

$Q_{(p-1)}$ = 14

Q_p = 32 – $Q_{(p-1)}$ = 18, as Q_p + $Q_{(p-1)}$ = 32

the bit patteren of configuration word

0000-Q_p(5)-$Q_{(p-1)}$(5)-P(4)-000-PostDivSel(5)-N(3)-M(3)

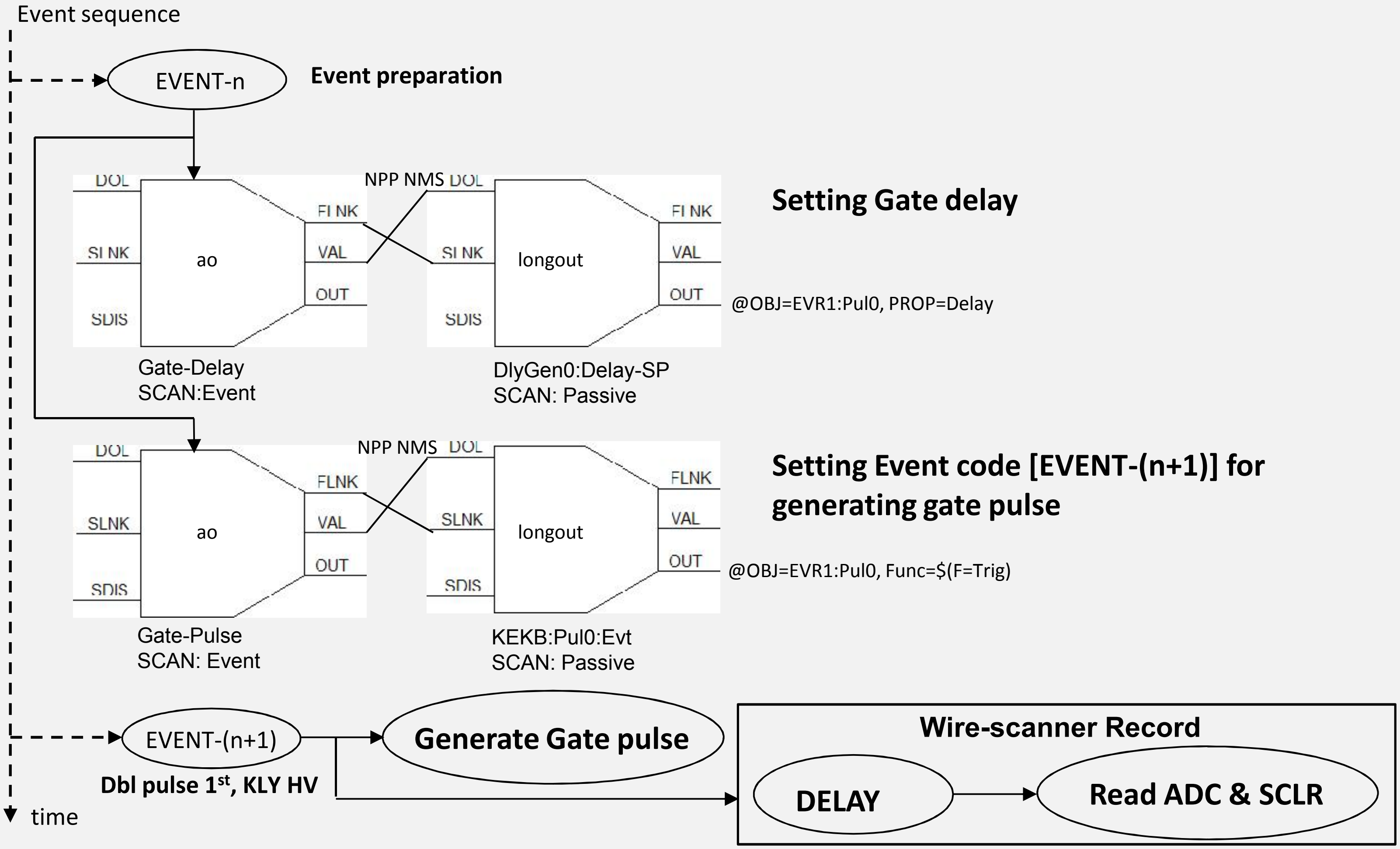
Hence the configuration word for 114.24 MHz is

093B01AD (0000-10010-01110-1100-000-00110-101-101)

- The configuration word to be stored in EVR non-volatile memory 10baseT network interface

Data acquisition strategy

- **Requirement:** configuration of pulse generator (i.e. delay & timing event) for each beam mode to generate ADC Gate synchronized with beam pulse
- **Solution:** Utilize two consecutive timing events for pulse generator configuration and data acquition



Wire-scanner (WS) Record

- **Why a new record?**

  To keep the interface (software) to Wire-scanner User interface (SAD panel) UNCHANGED,

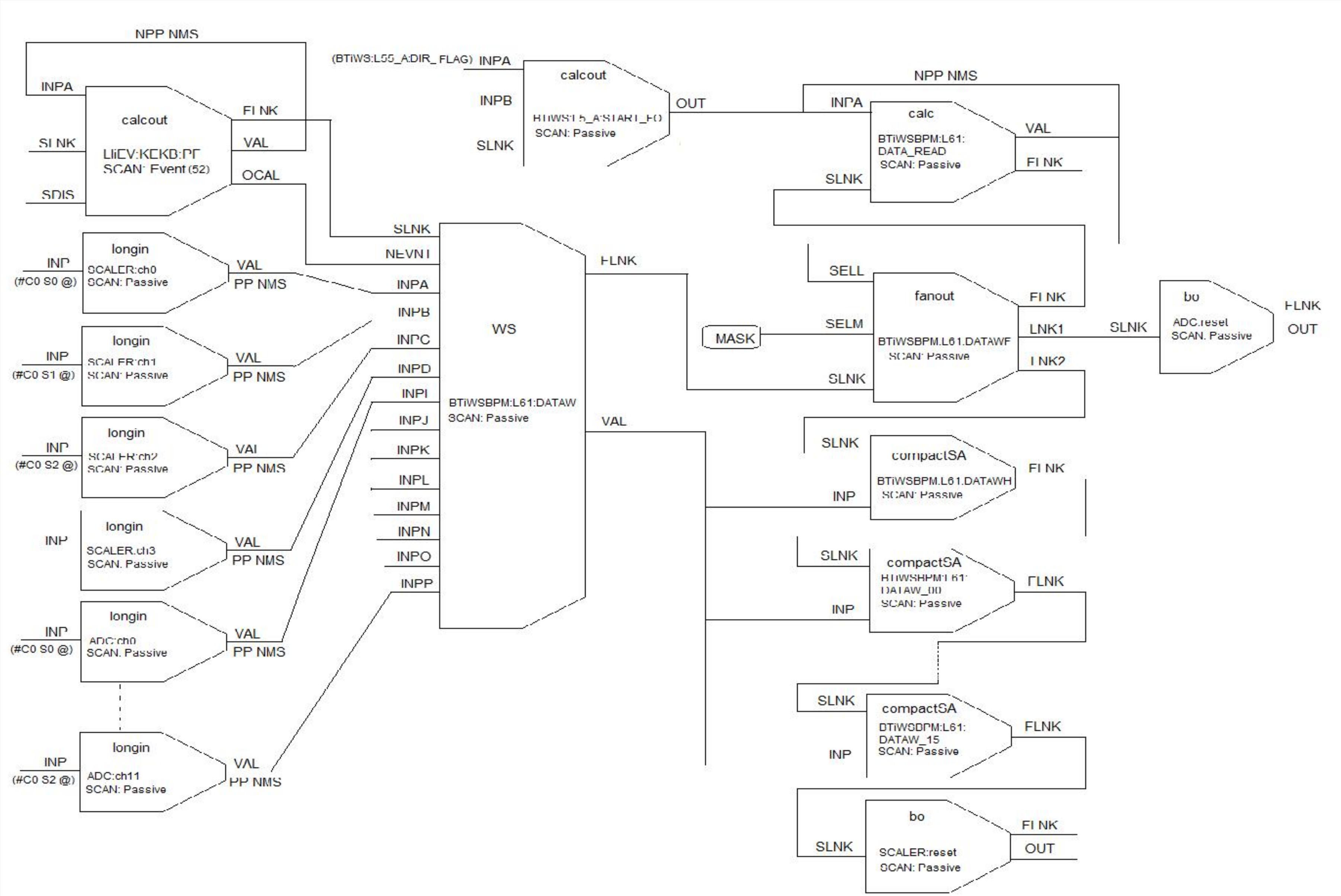
  To minimize impact on OTHER proven PROCESSES

- **Features**
  - ✓ Application specific – suit the purpose of wire scanner system
  - ✓ A waveform record with multiple input links (26, INPA….INPZ) for collecting data
  - ✓ A Ring Buffer, appending an array of data (from input links) on every scan
  - ✓ Provision for delay the processing – to ensure completion of ADC conversion (if LAM is absent!)
  - ✓ Fields for defining number of SCALER, BPM (4 ADC per BPM) and wire-scanner ADC channel
  - ✓ Field for appending BEAM mode (event code) to data on every scan for identification
  - ✓ Option for resetting the buffer
  - ✓ Option for calibrating BPM signal using calibration data (Not yet tested!)
  - ✓ Field for defining calibration data file path

- ✓ **Constraint**

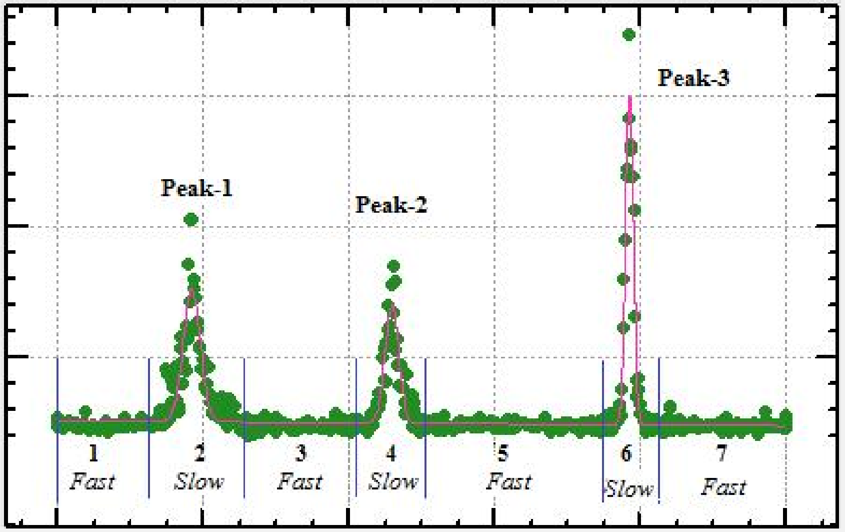
  SCALER, BPM and ADC data sources (pv links) should be defined in sequence at the input links (i.e. from INPA…..) according to the respective numbers

Record link

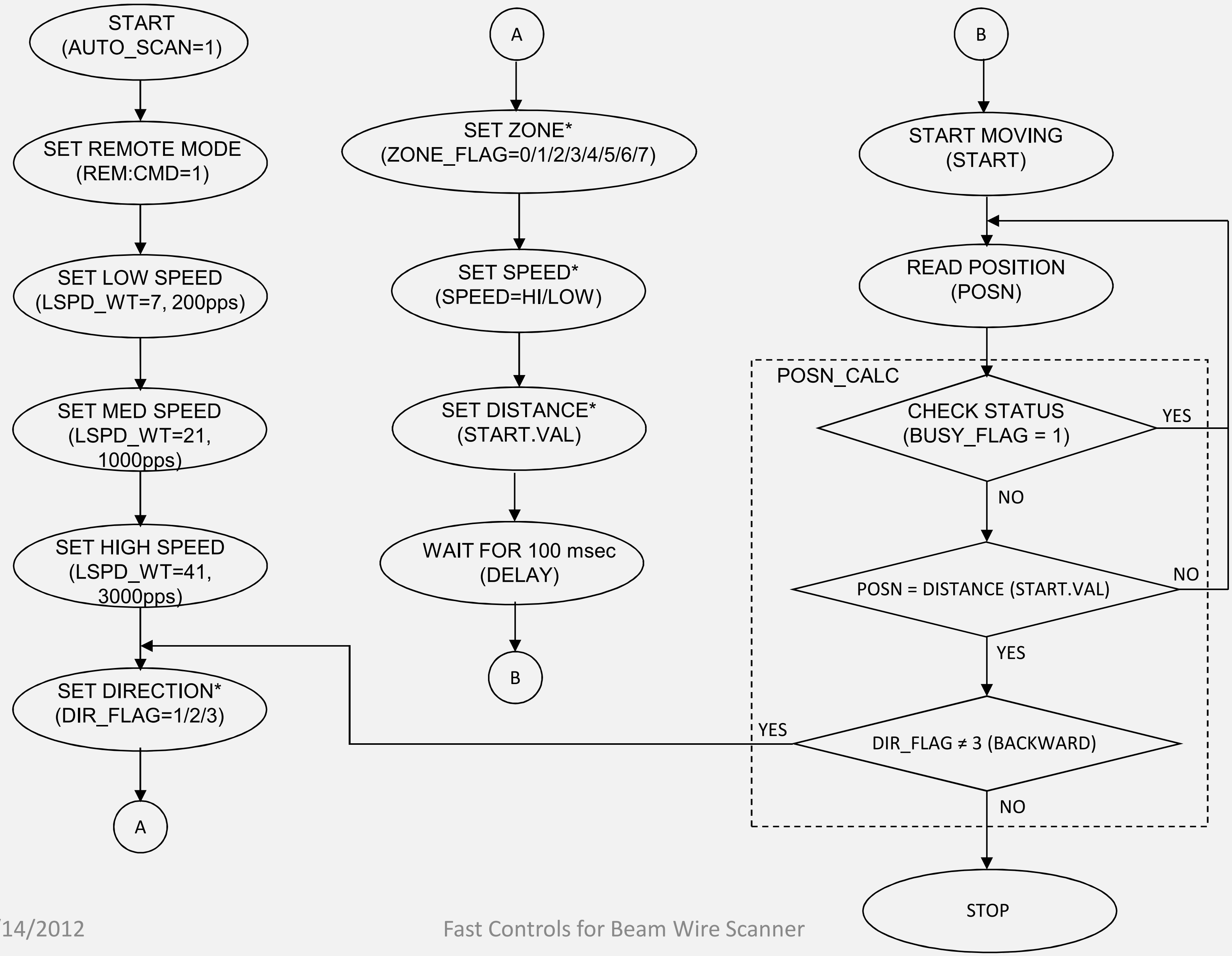

NPP NMS
INPA
calcout
FLNK
SLNK
VAL
SCAN: Event (52)
OCAL
SDIS
(BTiWS:L55_A:DIR_FLAG) INPA
calcout
INPB
SCAN: Passive
SLNK
OUT
NPP NMS
INPA
calc
BTiWSBPM:L61:
DATA_READ
SCAN: Passive
VAL
FLNK
SLNK
longin
INP
(#C0 S0 @)
SCALER:ch0
SCAN: Passive
VAL
PP NMS
longin
INP
(#C0 S1 @)
SCAN: Passive
VAL
PP NMS
longin
INP
(#C0 S2 @)
SCAN: Passive
VAL
PP NMS
longin
INP
SCALER:ch3
SCAN: Passive
VAL
PP NMS
longin
INP
(#C0 S0 @)
ADC:ch0
SCAN: Passive
VAL
PP NMS
longin
INP
(#C0 S2 @)
ADC:ch11
SCAN: Passive
VAL
PP NMS
SLNK
INPA
INPB
INPC
INPD
INPI
INPJ
INPK
INPL
INPM
INPN
INPO
INPP
WS
BTiWSBPM:L61:DATAW
SCAN: Passive
FLNK
VAL
SELL
SELM
MASK
SLNK
fanout
BTiWSBPM.L61.DATAWF
FLNK
LNK1
SLNK
bo
ADC.reset
SCAN. Passive
FLNK
OUT
SLNK
compactSA
BTiWSBPM.L61.DATAWH
INP
FLNK
SLNK
compactSA
INP
SCAN: Passive
FLNK
SLNK
compactSA
DATAW_15
INP
SCAN: Passive
FLNK
bo
SLNK
SCALER:reset
SCAN: Passive
FLNK
OUT

Wire-scanner movement

- Driven by Pulse motor (4µm/pulse) – maximum distance of 100mm
- Controlled by 4 channel Pulse Motor Controller with GPIB interface and
  - Provisions for Mechanical and Logical Limit to restrict over drive
  - Pulse feedback while moving forward direction only
  - Options for HIGH / MEDIUM / LOW speed movement
  - Options for Relative / Absolute movement
- **Why Multi-speed scan?**
  - X, Y & U wire interact with beam at three distinct regions of whole span
  - Slow speed scanning at regions of interactions results into better beam profile
  - Higher speed (Fast) at other regions results into optimum scan time
- **Implementation**
  - Divide span of movement into seven regions (as shown in Fig)
  - Slow speed regions are defined by peak position and width around the peak
  - Defining HIGH & LOW speed values
  - Absolute scan, to restrict over drive

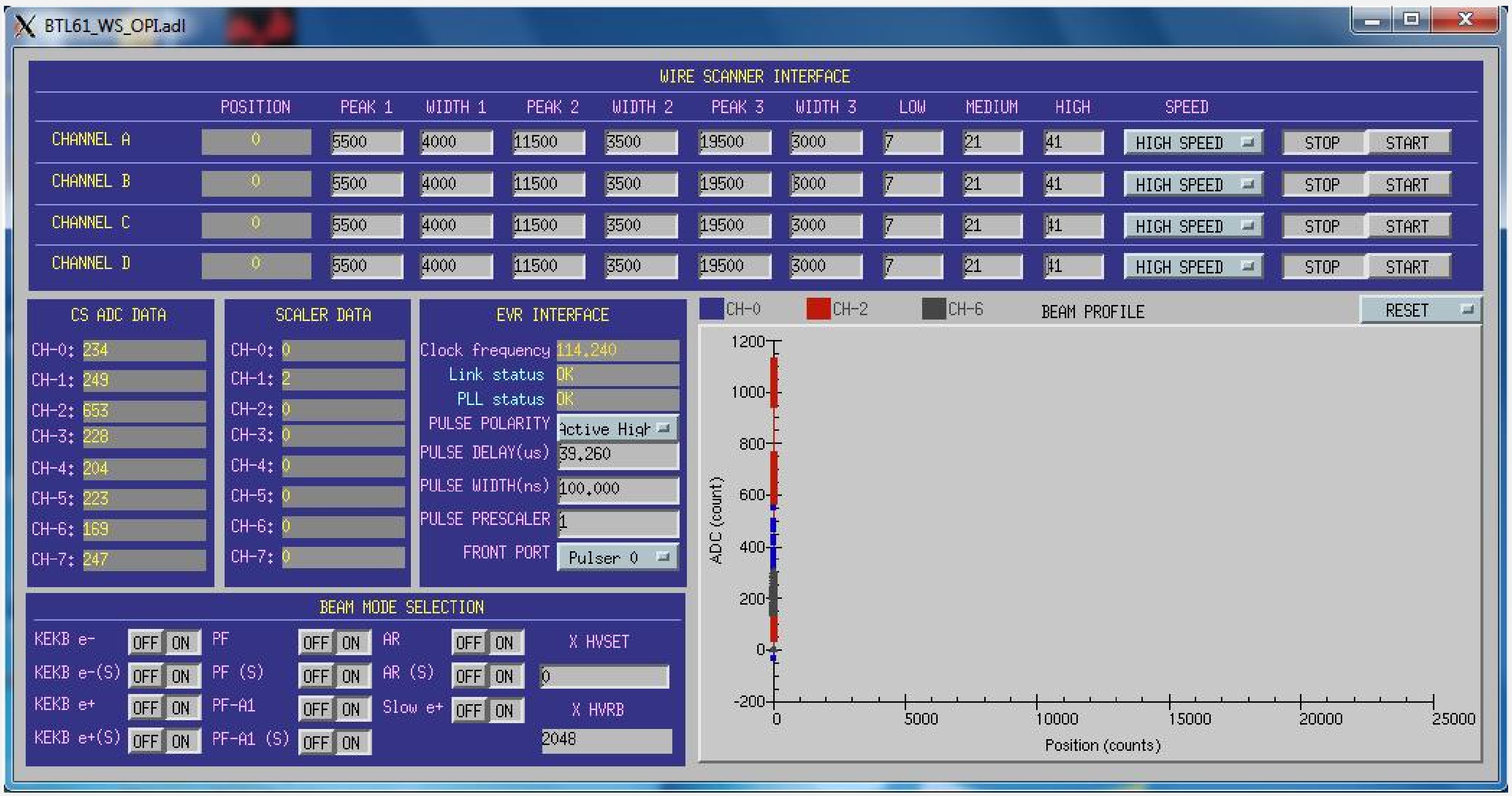


Wire-scanner movement (Flow-chart)



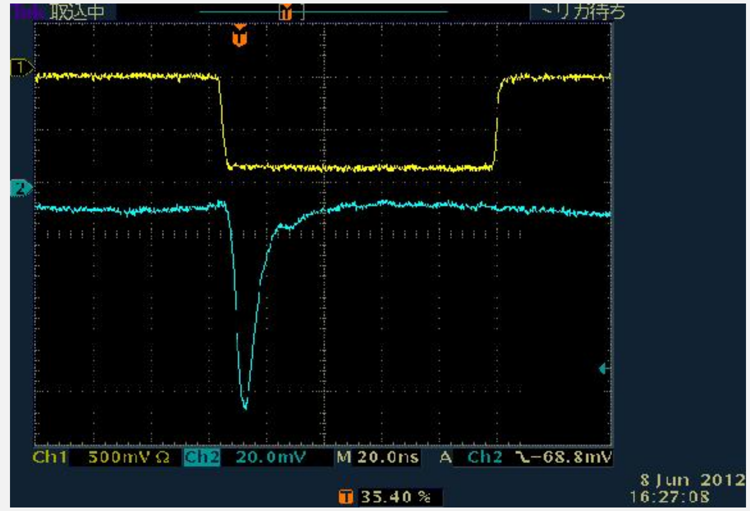
User interface (MEDM)

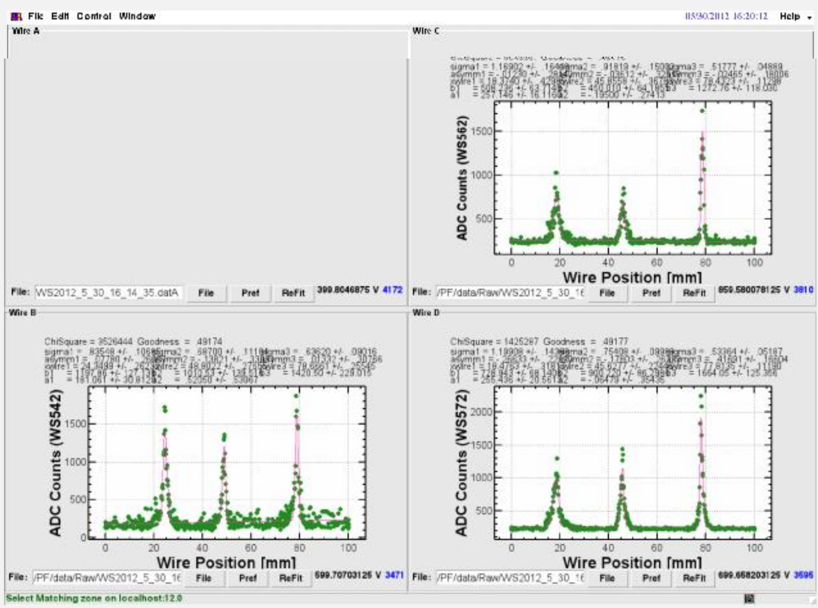
- To control and monitor the data acquisition process, built using MEDM
- Provisions provided for
  - Defining peak positions and corresponding width around them
  - Specifying HIGH / MEDIUM / LOW speed index
  - Specifying EVR output, Pulse delay, pulse width, polarity etc. for particular BEAM mode
  - Selecting BEAM modes

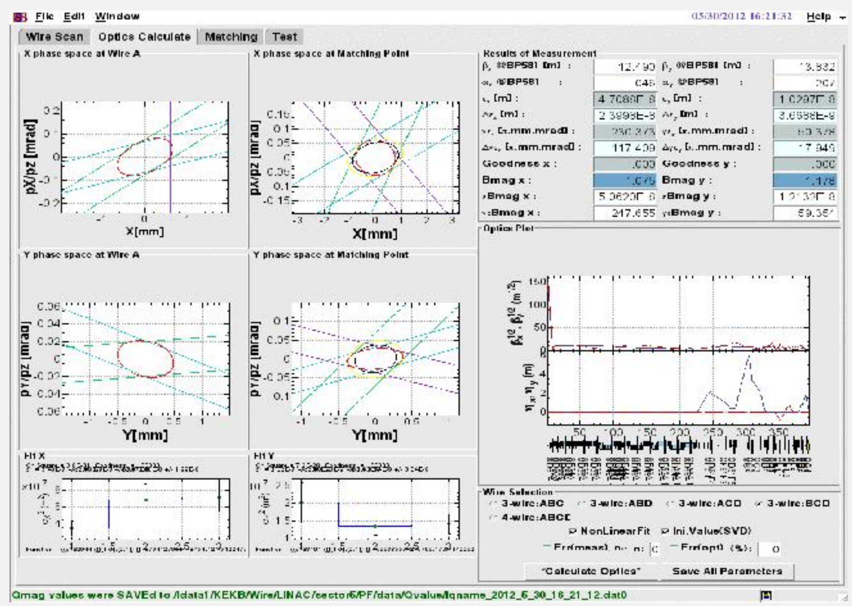


Test Results

As obtained, seems to be satisfactory!







Sincere Regards to:

Kazuro Furukawa

Tatsuro Nakamura

Naoko Iida

Special Thanks to:

Tomohiro Okazaki

Shiro Kusano

Thank you!