Accelerator Controls

December 3, 2012

Old Dominion University

Outline

- Brief Introduction
- Components
- Requirements
- Control System Architecture
- EPICS at Jefferson Lab
- System Examples
- Conclusion

Brief Introduction

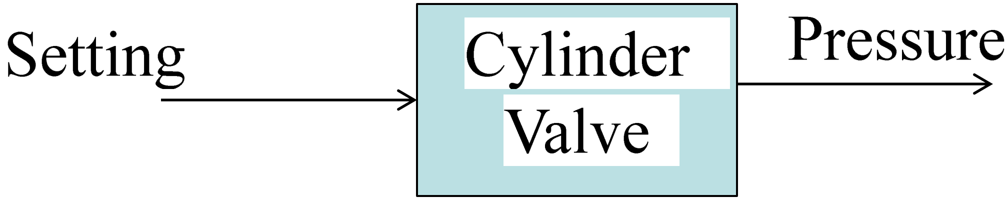
- Purpose of Control Systems
  - Setup, monitor and manipulate
  - Supplement, reduce or eliminate human factors (slow, lapse of attention, ..)
- Daily Examples
  - Biological Organisms (mostly autonomous)
  - Automobiles (semi-autonomous)

Open Loop

- Open Loop
  - System is setup and no intervention
    - Example: Gas cylinder for purge – you set it up for a certain pressure/flow rate

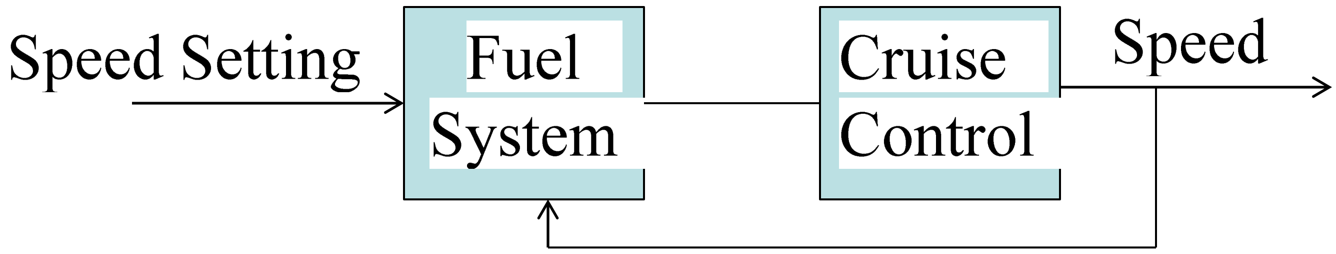
    -Q. How do you know the pressure or flow rate?

    - A. A gauge or a set of calibration curves



Closed Loop

- Closed Loop (Feedback)
  - System is setup and regulated for a desired result
    - Example: Cruise control on your car



- Note that in the open loop example, if the pressure in the gas cylinder falls, you can open the valve (changing the open loop to feedback loop)

Loops

- The relationship between the output and input of the system is its transfer function
  - Size relationship
  - Timing Relationship
- Example: A sinusoidally varying input A ($\sin\vartheta t + \phi$) with a sinusoidally varying output B ($\sin\vartheta t + \psi$)
  - Size gain is B/A
  - Phase difference is $\psi - \phi$
- The amplitude gain and the phase difference at a given frequency – Frequency Response

PID

- For an error, which is the difference between the desired value and the measured value
- Proportional Feedback
  - The applied correction is proportional to the error
- Integral feedback
  - The applied correction is an integral of accumulated errors over some time
- Derivative Feedback
  - The applied correction depends on the rate of change of the error

Components

- Two much used components
  - Analog to Digital Converters (ADCs)
  - Digital to Analog Converters (DACs)
- Modern Controllers
  - Microprocessors
- Interconnects
  - Serial (e.g. Ethernet)
  - Parallel (e.g. VME Bus)

ADCs

- Simple ADC would be
  – Charge a capacitor
  – Discharge the capacitor through a resistor
  – Start a counter at the beginning of the discharge
  – Record the number of counts when the capacitor is completely discharged

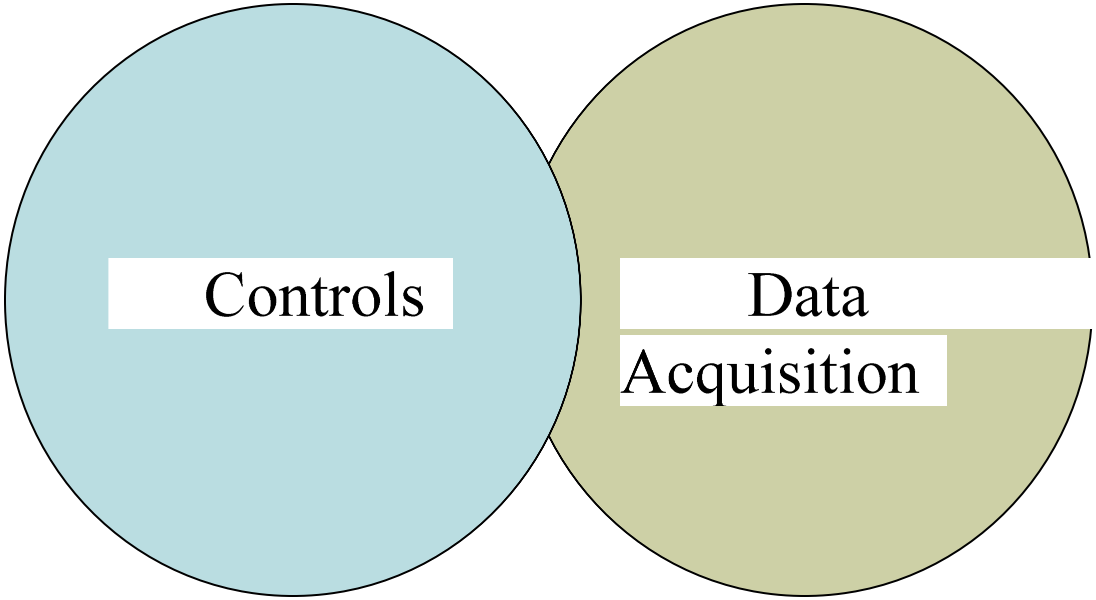
There are many kinds of ADCs one can buy depending on the application (speed of conversion, number of bits, linearity, etc.)

DACs

- A simple DAC would be
  - Connect a bit to a known voltage
  - Convert the voltage to current through a resistor
  - If the MSB is connected to a resistor of R Ω, the next significant bit will be connected to a resistor of 2R Ω and the next one 4R Ω etc.
  - Sum the currents for analog signal

Controls/DAQ

- In Nuclear/Particle Physics experiments, one distinguishes between 'slow' controls and 'fast' data acquisition.
- However, there is overlap between these two systems



Controls/DAQ

- 'Slow' controls examples
  - Setting up voltages for photomultiplier tubes
  - Monitoring the gas flow through wire chambers
- Data Acquisition example
  - Reading out all the detectors on a trigger
- Overlap examples
  - Monitoring the wire hits (snoop on the DAQ)
  - Record the beam positions (snoop on controls)

Features of Acclerator Control Systems

- The software that allows us to talk to the devices such as ADCs, DACs, stepper motor controllers etc. comes under Low Level Applications
- The software that uses this data, for example, to control the beam orbit comes under High Level Applications.
- All user driven software is High Level Apps and depends on information from Low Level Apps

Accelerator Control System Requirements

- We want the system to have a long life time
  - Modular
  - upgradeable
  - Scalable
  - Accommodate newer technologies
  - Implement a range of timing requirements
    - Some information can be obtained leisurely
    - Some information needs to be obtained at a greater frequency

Accelerator Control System Requirements

- Reliable
  - Machine operates 24/7 for experiments
  - Need almost 100% uptime
  - Barring that, control system failure should not shutdown the machine
- Open Standards wherever possible
  - Less expensive
  - Larger user base

Accelerator Control System Requirements

- Accommodate many types of users who have different roles
  - Accelerator Operators
    - Friendly Interface
    - Alarms they can configure
    - Backing up and restoring machine configuration
    - Log books, ….

Accelerator Control System Requirements

- Accelerator Physicists
  - Data Acquisition
  - High Level Apps
  - Data storage for offline analysis
- Engineers and Technicians
  - Testing and commissioning devices
  - Diagnosing device failures
  - Archival and retrieval for troubleshooting
- Experimenters

Requirements

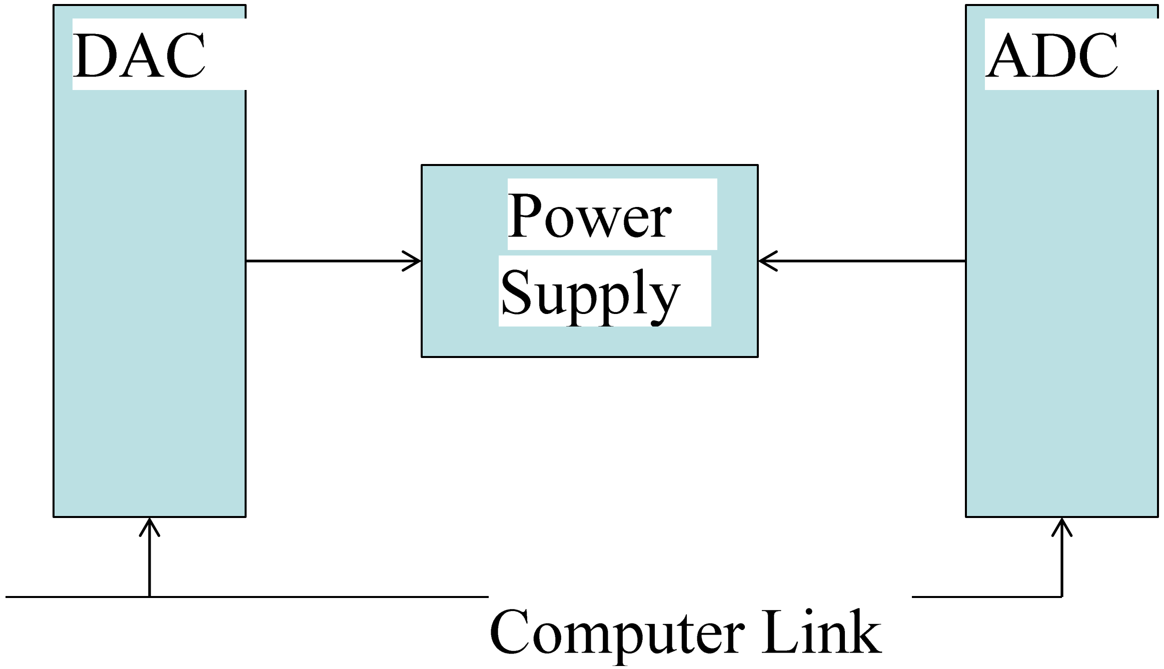
- Additionally,
  - Code Maintenance
  - Security

Control System Hardware Architecture

- Putting the components together
  - Need to communicate with the components
    - Computer control
    - Connection schemes
- Example: Controlling a power supply
  - Send a value (binary) to a DAC
  - Read back a value (binary) from an ADC

Example Diagram

- Modern power supplies contain DACs and ADCs



Multitudes

- With only a few components, one can imagine simple serial communications (a la keyboard)
- As the number of components to control increases, one needs more sophisticated systems
- Enter bus based systems

Busses

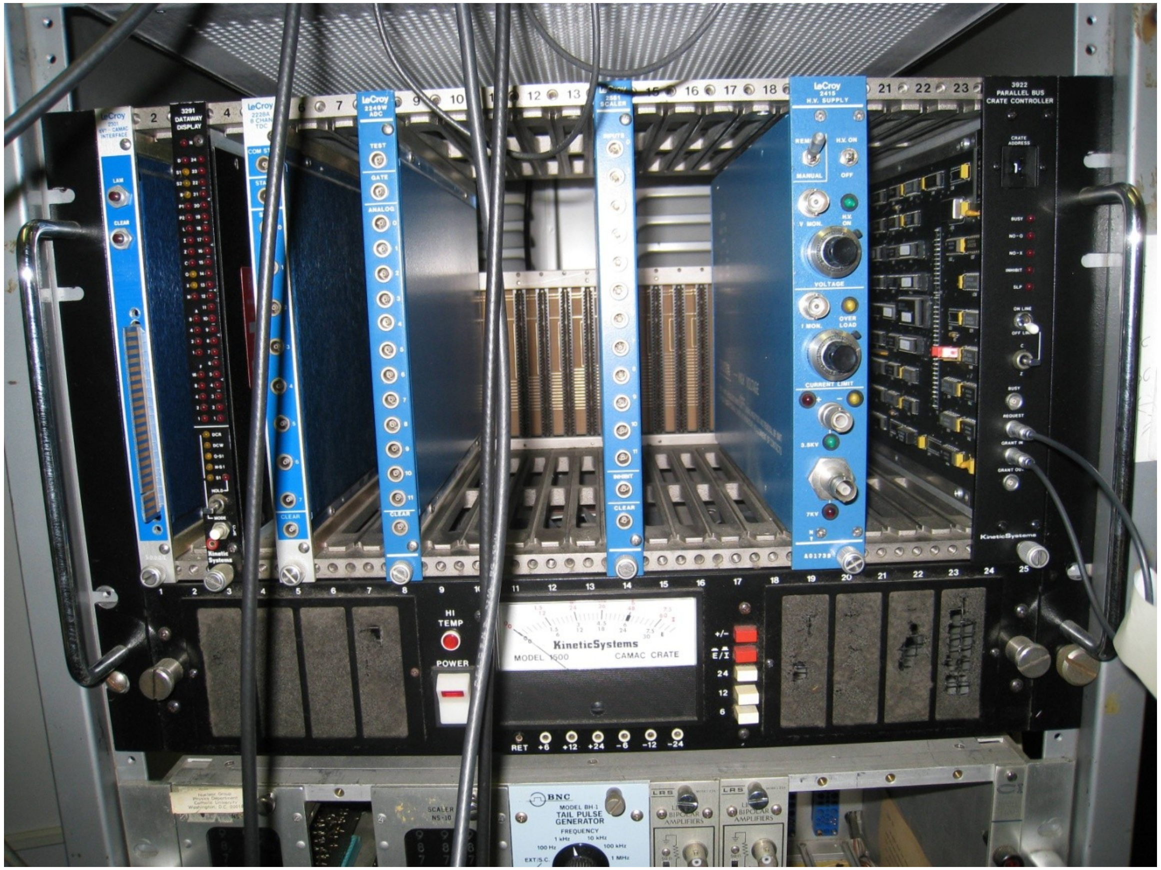
- CAMAC
- VME
- PC104
- Others

A good control system should be able to accommodate a variety of systems and communication schemes

CAMAC

- CAMAC is a venerable standard which grew out of the need to clean up counting houses in Nuclear and Particle Physics experiments
  - This is because each electronic module was individually interfaced to a readout (computer) system

CAMAC crate



LeCroy
3291
DATAWAY
DISPLAY
2249W
ADC
2415
H.V. SUPPLY
3922
PARALLEL BUS
CRATE CONTROLLER
HI
TEMP
POWER
KineticSystems
MODEL 1500
CAMAC CRATE
RET +6 +12 +24 -6 -12 -24
BNC
MODEL BH-1
TAIL PULSE
GENERATOR

CAMAC Features

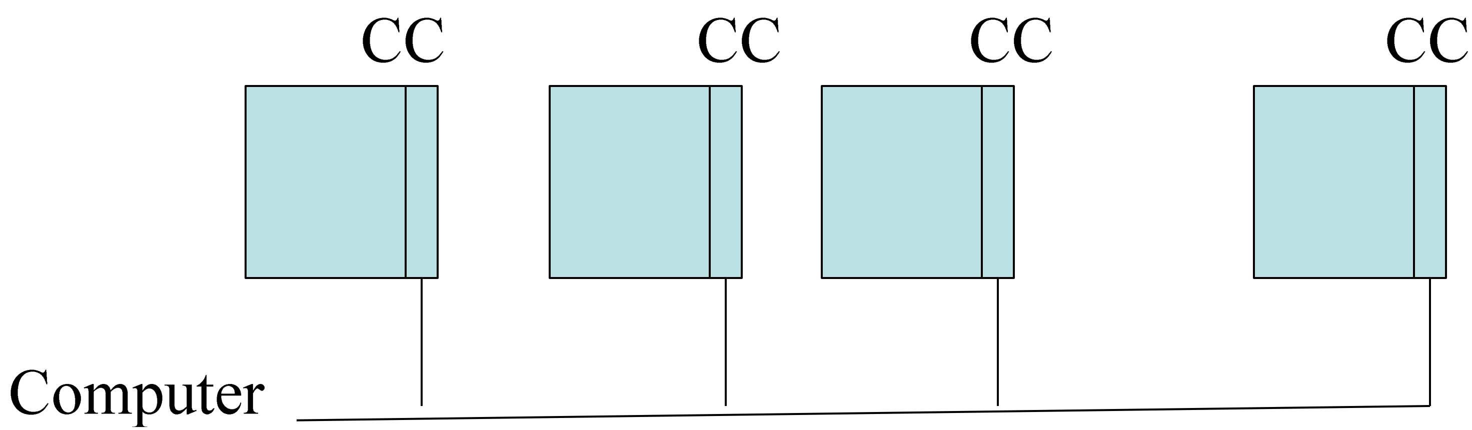
- 25 slots to house modules
- Slot 25 is special – reserved for a crate controller (master)
- Each slot has a number assigned to it (wires on the backplane)
- Only the crate controller can access the modules
  - Does this by pulling on the station number wires
  - Gives commands called F codes
    - For example F=0 is a read command, F=16 is a write command
  - Within the module, sub-address is accessed by A codes

CAMAC Features

- What is a sub-address
  - With integration, in good old days, one could have 16 channel ADCs (for example)
  - Each ADC channel can be readout by pulling on the A wires
- Thus a CAMAC command from the crate controller is
  - FNA (Function, Station, Sub-address)

CAMAC Features

- What happens if you need more than one crate?
  - Develop interconnects (FCNA commands)



CAMAC Serial or Parallel Highway

CAMAC Limitations

- Slow (1 microsecond for readout)
- Dataway (the number of wires on the backplane to transfer data) was limited to 24 bits
- Implementation of interrupts was inadequate
  - Interrupts are needed when some important event occurs
- Single master
- Cost

There still is a large number of CAMAC installations

(Not so) Modern Busses

- VME bus
  - Started life as an extension to Motorola's microprocessor busses
  - Multi-master
  - Priority Centralized Arbitration
    - Arbiter has to be in a particular slot in the crate
  - 16-bit data path, increased to 32 bit, now to 64 bits
  - Address range is also extended to 64 bits
  - Many extensions over the years

VME Crate

VME vs. CAMAC

- Connectors
  - Pins vs. edge connector (better contact, accidental hot insertion will not kill the components)
  - Crates are less expensive
  - Large address and data paths
  - Interrupt handling capabilities
  - Still has one limitation – the bus arbiter has to be in the first slot
    - Distributed arbitration - Fastbus

Communications

- While one can design specialized parallel communications (as was done at Fermilab for DAQ, the advent of high speed ethernet makes it the interconnect of choice
- Parallel communication between crates is very similar to communications within the crates
  - Each crate has a crate controller who can be a master on the interconnect bus.
  - Has an address on the bus
  - In simple implementations, the crate controller is the only master on the local VMEbus

Serial Communications

- Of the many types of serial interconnects, such as FDDI, ethernet is the most popular
- Starting at 10 Mbit/s, the speed has grown to 1Gb/s and up
- Multiple nodes on a wire
  - Uses Carrier Sense Multiple Access/Collision Detection scheme
  - When multiple nodes put signals on the wire at the same time, the nodes detect the collision and wait for a random time before trying to communicate again
  - Data integrity is maintained by checksum

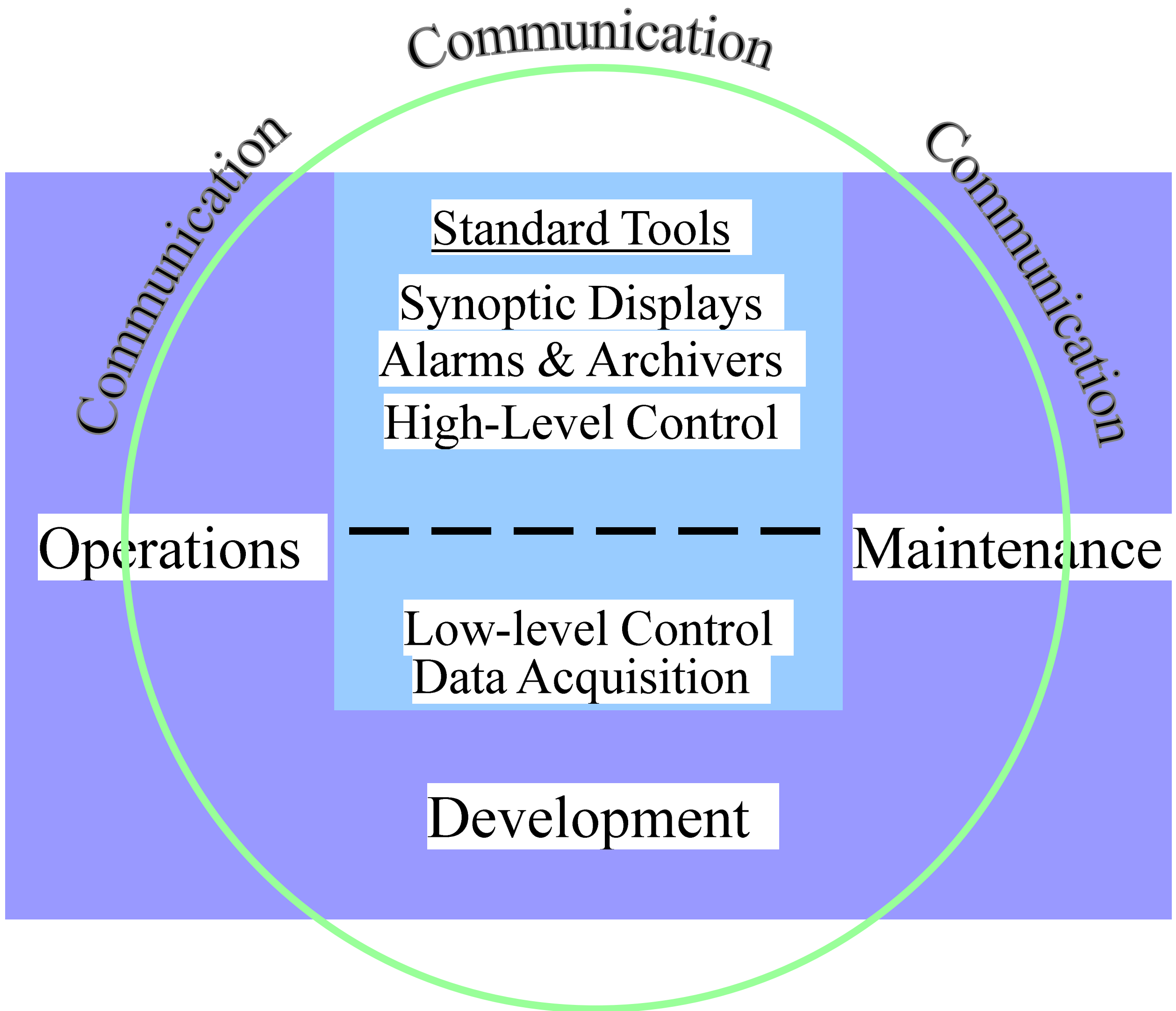
Accelerator Controls at a Glance

- Accelerator control system is more than a set of low-level applications and high-level user applications.
- It includes
  - a software development environment for users to create applications
  - tools to backup and restore machine status
  - integrates supplemental systems such as
    - safety systems that protect the machine components and assure personnel safety
    - timing systems that are used to acquire data
    - Alarms

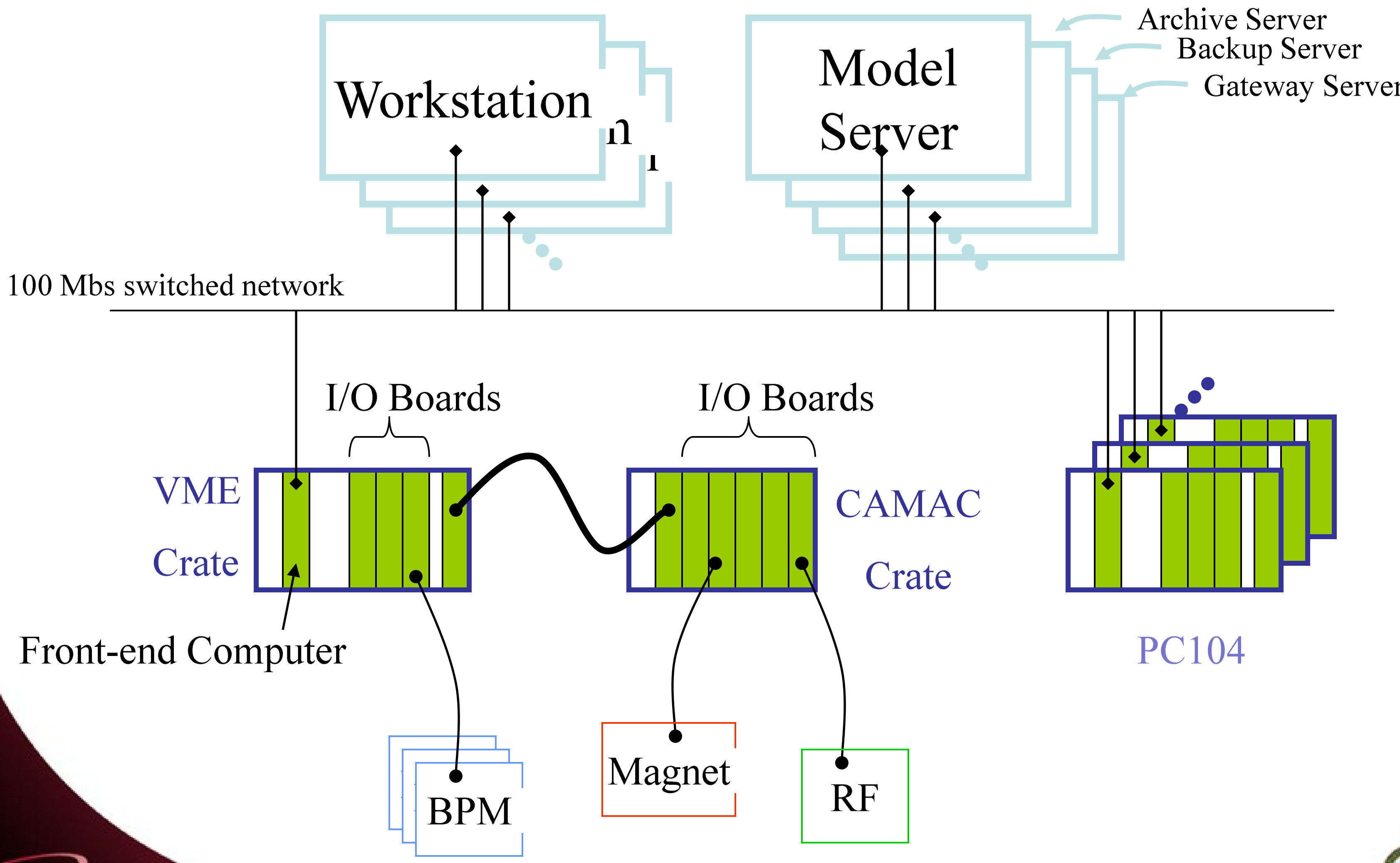
Accelerator Controls at a Glance

- It integrates supplemental systems such as
  - Safety systems that protect the machine components and assure personnel safety
  - Timing systems that are used to acquire data

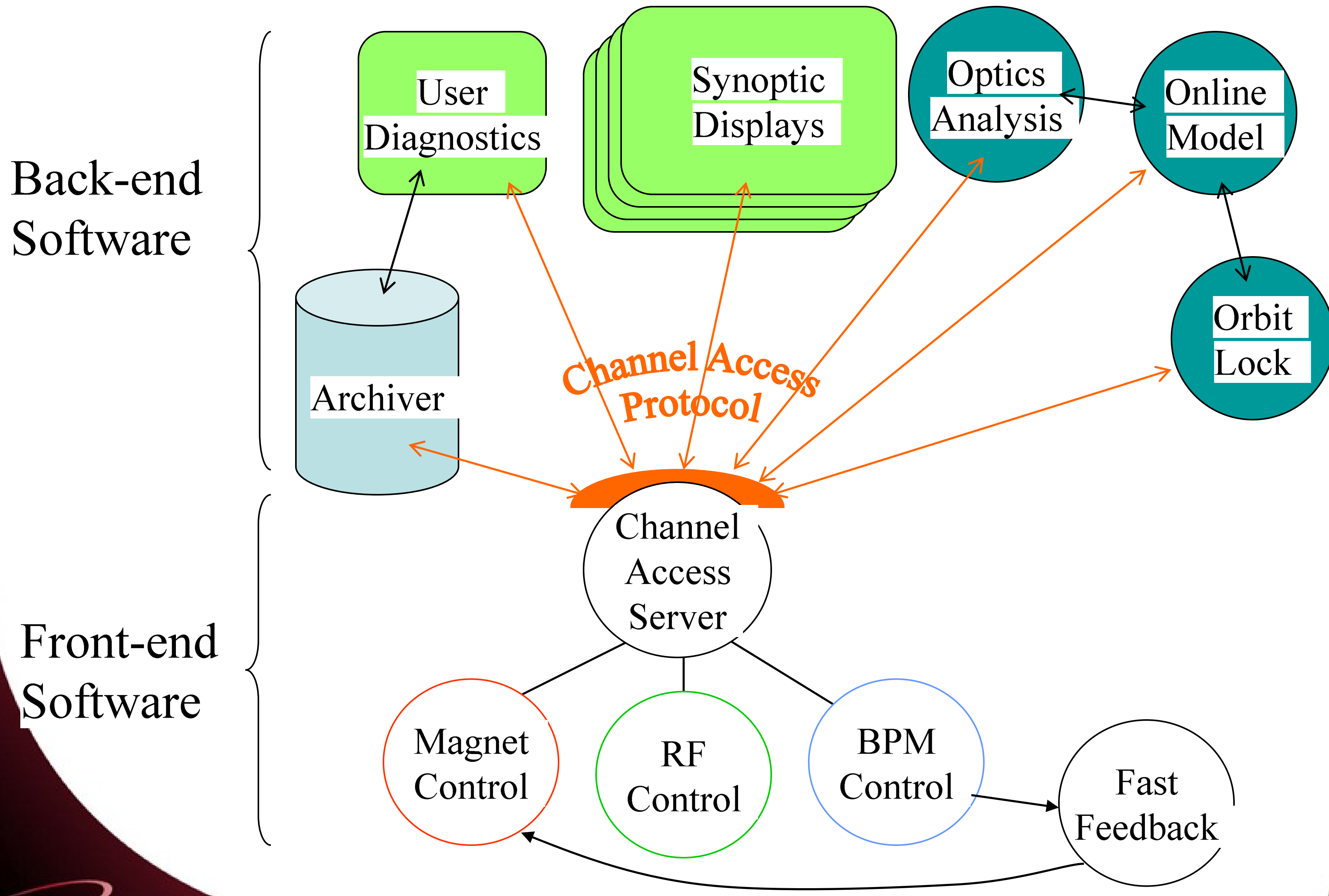
Control System



Control System Hardware Schematic



Control System Software Schematic



EPICS

- The Control Systems at Jefferson Lab are based on the Experimental and Industrial Control System known as EPICS.
- EPICS is a toolkit of software tools and applications that provide a software infrastructure for the development of distributed control systems.
- Developed by Los Alamos National Laboratory and Argonne National Laboratory
- Now an international collaboration with over 90 sites.

EPICS Features

- As illustrated in the figure, the basic components are
  - A user interface (a UNIX workstation)
  - An Input Output Controller (IOC), based on industry standard microprocessor boards, which access the control points
  - A Local Area Network (LAN) that facilitates communications between users and the IOCs
  - Uses a client/server model to provide communications between computers distributed around the site.

EPICS Features

- A communication protocol called Channel Access provides network transparent communication between a client and an arbitrary number of servers
- The servers, or Input/Output Controllers (IOCs) perform I/O and device control
- EPICS is designed to be event driven.
  - Polling for information is very inefficient and often unnecessary
  - Only communicating on a change results in efficient use of resources as well as to quick response times

EPICS at Jefferson Lab

- At Jefferson Lab, EPICS is used for control and monitoring of the
  - both Accelerators (Nuclear Physics and FEL)
  - the Experimental Halls

- The EPICS configuration for the Accelerator contains over 140 IOCs, roughly 80 UNIX hosts, and over 65,000 I/O control point, and over 250,000 EPICS records.

EPICS at Jefferson Lab

- Workstations for high level control, data archiving, retrieval and visualization and operator interfaces reside in the control room.
- People who are not accelerator operators can view control system information elsewhere on-site, but not manipulate the process variables.

EPICS at Jefferson Lab

- Many different flavors of iocs
  - Motorola/Power PC VME based single board iocs that run the Vxworks operating system.
  - PC-104 based iocs that utilizes RTEMS (Real-Time Executive Multiprocessor Systems)for its operating system.
  - IOCs that run on LINUX work stations.

System Examples – Fast Feedback

- The fast feedback system reduces fluctuations in beam position and energy at the hall A and C targets at power line harmonics
- It uses two processors
  - The first processor services EPICS requests
  - The second runs the feedback system algorithm and manages all hardware.
  - The first processor controls the feedback processor via 16 MB of shared memory on the second processor which can be accessed by both computers.

System Examples – Fast Feedback

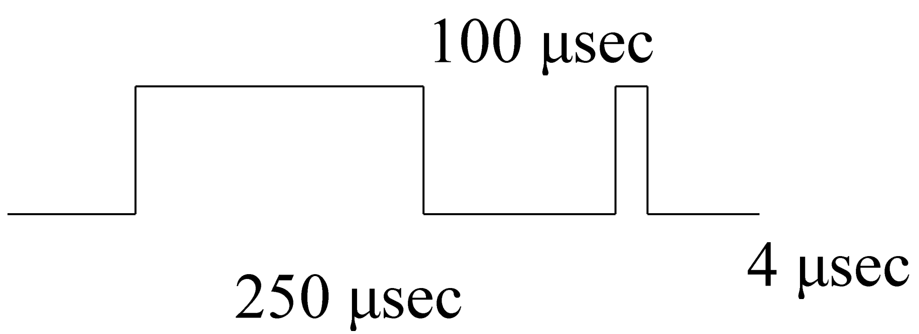
- Uses a number of BPMs and air core magnets in the hall beam line
- Vernier controls RF power to SRF cavities
- Operates at 2 KHz
- Uses the slow feedback locks to calculate corrections to be applied
- These corrections are fed forward to suppress AC power harmonics

System Examples –Supplementary Systems (Safety)

- There are two critical stand alone systems that interface to the control system for obvious reasons
  - Machine Protection System
    - An errant component that can deflect the beam and damage the accelerator
      - Need to identify the component and take action
  - Personnel Safety System
    - Protects personnel from radiation
      - Accidental or emergency access to accelerator tunnel or experimental halls

System Examples - Supplementary Systems (Timing)

- Beam Sync is used for
  - Tuning
  - Data Acquisition



System Examples - Supplementary Systems (Timing)

- Ring Network connects selected IOCS crates
  - Used for special purpose data acquisition
  - Message passing
  - Sends specific tasks to specific crates

Conclusion

- Accelerator Control Systems are very complex
  - Need a lot of resources
    - Equipment
    - Talent
- Future (some of it is here)
  - Artificial Intelligence
    - Fuzzy Logic
    - Neural Networks

Fuzzy Logic

- Fuzzy Logic is closer to human thinking and natural language
- It tries to capture the inexact way the real world works
- First obtain values and fuzzify the values
  - The measurement belongs to a membership function
  - Choice of membership function depends on the desired outcome
  - A set of rules are applied
  - The result is defuzzified to give a precise output