Proposal To Implement Data Stream Consolidation In The Fanout Concentrators

Eric Björklund
Los Alamos Neutron Science Center

(LA-UR-13-25772)

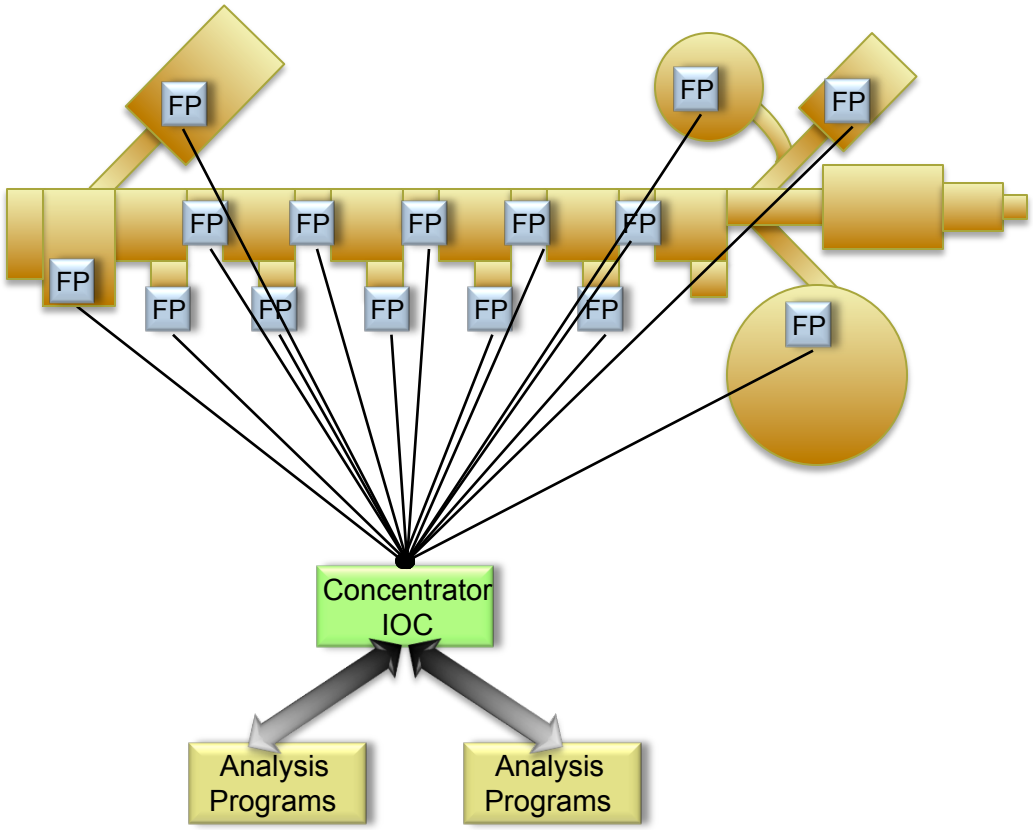
The Application

The LANSCE Fast Protect System

The LANSCE Fast Protect System

- **The Fast Protect System monitors various devices along the accelerator and beam lines, looking for indications that the beam is not going where it is supposed to go.**
  - Radiation monitors
  - Transmission loss monitors
  - etc.
- **When a problem is discovered, the Fast Protect System shuts the beam off within 10 microseconds (this is a hardware-only operation).**
- **Message is sent to operator consoles detailing what happened and where.**
- **System is efficient enough that at low beam intensities the Fast Protect reporting system is used as a beam tuning diagnostic.**

The LANSCE Fast Protect System



- **15 FP monitoring stations at strategic locations along the accelerator.**
- **When a Fast Protect occurs, data from each FP station is sent to a Concentrator IOC.**
- **Concentrator IOC collects all the FP data and makes it available to analysis programs.**

The Timing System's Role

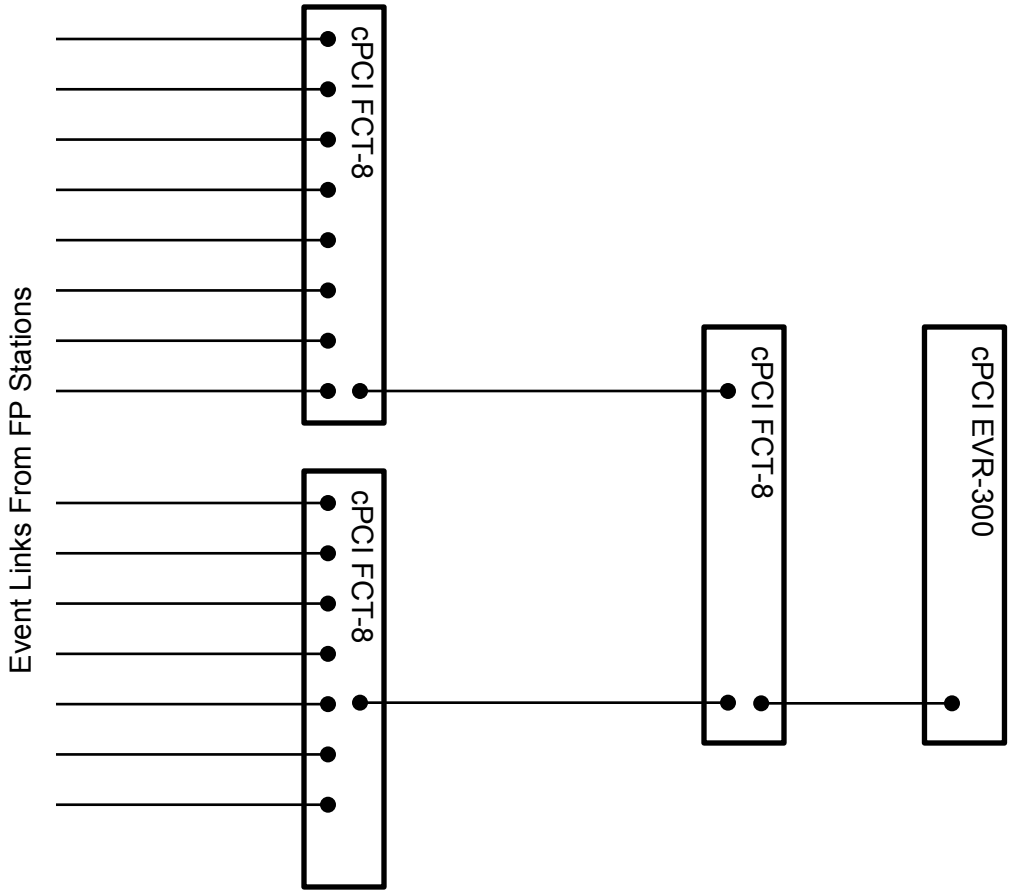
- **Each Fast Protect station needs timing gates.**
  - Only Fast Protect when there is beam.
- **Therefore, each Fast Protect station has an EVR.**
- **Fast Protect signal is monitored by the Master EVG.**
- **Fast Protect signal generates a "Fast Protect Event".**
  - Tells application programs that their data may be suspect.
  - Tells FP monitoring stations to send data to the collector.

The Problem:

How To Send Data Back To The Collector IOC

- **Could use regular network backbone.**
  - EPICS Channel Access Protocol
  - TCP/IP, RPC, etc.
- **Since the FP stations already have an EVR, they could use the data buffer feature to send data back on the event link return path.**
  - Faster and more deterministic than ethernet.
  - Guarantees all data came from the same machine cycle.
- **Not much data needs to be transferred.**
  - ~40 bytes per FP station

Proposed Architecture



The Tricky Part

Make Sure The FP Data Packets Don't Collide

- **Each FP station must time the transmission of its data packet so that the data packets do not collide at the concentrator or overwrite the data stream buffer in the EVR before they can be processed.**
- **Packet transmission time ≈ 398 nSec**
  - 40 bytes/packet @ 100.625 MHz Event Clock
- **Packet transmission line "jitter" ≈ 5 µSec**
  - Distance to furthest FP station ≈ 1 Km
- **Packet processing time ≈ 10 µSec**
- **Interrupt jitter (at the FP station) ≈ 60 µSec**
- **Maximum interrupt latency (at the Concentrator) ≈ 75 µSec**
- **Time window for FP data packet ≈ 150.4 µSec**
- **Total time to receive 15 packets ≈ 2.256 mSec**

The Tricky Part

Make Sure The FP Data Packets Don't Collide

- **Total time to receive 15 packets ≈ 2.256 mSec**
- **Time budget for processing all FP data packets is 4 mSec, so this would work…**

However…

- **If the fanout concentrator modules could buffer up the data streams from each of the incoming links and merge them into a single data stream…**
  - Plenty of room for all the data (40 bytes X 15 stations = 600 bytes total)
  - All FP stations can send their data at the same time (no need to "tune" the transmission times)
  - Total time to read data from all 15 stations is ≈ 150 µSec (not counting transmission delays, interrupt latencies, etc.)
  - Easier to add additional FP stations.

But..

How Does The Fanout Concentrator Know When It Is Done?

- **Not all the FP stations may be operational.**
- **Still want data from the operational stations.**
- **Tuning for coincidence is even harder than tuning for non-coincidence:**
  - Packet width is 0.398 µSec.
  - Transmission line length "jitter" is ≈ 5 µSec.
  - Interrupt jitter at FP station is ≈ 60 µSec.
- **Suggestion:**
  - Start a timer (e.g. ≈ 70 µSec in our case) when the first data stream byte arrives.
  - If another data stream arrives before the timer times out, restart the timer.
  - If no new data streams arrive before the timer times out, send the accumulated buffer.

But..

What If There Is Too Much Data?

- **If everything is working correctly, this should not happen. However…**
- **Suggestion:**
  - If more than 2048 bytes are received before the timeout window, send the first 2048 and drop the rest.
  - Since this should not happen, the receiving software can use a received buffer size of 2048 as an indication that data is coming too fast.

Other Considerations

- **Applications using this feature should not expect the merged data stream packets to be in any particular order?**
  - Data packet format rather than positional order determines the sender.
- **Or… Individual data stream packets are assembled according to the input link they came in on?**
  - Position does determine order
  - However, this does not account for missing data from stations that are down.
- **Need mechanism to enable or disable this feature**
  - E.g. setting the timeout clock to 0 could disable data stream merging.
  - Or bit in Control Register
- **Timeout counter counts event clock ticks.**

Discussion