FERMILAB-POSTER-24-0127

High-throughput Custom Monitoring for the Mu2e TDAQ System

Sean Cummings*, Nishanth Shyamkumar*, Michael H L S Wang△, James B Kowalkowski△, Ryan Rivera△, Nik Sultana*
*Illinois Institute of Technology, △Fermilab

Introduction

- The Mu2e experiment is designed to observe muon to electron conversion and help understand the purpose of muons.
- Mu2e uses a carefully tailored Trigger and Data Acquisition (TDAQ) system that relies on custom network protocols (Fig 1).
- Developing, maintaining, and using custom protocols at scale requires specialized diagnostic equipment.
- We propose to use *programmable networking* to develop flexible, high performance, custom monitoring for Mu2e.
- Programmable networking relies on commodity, programmable network hardware that provides increased control and visibility at the dataplane level by applying a custom program to each packet.

Event Tag Low four bytes (63-32)				Res(31-24)	Inclusive Sub Event Byte Count(24-0)		
Event Mode (five bytes) (63-24)					Num ROCs (23-16)	Event Tag high two bytes (15-0)	
Reserved (63-32)				Data Source DTC ID (31-24)	EVB Mode (23-16)	DTC MAC Byte 2 (Part ID) (15-8)	DTC MAC Byte 0 (DTC Add) (7-0)
EMTDC (63-56)	Reserved (55-48)	Link 5 Err/stat (47-40)	Link 4 Err/stat (39-32)	Link 3 Err/stat (31-24)	Link 2 Err/stat (23-16)	Link 1 Err/stat (15-8)	Link 0 Err/stat (7-0)

Fig 1. Mu2e subevent header format.

Motivation and Goals

1. The network can “see” each packet produced by Mu2e’s sensors, including metadata that indicate instrument status and errors.
2. By leveraging the visibility of the network through programmable hardware, we also enhance visibility of the Mu2e’s experiment.
3. We can use this enhanced visibility to understand and tune the behavior of the system and diagnose problems.

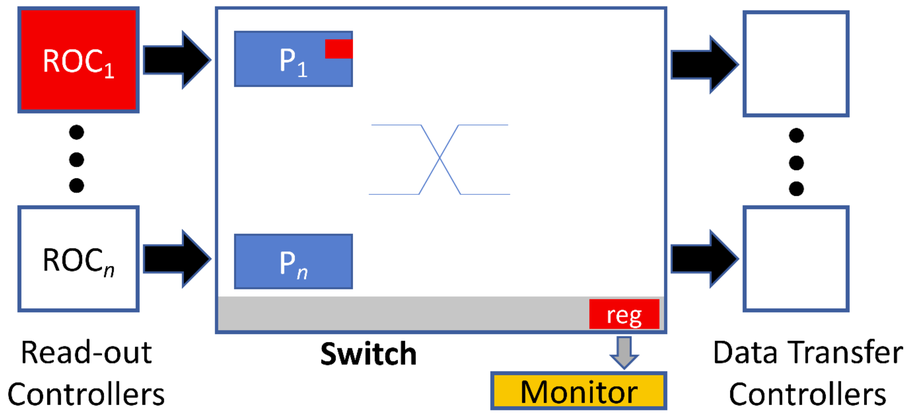

Fig 2. Overview of our prototype running on a programmable network switch.

Approach

1. We use a programmable network switch within the TDAQ network instead of a regular network switch. Both types of switch would be wired to the rest of the system in the same way.
2. In addition to carrying out a packet-forwarding function between Read-out Controllers (ROCs) and the Data Transfer Controllers, this switch is programmed to parse Mu2e’s custom packet headers.
3. Mu2e’s custom headers contain information that we can extract to learn information about the status of the experiment’s modules. This is learnt while “in flight” – before the packet reaches its destination.
4. We extract metadata from each packet that tells of per-link state and use registers in the programmable switch to store information that is gathered from packets. Through this collection of data, we can detect link failures and monitor data transfer rates.
5. Fig 2 sketches how a fault in ROC_1 is indicated in a header field within the packets (P_1) it produces. This information is stored on a register (“reg”) in the switch, which is polled by an external monitor program.

Status and Future Work

- We have a working prototype of a switch program that parses Mu2e headers and processes metadata as described above. It can detect errors and collect telemetry. We also have a simple control program that acts as a monitor (Fig 2).
- Future work includes runtime reconfiguration of the TDAQ system when error conditions are detected.

Work supported by the URA Visiting Scholars Program Award #24-S-23, and by the Fermi Research Alliance, LLC under Contract No. DEAC02-07CH11359 with the United States Department of Energy