

Synchronous High-Frequency Distributed Readout for Edge Processing at the Fermilab Main Injector and Recycler

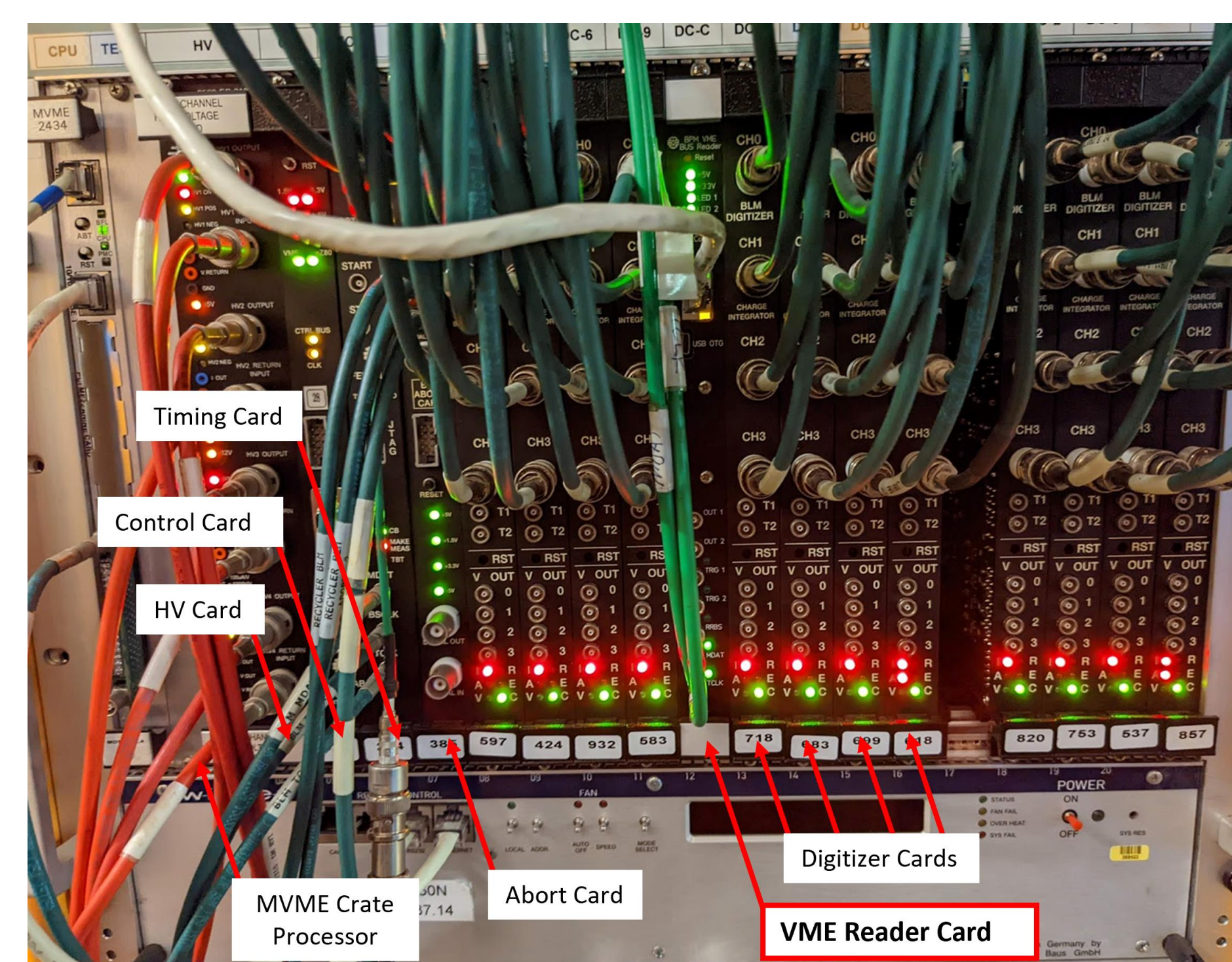
J.R. Berlioz, M.R. Austin, J.M. Arnold, K.J. Hazelwood, P. Hanlet, M.A. Ibrahim, A. Narayanan, D. J. Nicklaus, G. Pradhan, A.L. Saewert, B.A. Schupbach, K. Seiya, R.M. Thurman-Keup, N.V. Tran - Fermi National Accelerator Laboratory, Batavia, IL USA
J. Jang, H. Liu, S. Memik, R. Shi, M. Thieme, D. Ulusel - Northwestern University, Evanston, IL.

Introduction

The Real-time Edge AI for Distributed Systems (READS) project aims to deploy a Machine Learning (ML) model that will infer in real-time machine loss origins in the Main Injector (MI) and Recycler (RR). READS requires a high-frequency, low-latency collection of synchronized Beam Loss Monitors (BLM) readings from around the approximately two-mile accelerator complex.

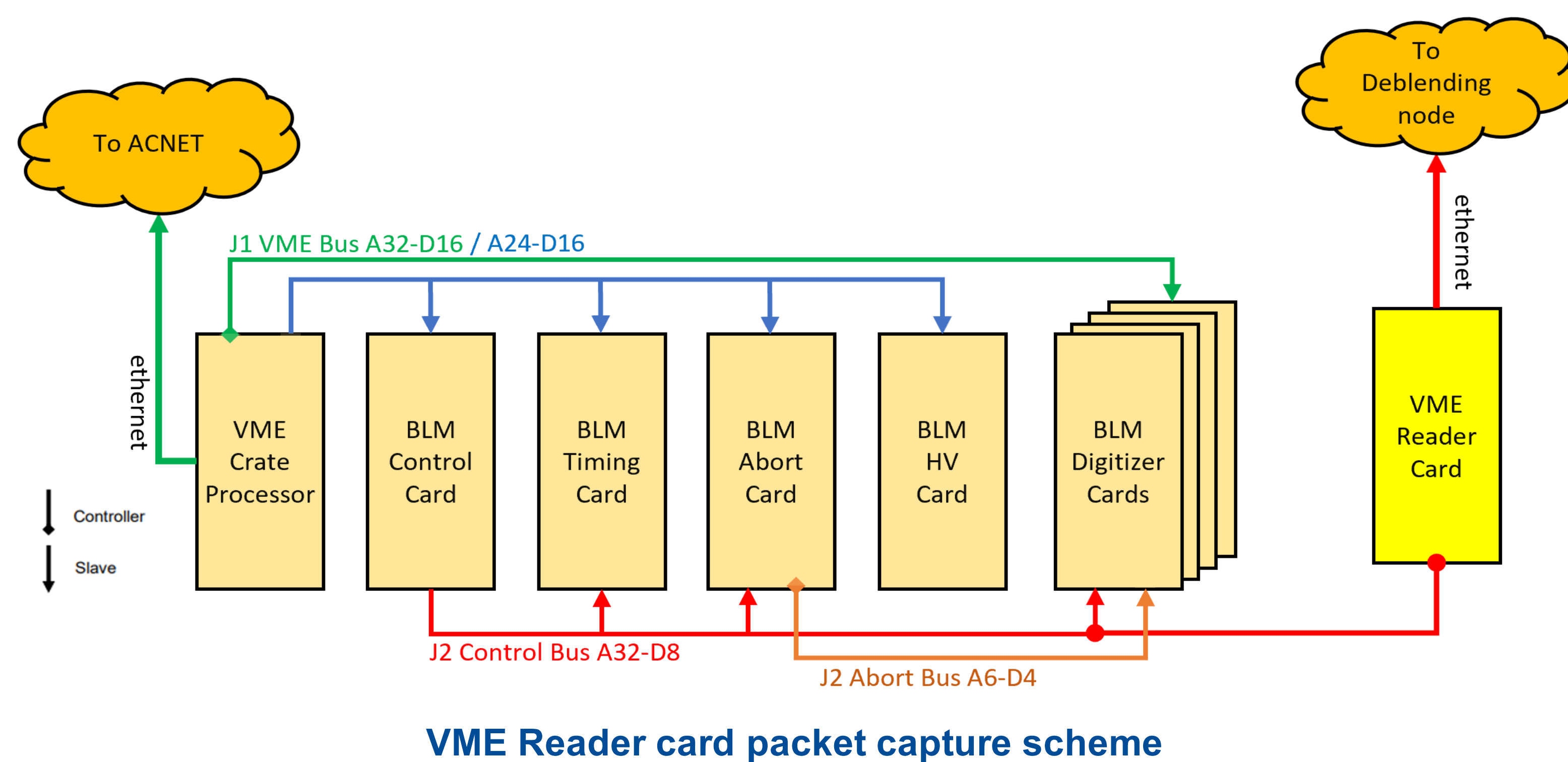
Problem

The current BLM readout system presents a bottleneck for 333 Hz data collection.

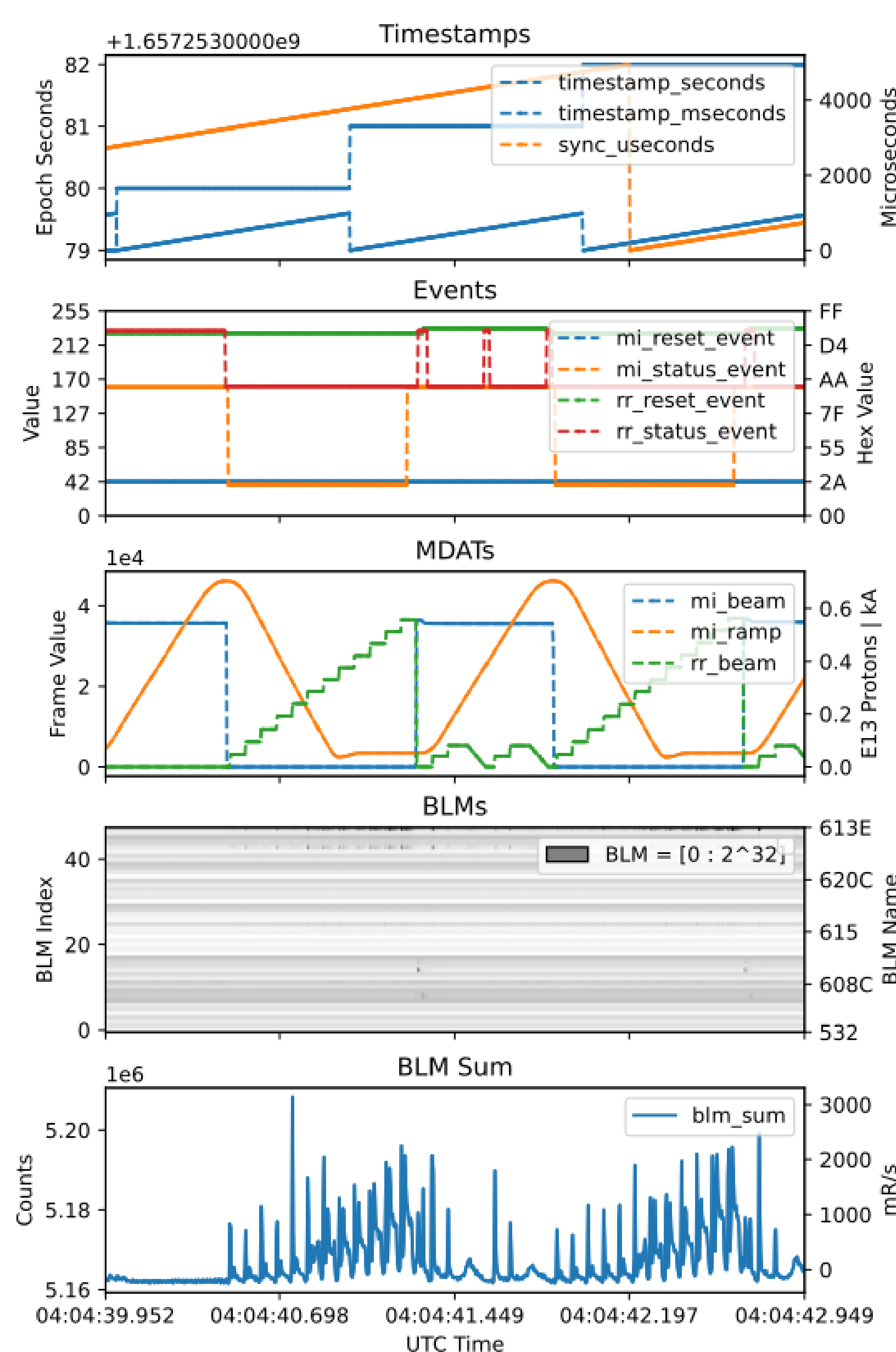


BLM Monitor System

The BLM system is a VME-based architecture composed of several cards.

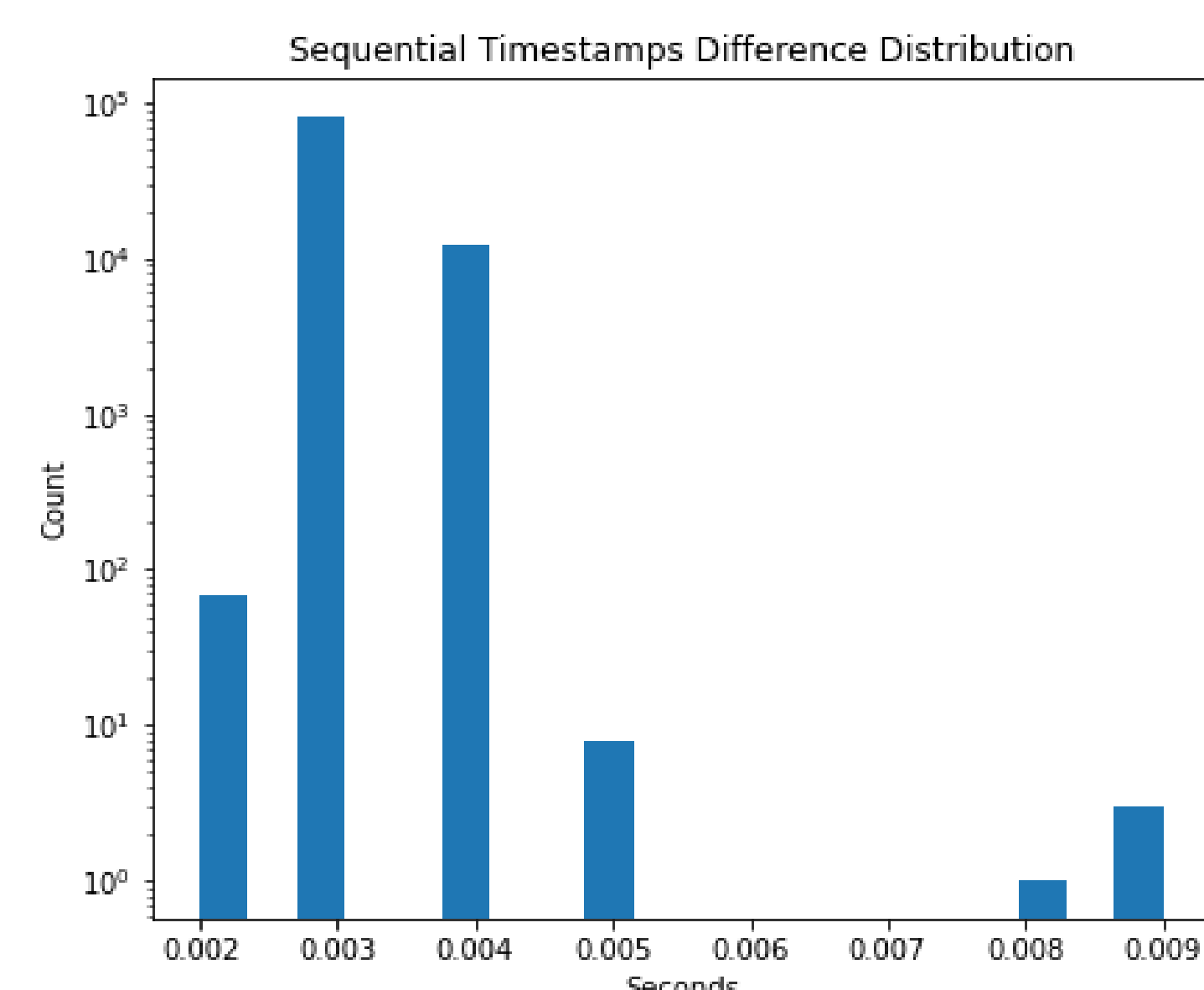


VME Reader card packet capture scheme



Data Collection

VME Reader Cards were installed and tested on seven BLM crates around MI houses in early Summer 2022.



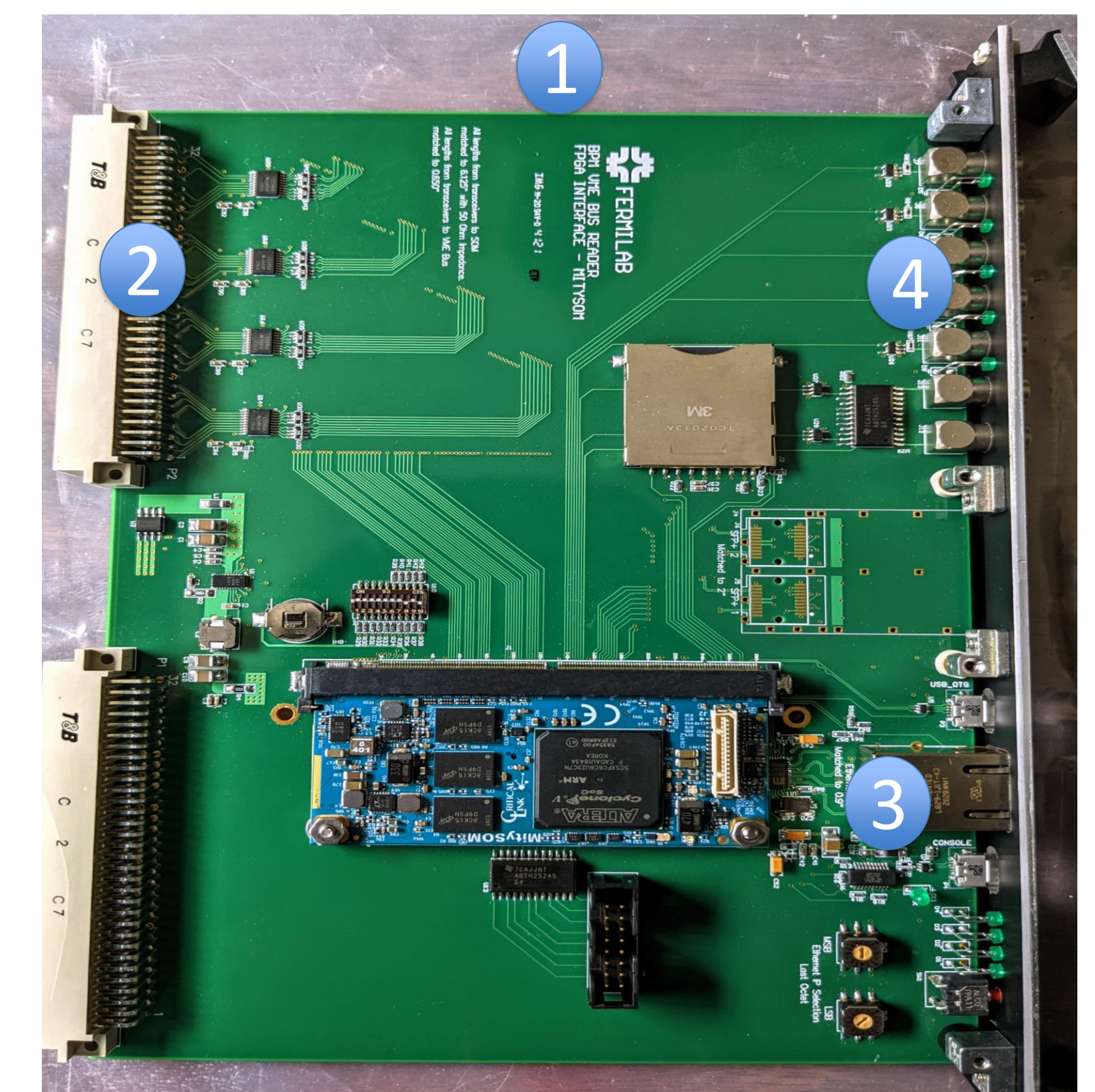
Three seconds of data collected from VME Reader in MI-60S.

VME Reader Card

The VME Reader card was designed to monitor VME Backplane activity on the J2 bus for BLM sum communication and create a dedicated channel for ML training.

Main Properties:

- 1) Mechanically fitted for the existing VME architecture.
- 2) Provides access to the J2 connector.
- 3) Provides an Ethernet communication link.
- 4) Receives clock events and machine state data from the control system through LEMO connectors.



Architecture Benefits:

- Provides a **dedicated 333 Hz Ethernet data link** for ML data collection.
- Data collection for the ML model **does not interfere with normal BLM operations**. The BLM cards in the crate are agnostic to non-communicating cards connected to the VME backplane.
- Leverage the **development framework and expertise** used in other distributed system projects.
- **Scalability as well as rapid debugging and support** of the cards after deployment.

Future Work

- A. Eliminating the transmission jitter from the cards.
- B. Building the central node capable of making inferences and handling the multiple data streams.