FY 2024 LDRD Proposal

Program: LD2410

Proposal Title: Streaming Readout Real-Time Development and Testing Platform

Principal Investigator, Division: David Lawrence , CST

Co-Investigator, Division:

Contributors, Division:	Vardan Gyurjyan, CST
	Xinxin “Cissie” Mei, CST
	Jeng-Yuan Tsai, CST
	Michael Goodrich, CST
Advisors/consultants:	Marco Battaglieri, INFN
	Sergey Furletov, ENP
	Sergey Boyarinov, ENP

Budget ($K)	**Total**	**FY25**
	$285k	$285k

Streaming Readout Real-Time Development and Testing Platform – LD2410

David Lawrence

1. Project Summary (One page maximum)

We will develop a full-scale design, testing, and validation platform for Streaming Readout Data Acquisition (SRO) systems[7,8]. It will be capable of connecting an experimental Hall to the JLab Computer Center, HPDF, or an offsite compute resource (e.g. NERSC). This will be in support of the JLab S&T vision as described in the TJNAF 2022 annual plan [1]. The platform will combine existing hardware and software components so that complete SRO systems can be configured and tested at scale. Complete SRO systems will include receiving data from the Front End Electronics (FEE), applying multiple levels of data filters, storage components, calibration components, and reconstruction components. The proposed platform will allow full SRO ecosystems to be developed and tested at high bandwidth using existing hardware components at JLab (networks, CPUs, GPUs, FPGAs, and storage) and existing software components (ERSAP, JANA, CLARA, EJFAT, JIRIAF). Components that do not exist will have software simulators inserted to provide an *effects-based* simulation. Complete monitoring of all components in the system will be developed to identify pain points and help determine resources that will be needed to implement final system designs. This will combine multiple components, some of which were developed specifically for SRO, into a single SRO system. This fills a critical gap in the development and deployment of full streaming systems for future JLab experiments like SoLID [2], CLAS12 operations at high luminosity [9] ,TDIS [3], and ePIC [4] at EIC.

FY24 Achievements

- Github repository established with connected Project for task tracking
- CLAS12 Beam-on SRO Data Capture exercise in Dec. 2023
- GlueX EVIO raw data files split on rocid
- Project presentation (poster) at ACAT workshop
- ePIC streaming demonstrator of simulated data in edm4eic PODIO format
- CLAS12 Beam-on SRO Data Capture exercise in May 2024
- NERSC exploratory allocation acquired
- DPPUI development started

FY25 S&T goals

- Complete basic framework of primary RTDP tool
- Create configurations that implement existing ePIC, CLAS12, and GlueX stream examples
- Implement emulation/simulation modules for framework
- Stress test system to demonstrate identification of bottlenecks
- RTDP Documentation

2. Detailed report of FY24 accomplishements (5 pages maximum, including figures)

FY24 saw R&D on various fronts that included beam-on data capture of CLAS12 and code development in some key areas. Two different packet capture exercises were performed that read out data from the CLAS12 detector under beam conditions. The first was done during RG-K in Dec. 2023. Data from F250 flash ADC modules[5] was read out in streaming mode and sent directly from a pair of VTP modules[9] to a compute node in the Data Center in CEBAF Center. The data passed through two 100Gbps switches and landed on a node where the packets themselves were captured using hardware timestamp information into *pcap* files (a standard format used in network testing and diagnostics). By capturing the packets exactly as they appered on the network, the data can be replayed later (e.g. *tcpreplay*) in a way that reproduces the time structure of the original data. This means dips in rate due to beam trips or spikes due to network bursts can be reproduced exactly even when beam is off. The first capture exercise read out 4 streams (2 from each VTP) at various beam currents from 10nA-150nA. The second packet capture exercise was performed in May 2024 and included the new streaming-capable firmware for the DCRB boards used to read out the Drift Chambers in CLAS12. For this exercise 24 streams were captured during RG-E which included the DC and all F250 channels for sectors 2 and 5 of the detector. Beam currents from 5nA-80nA were recorded. Unfortunately, issues with the storage on the node used to capture the data for this exercise meant only the 5nA data point was captured cleanly. Figure 1 shows distributions from the F250 for all channels from one of the 4 streams in the Dec. 2023 exercise. Each stream in this data set averaged around 4-6MB/s. These distributions are consistent with expectations. Figure 2 shows the rates for the 24 streams captured during the May 2024 exercise. Analysis of this data is still underway. One thing noted right away is that stream 7020 which shows a >30MB/s rate from one of the F250 crates was likely triggering on noise. This is evident in how little the overall rate changed for that stream during a beam trip (not shown). This just underlines the need for careful monitoring of SRO systems. Numerous utilities have been developed under this project to help process this data which could be used to form some of the tools needed for this level of monitoring. These can be found in the SRO-RTDP Github repository here: https://github.com/JeffersonLab/SRO-RTDP .

For RTDP to be effective, it needs to be experiment agnostic. In order to achcheive this, effort has been made to identify and implement data flows for different experiments where data is continuously processed from a network socket as opposed to reading from a file. Existing reconstruction codes are used for this (ePIC, CLAS12, GlueX) which currently operate on triggered, file-based data. For ePIC in particular work was needed to allow stream-like reconstruction as only simulated data currently exists and online system development is years away. Implementing this required changes to the underlying PODIO package used by ePIC. Details for this including links to the Github PR can be found in the SRO-RTDP Github site under `src/utilities/cpp/podio2tcp`. The system was tested using the FABRIC testbed. Details can be seen in the slides referenced below for the May 3, 2024 JLab EIC meeting. Similar configurations for CLAS12 and GlueX exist and are ready to be translated into RTDP configurations once the format is finalized.

The RTDP system will ultimately require detailed and arbitrarily complex configurations for the systems it implements. It was decided that this would be best done using a user-friendly graphical tool. A similar tool, *jcedit*, has been in use for years to create and maintain CODA configurations. New considerations in the era of streaming DAQ though motivate development of a new tool. Since the overlap of requirements for both SRO CODA and RTDP was large, a single tool that could serve both purposes was determined to be the most practical solution. Therefore, the *DPPUI* (Data Processing Pipeline User Interface) tool began joint development between the two projects. It is currently still under development which will continue through the end of FY24.

Other areas of R&D done for the project include identifying external I/O monitoring tools. RTDP, like DAQ systems, will require detailed monitoring at all points in all streams. CPU and memory are easily accessible for external monitoring through a number of system tools (e.g. top). Network and disk I/O has traditionally been done via in-process, self-monitoring in our field. RTDP configurations, however will need to support components that do not supply this, yet still need dynamic I/O monitoring. The RTDP project has explored two low-level mechanisms for doing this that do not require admin privileges. One uses the /proc file system under Linux and the other through a similar mechanism provided by docker to access details on processes it manages in its containers. Work has started that will populate a prometheous DB with data harvested from these sources that can be used with in Grafana to monitor processes dynamically external to the process itself. This work is partially complete and will continue through the end of FY24.

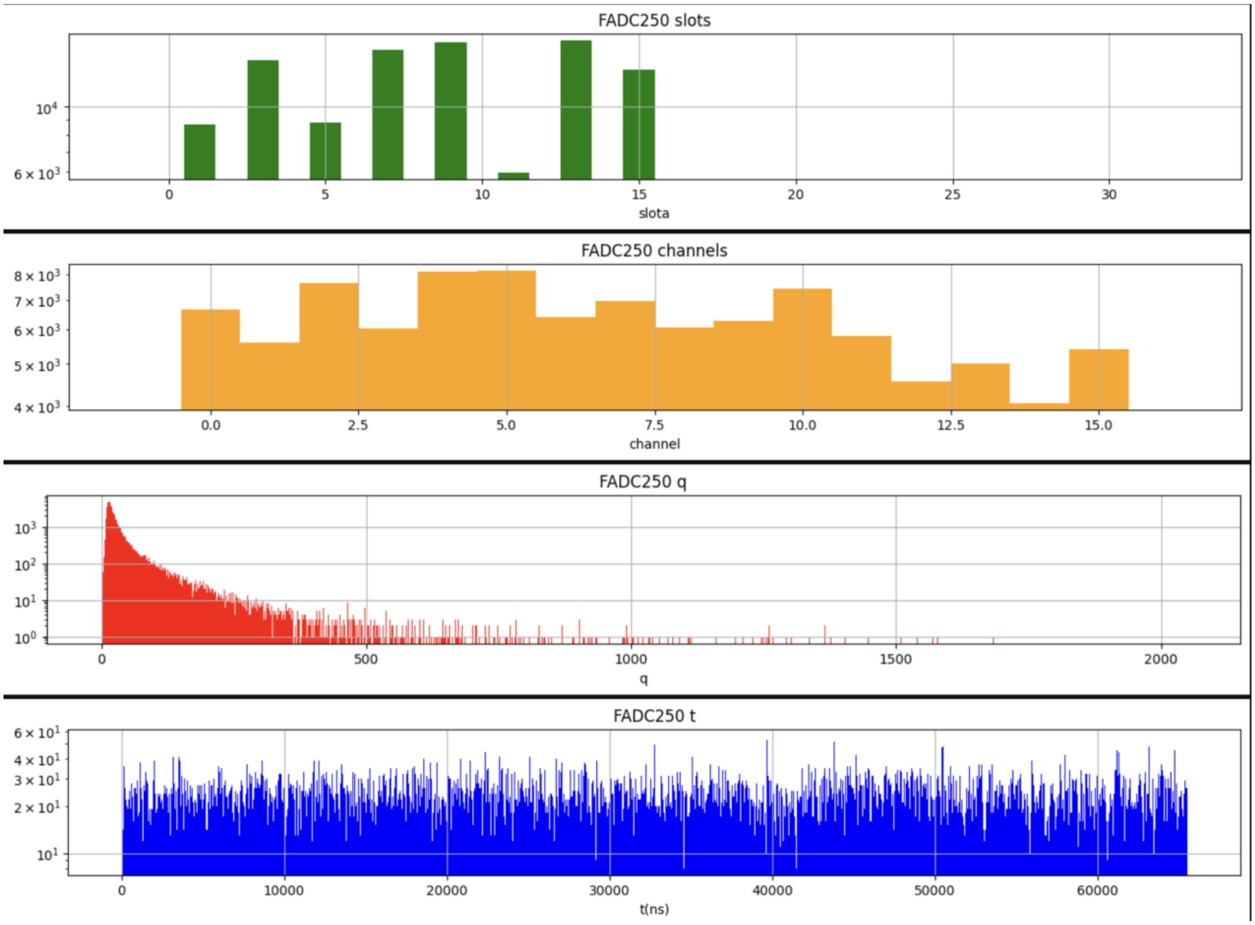


Figure 1: Basic information extracted from the Dec. 2023 packet capture exercise. From top to bottom respectively, these show the F250 ADC values for: slot, channel, pulse integreal, and pulse time. N.b. the pulse time is measured relative to the frame start time which should lead to a flat distribution as is seen here.

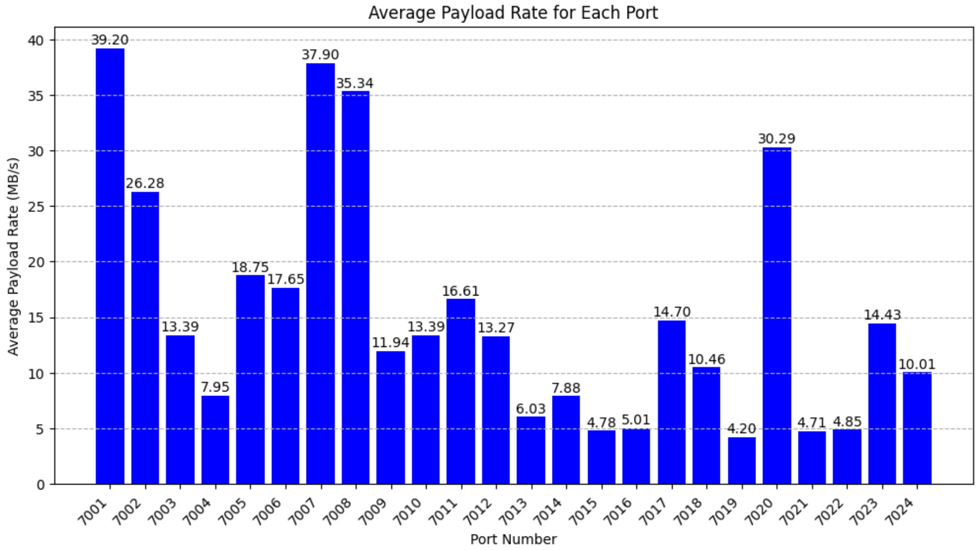


Figure 2: Average payload rate in MB/s for each stream in the May 2024 CLAS12 data packet capture exercise. Rates are shown for the 5nA data point taken under RG-E conditions. Streams going to ports 7001-7012 are from the drift chambers while ports 7013-7024 are from flash ADC F250 modules reading out various other detectors (ECAP, PCAL, …). Only data from sectors 2 and 5 were read for this exercise.

Presentations:

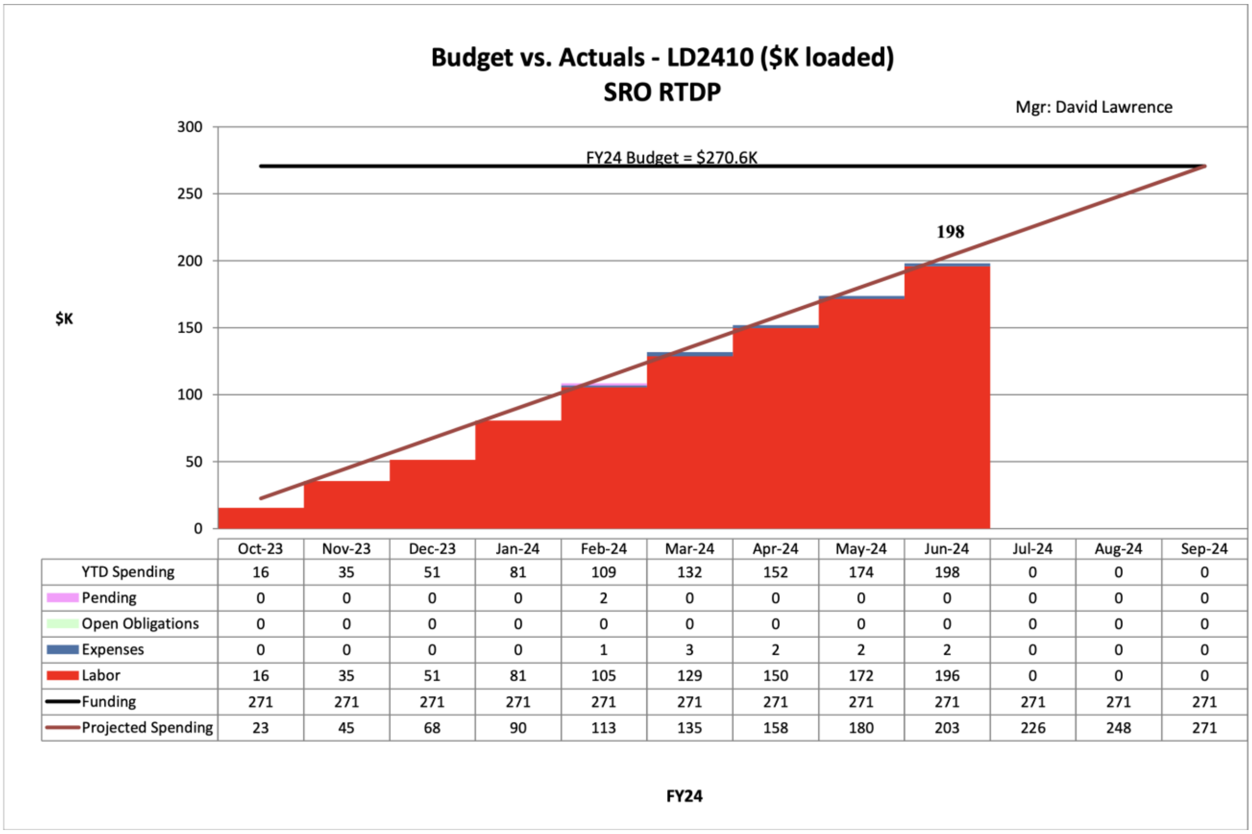
The RTDP project has been presented in several internal meetings and publically at the *Advanced Computing and Analayis Techniques 2024* workshop (poster). The table below lists the presentations along with links to the presentation material. A paper is being prepared for the peer-reviewed proceedings of *ACAT2024* at this time.

Date	Presenter	Event	Slides
June 22, 2024	David L.	SoLID Collaboration Meeting (presented as part of GCII talk)	Google Slides
June 5, 2024	David L., Patrick A., Sergey B., Ayan R., et al.	JLab Weekly Newsletter	Link
May 3, 2024 ~~Apr. 26, 2024~~	David Lawrence	JLab EIC meeting	Google Slides
Mar. 12, 2024	Ayan Roy	ACAT 2024	Google Slides
Jan. 8, 2024	David Lawrence	Hall-B Weekly Meeting	Google Slides
Aug. 10, 2023	David Lawrence	LDRD Proposal Presentation	Google Slides

This table reproduced from the SRO-RTDP wiki page where this list is maintained here: https://wiki.jlab.org/epsciwiki/index.php/SRO_RTDP

ID	Task	status	Comments
M01	Create prototype ERSAP configurations for INDRA and CLAS12 test systems	🟨	A CLAS12 example and ”Hello World” example have been placed in Github. INDRA has not been done yet.
M02	Identify or capture SRO formatted data from CLAS12 and INDRA test systems with data tag/filtering capability (output data ready for further offline processing)	✓	Data was captured at various beam currents from CLAS12 on Dec. 17. INDRA data capture done using pulser inputs to SAMPA setup.
M03	Evaluate existing solutions for configuring and launching remote distributed processes	✓	see evaluations in document on EPSCI wiki.
M04	Establish code repository(s), project site, and method of documentation	✓	This has been done here: https://github.com/JeffersonLab/SRO-RTDP
M05	Create stream splitter program for EVIO or HIPO data formatted files	✓	Created for GlueX. (See text for details on HIPO)
M06	Create stream splitter program for simulated data in PODIO for ePIC	✓	Prototype tested using FABRIC testbed. Simulated ePIC data sent from CERN to 8 different US sites.
M07	Create VTP emulator using files produced by stream splitter	🟨	Mostly done for raw data. Not started for simulated data.
M08	Create controller program to synchronize multiple VTP emulators	✓	Satisfied through alternate design using synchronized system clocks.
M09	Determine appropriate schema for all aspects of monitoring	🟨	Monitoring info. extracted as JSON records from both docker and /proc sources on Linux. Display in Grafana prototyped, but not yet complete.
M10	Establish databases for monitoring system using existing JLab servers.	×	This work has not begun
M11	Integrate Hydra as monitoring component.	🟨	Work done to containerize GlueX online monitoring in order to allow full test with Hydra. The Hydra is nearly complete containerizing Hydra (for off project purposes) which we will use.
M12	Integrate off-line data analysis framework into platform for CLAS12 data	-	planned for FY24Q4
M13	Integrate off-line data analysis framework into platform for ePIC or GlueX simulated data	-	planned for FY24Q4
M14	Integrate example JANA2 analysis into platform	-	planned for FY24Q4

Figure 3: FY24 Milestones and status. (See [6] for details on Hydra). Milestones marked with a yellow square are partially complete. M10 was planned for FY24Q3, but has not yet started.



	Oct-23	Nov-23	Dec-23	Jan-24	Feb-24	Mar-24	Apr-24	May-24	Jun-24	Jul-24	Aug-24	Sep-24
YTD Spending	16	35	51	81	109	132	152	174	198	0	0	0
🟪 Pending	0	0	0	0	2	0	0	0	0	0	0	0
🟩 Open Obligations	0	0	0	0	0	0	0	0	0	0	0	0
🟦 Expenses	0	0	0	0	1	3	2	2	2	0	0	0
🟥 Labor	16	35	51	81	105	129	150	172	196	0	0	0
— Funding	271	271	271	271	271	271	271	271	271	271	271	271
— Projected Spending	23	45	68	90	113	135	158	180	203	226	248	271

Figure 4: FY24 Spending (first 3 quarters)

3. FY24 Research Plan (3 pages maximum)

- **FY25 Project Description:**
  The second year will leverage the R&D, datasets, and software development produced during the first year of the project. Specifically, the formal RTDP software framework will be developed using configurations generated from, the DPPUI project. *DPPUI* is the *Data Processing Pipeline User Interface* graphical configuration tool developed jointly for RTDP and SRO CODA. Proxy software components will be developed that can be configured via DPPUI. Components that directly access heterogeneous hardware will also be developed. Exercises that test the system under stress conditions will be executed. Success will be determined by demonstrating the various components working together for these exercises. Documentation accessible by non-experts will be produced.

  The primary objectives of the project remain unchanged from the original proposal and are copied here. Detailed milestones, revised for FY25 are listed below.

  Objective 1: **Process Launcher:** The software that will be used to configure and launch each component does not itself need to be highly performant. The highly performant and specialized codes will already be encapsulated in the components themselves that that will have been developed outside of this project (sans objectives 3 and 4 below). The most appropriate language for this will be something like python3 or possibly julia. The configurations themselves should be expressible in a static file format to allow other tools for editing and visualization of the configurations to be developed in the future. A configuration format such as YAML would be a good choice as it is widely supported across numerous programming languages. For inter-process communication that will need to support both local and wide area networks, ZeroMQ or a similarly common, open source product will be used.

Objective 2: **Monitoring System:** The monitoring system will be developed using an appropriate set of existing free or open source tools such as Prometheus, and Grafana. Both Prometheus and Grafana are already in use in other places at JLab which will allow leveraging local expertise, installations, and network configurations. Additional visualization tools will need to be developed as part of this project. These will be web-based were appropriate, though some specialized tools may also be needed. Advanced monitoring for continuous data validation will also be needed. Use of the AI/ML based Hydra [6] monitoring system would be appropriate for this purpose.

Objective 3: **Proxy Components:** The proxy components that will allow effects-based simulations will be developed using a performant language such as C++. These components will need to read in and write out data streams to mimic the operation of the real component that does not currently exist. For example, a proxy component that is used to represent a reconstruction algorithm that is expected to read data from a single stream, perform X Mflops/kB, and write roughly 1/2 of the data to the output. The proxy would need to understand the header information of the incoming stream enough to modify it for the output, but would not need to understand the payload. It would also need to exercise enough dummy operations on the CPU cores to mimic the X Mflops/kB it was configured for.

Objective 4: **Multi-stream Event Source:** The multi-stream event source will need to be written in a highly performant language such as C++ or possibly Java. Data will be read from a file that is either in an experimental raw data format such as EVIO or simulated event data format such as ROOT. For data that is not already in a format that includes DAQ system indexing (e.g. simulated data) it will need to apply an inverse translation table to convert from detector component indexing. The DAQ indexing is needed to identify the crate/slot/channel element the data would have originated from so it can be sent over the appropriate output stream so as to mimic live data. This component will require multiple processes spread over multiple compute nodes in close coordination in order to achieve the high bandwidths needed.

Objective 5: **High Bandwidth Test:** Configuring a full scale system that includes both real and proxy components and testing it at high bandwidth is necessary to demonstrate the platform's core functionality. Current expectations are to have a 400Gbps link available between the Hall A,B,C counting house and the Computer Center in CEBAF Center sometime in FY2024. The high speed testing will be coordinated to occur when the beam is down so that the full bandwidth will be available for the testing periods. The SoLID experiment serves as an example of the type of high bandwidth experiments being anticipated to run at JLab in the future. It will then serve as a useful guide for the testing configuration, even if the configuration is not an exact match for SoLID. There are currently eight U280 FPGA cards in the Computer Center purchased for use with the EJFAT project which would be available to use for these tests. Similarly, the Scientific Computing farm will have a few dozen GPUs (mostly Tesla T4's) available that could also be utilized for these tests. Utilizing real hardware components will be an important part of the platform and so will need to be included for the full scale configuration testing. We will utilize existing components developed outside of this project to exercise the heterogeneous components. For example, PHASM, CLAS12 tracking, and the EIC R&D project: ML4FPGA [12].

Objective 6: **Insights for HPDF:** The platform will be a tool for developing, testing, and validating SRO systems that utilize remote compute facilities. Here, "remote" can mean the Computer Center relative to the Counting House. Similarly, the HPDF is expected to serve as a remote compute facility for experiments outside of JLab. The proposed platform tool will provide valuable insights into how to best use the HPDF to support experiments. The approach to obtaining this objective will be to partner with a remote facility such as BNL or NERSC to perform some limited testing once the project matures. Exercising a high bandwidth application where the data originates at JLab and is sent to a remote compute facility will give valuable insight on how the HPDF might handle data streaming in from a remote site. A reach goal will be "bouncing" the data stream(s) we send to a remote site back to the HPDF as an additional testing phase.

Milestones:.

Y2Q1

- **M15**: Establish general framework for RTDP simulation
- **M16**: Create configurable CPU proxy component
- **M17**: Create configurable GPU proxy component (hardware and software)
- **M18**: Create configurable FPGA proxy component (hardware and software)

Y2Q2

- **M19**: Create functioning hardware GPU component (e.g. CLAS12 L3)
- **M20**: Create functioning hardware FPGA component (e.g ML4FPGA)
- **M21**: Configure simulation of full SRO system using existing JLab hardware resources

Y2Q3

- **M22**: Establish working test of system that transfers >=100Gbps from CH to compute center
- **M23**: Establish working test of system that includes GPU component for portion of stream
- **M24**: Establish working test of system that includes FPGA component for portion of stream
- **M25**: Test system with remote compute facility (e.g. BNL or NERSC) at limits of available resources

Y2Q4

- **M26**: Configure system that results in stream(s) being received by JLab from external source
- **M27**: Collaborate with HPDF group to evaluate processing SRO data at JLab for external experiments
- **M28**: Complete documentation for platform to be used by non-experts

- **Potential funding opportunities:** If successful, the project is expected to be maintained under operations. Ideally, the outside community will contribute to the maintenance and feature upgrades.

4. Budget

Requested Budget for Effort by Investigator				
Name of Investigator	**Role (PI, Co-I, etc.)**	**FY25 Budget ($K)**	**FY25 Effort (% FTE)**	**Total Effort (%FTE)**
David Lawrence	PI	$83	25%	25%
Vardan Gyurjyan		$42	15%	15%
Cissie Mei		$40	20%	20%
Jeng Tsai		$63	50%	50%
Michael Goodrich		$51	20%	20%
	Subtotal for effort	$279	130%	130%
Equipment	Non-capital			
	Capital			
Subcontracts	Person/ organization			
Materials/ Supplies				
Travel		$6.5		

5. Budget Justification

Personnel:

Team Member	Role	Project Contribution	Specific Aims (see first year proposal for details)
David Lawrence	PI	Senior project manager	1-6
Vardan Gyurjian	Contributor	SRO DAQ expert. ERSAP,CLARA, JIRIAF	1,4,5
Cissie Mei	Contributor	Benchmarking/Monitoring GPU,FPGA	2,3
Jeng-Yuan Tsai	Contributor	Software developer, HTC systems	1-6
Michael Goodrich	Contributor	Simulation/Emulation	3

Requested New Hires:

Name of Hire	Type of hire (strategic, staff, PD)	Position Description/Justification	Projected Cost ($K/FY)	Expected timeline

Equipment:

Equipment	Justification	Projected Cost ($K in FY25)

Materials:

Name of Material	Description	Cost per FY	Total Cost

Sub-Contracts:

Subcontract	Institution	Description/Justification	Duration	Cost/FY	Total Cost

Travel:

Activity	Destination	Name of travelers	Estimated Cost
ACAT2025	TBD	TBD	$6500

Current and Pending FY 2024 and FY 20245 Funding:

Team Member	Project Number, Sponsor	FY 2025 %FTE Anticipated
David Lawrence	**This DRD project**	**25%**
	SCIOPS/SCI, DOE NP (JLab OPS)	75%
Vardan Gyurjyan	**This DRD project**	**15%**
	HPDF	50%
	PET Scan LDRD (proposed)	10%
	SCIOPS/SCI, DOE NP (JLab OPS)	25%
Cissie Mei	**This DRD project**	**20%**
	HPDF	30%
	DPU Benchmarking LDRD (proposed)	50%
Jeng-Yuan Tsai	**This DRD project**	**50%**
	DPU Benchmarking LDRD (proposed)	50%
Michael Goodrich	**This DRD project**	**20%**
	SCIOPS/SCI, DOE NP (JLab OPS)	80%

6. References (Not included in page count)

[1] TJNAF Annual Lab plan. *(see section Advanced Computer Science, Visualization, and Data ppg. 5-6)* https://www.jlab.org/sites/default/files/documents/lab/TJNAF%20Annual%20Lab%20Plan.%20public%20FY2022.pdf

[2] *SoLID (Solenoidal Large Intensity Device) Updated Preliminary Conceptual Design Report*, The SoLID Collaboration, November 2019
https://solid.jlab.org/DocDB/0002/000282/001/solid-precdr-2019Nov.pdf

[3] *Measurement of Tagged Deep Inelastic Scattering (TDIS)* Hall A and SBS Collaboration Proposal PR12-15-006
https://www.jlab.org/exp_prog/proposals/15/PR12-15-006.pdf

[4] R. Abdul Khalek, A. Accardi, J. Adam, et al. *Science Requirements and Detector Concepts for the Electron-Ion Collider: EIC Yellow Report* NIMA 1026, 122447 (2022).
https://www.sciencedirect.com/science/article/abs/pii/S0168900220302539

[5] See CODA hardware manuals for fADC250, fADC125, F1TDC
https://coda.jlab.org/drupal/node/57/724298

[6] Thomas Britton, David Lawrence, and Kishansingh Rajput *AI Enabled Data Quality Monitoring with Hydra*
EPJ Web of Conferences 251, 04010 (2021) CHEP 2021
https://doi.org/10.1051/epjconf/202125104010

[7] Heyes, G. (2019, February 12). Streaming Grand Challenge [Slides]. Retrieved from
https://indico.jlab.org/event/307/contributions/4681/attachments/3852/4660/20190212_Grand_Challenge_Overview.pdf

[8] Ameli, F., Battaglieri, M., Berdnikov, V.V. *et al.* Streaming readout for next generation electron scattering experiments. *Eur. Phys. J. Plus* **137**, 958 (2022). https://doi.org/10.1140/epjp/s13360-022-03146-z

[9] S. Boyarinov, B. Raydo, C. Cuevas, *et al. The CLAS12 Data Acquisition System* NIMA 966, 163698 (2020).
https://www.sciencedirect.com/science/article/abs/pii/S0168900220302539