UEI offers data acquisition and control modes that satisfy an assortment of application requirements, such as immediate-data control modes, VME/Reflective Memory alternatives, and high-speed acquisition modes, to name a few.

This FAQ provides a comparison of our DAQ mode features followed by a brief description of each mode and answers to frequently asked questions.

Overview of UEI Data Acquisition and Control Modes

UEI data acquisition modes are available for transferring data using hosted PowerDNA Cube or RACK systems and/or using UEIPAC embedded systems.

For **PowerDNA** systems, the user application runs on a host PC and the Cube / RACK acts as a slave over Ethernet. For embedded **UEIPAC** systems, the user application runs locally on the PowerPC processor on the UEIPAC CPU board.

The following table provides a comparison of our most commonly used DAQ modes:

DAQ Mode	Acquisition Speeds	Good for Closed Loop Applications?	No. of Data Points	Timebase Maintained By	Supported on PowerDNA/ UEIPAC/Both
Point-by-point	up to 100 Hz	Yes, if running < 10 Hz	1 per channel	User application	Both
ACB (Advanced Circular Buffer)	up to speed of I/O board	No	Multiple per channel	Cube/RACK hardware	PowerDNA
RtDMap	up to 5000 Hz[1]	Yes	1 per channel	User application	Both
RtVMap	up to speed of I/O board[1]	Yes	Multiple per channel	User application	Both
ADMap	up to 5000 Hz[1]	Yes	1 per input channel	Cube/RACK hardware	PowerDNA
AVMap	up to speed of I/O board[1]	Yes	Multiple per input channel	Cube/RACK hardware	Both

Note that UEI also offers modes for special purpose data / message transfers:

- **Asynchronous mode:** an event-driven mode used to acquire data on a specific event, (e.g., digital I/O pin change of state, CAN or ARINC message received, counter value match, timer expiring, etc.)
- **Messaging modes**: ACB, RtVMap, and/or a Messaging-specific protocol (Msg) can be used with messaging I/O boards where data is a stream of bytes logically divided into frames, messages, or strings, (e.g., serial, CAN, ARINC boards)

[1] Note that the maximum speed for RtDMap, RtVMap, ADMap, and AVMap is additionally limited by system hardware. Acquisition speeds for PowerDNA systems are limited by the speed of your Ethernet port.

© United Electronic Industries, Inc.
27 Renmar Avenue.
Walpole, MA 02081
www.ueidaq.com
508.921.4600

How does Point-by-point Mode work?

Point-by-point mode provides easy access to a single I/O board at a non-deterministic pace. Point-by-point mode is also referred to as Immediate mode.

Using Point-by-point mode, your application acquires a single data point for each configured channel of a single I/O board.

Point-by-point mode works with all UEI I/O boards in all UEI products; however, this mode is limited to ~100 Hz as the maximum acquisition speed.

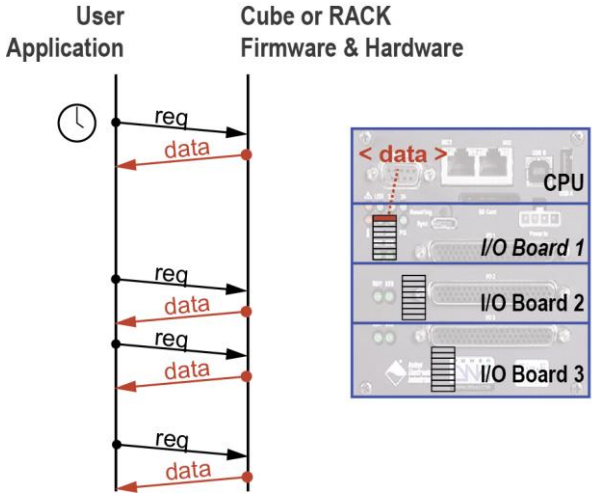

Figure 1. Input Board Using Point-by-point Mode

How does ACB Mode work?

Advanced Circular Buffer (ACB) mode is a buffered data transfer protocol. ACB mode uses circular buffers for outgoing data packets and for incoming packets. Each buffer is divided into frames, allowing a packet ring for temporary and sequential storage of received or transmitted packets.

Cube/RACK firmware issues event notification (referred to as DQE events) when incoming or outgoing data crosses a frame boundary, initiating packet assembly and a transmission. If the application detects a missing packet, it requests retransmission.

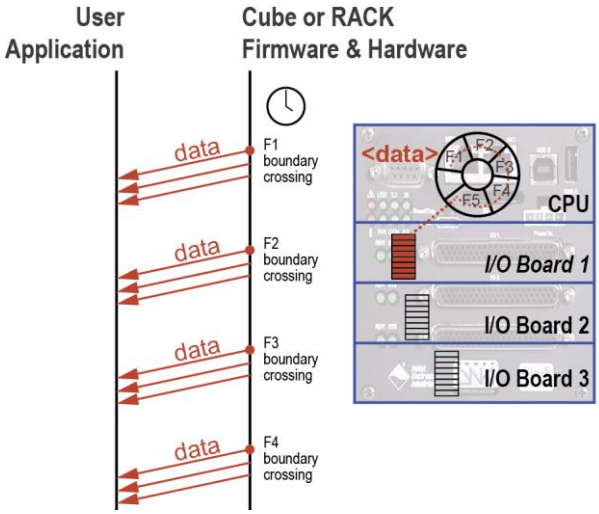

Figure 2. Input Board Using ACB Mode

Every data point is guaranteed delivery; however, because data is accumulated in a circular buffer of output or input packets, the data will have a delivery delay. For this reason, ACB mode is not appropriate for closed-loop control applications but is ideal for data acquisition applications that require all data to be delivered without gaps but can accept slight delays in delivery.

The ACB protocol is supported in PowerDNA systems (user application running on a host PC) but not UEIPAC systems. UEIPACs do not require ACB mode because the user application runs on the onboard PowerPC processor; without Ethernet data transfers, lost packets are not an issue.

© Copyright 2018
United Electronic Industries, Inc.

www.ueidaq.com
508.921.4600

How does RtDMap Mode work?

RtDMAP is designed for closed-loop (control) applications. Users set up a "map" of I/O boards and physical channels to acquire input data from or deliver output data to.

RtDMAP optimizes transfer overhead by packing all channel data from multiple I/O boards into a single transfer. The timebase for transferring data is maintained by the user application.

Transferred data is composed of **one data point** per DMAP-configured physical channel. In PowerDNA mode applications, this results in transferring a single UDP packet for all user-configured channels, reducing network overhead.

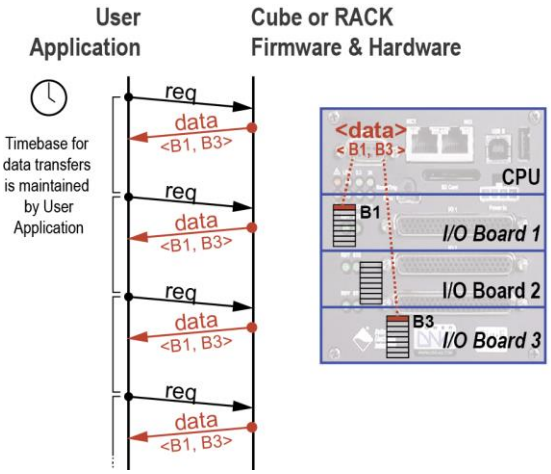

Figure 3. Input Boards Using RtDMap Mode

The maximum speed of the transfer is limited to ~5000 Hz or the speed of the board.

How does RtVMap Mode work?

RtVMap is a data transfer protocol best used for control applications.

RtVMap is similar to RtDMap; however, RtVMap allows a **variable number of data values** to be acquired or delivered for each RtVMAP-configured I/O channel.

Similar to RtDMap, users set up a map of I/O boards and physical channels. A timebase maintained by the user application paces output data delivery from the user application, as well as requesting input data from the Cube/RACK.

Upon request, the Cube/RACK returns the RtVMAP input data, with all configured channels packed into a single transfer. Packets can be resized dynamically to optimize bandwidth use.

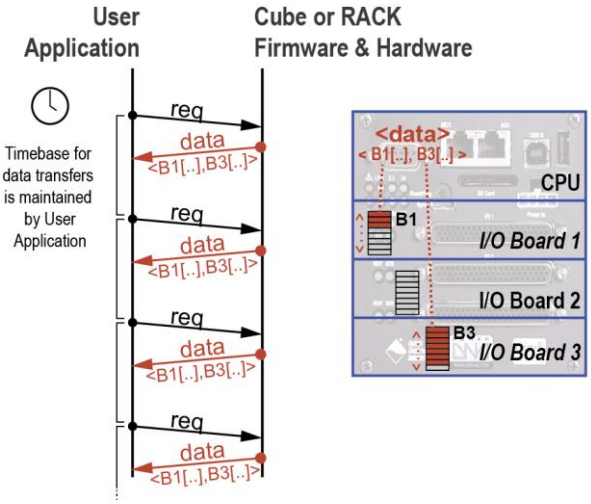

Figure 4. Input Boards Using RtVMap Mode

The maximum speed of the transfer is limited to the speed of the board, though in larger PowerDNA systems, the speed is also limited by Ethernet data transfer restrictions.

© Copyright 2018
United Electronic Industries, Inc.

www.ueidaq.com
508.921.4600

How does ADMap Mode work?

ADMap is a data-mapped protocol best for control applications and available for use with input boards.

With ADMap, users map which I/O boards and physical channels to acquire data from for each data transfer; users also configure a hardware condition to trigger the transfer.

Once the user-programmed system condition occurs (a condition based on a synchronized clock or timer countdown), the UEI system reads **a single piece of channel data** for all channels configured for the ADMap. The CPU assembles the data and transfers it to the user application.

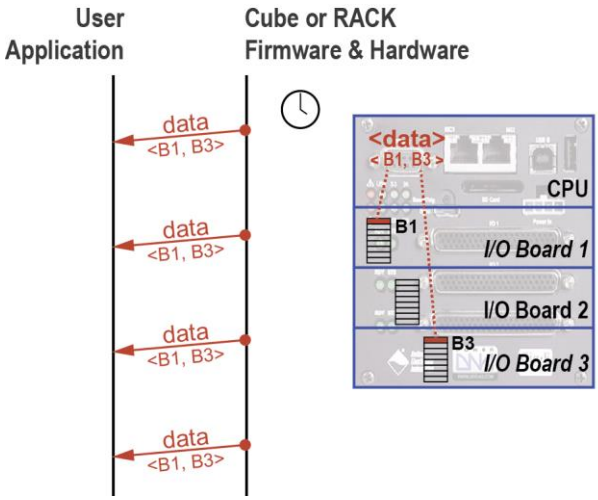

Figure 5. Input Boards Using ADMap Mode

How does AVMap Mode work?

AVMap is a variable data-mapped protocol best for control applications and available for use with input boards.

AVMap is similar to ADMap, except AVMAP allows users to acquire a **variable number of samples per configured channel** instead of a single sample.

With AVMap, users configure which I/O boards and physical channels will be acquired for each data transfer and configure the hardware condition to trigger the transfer.

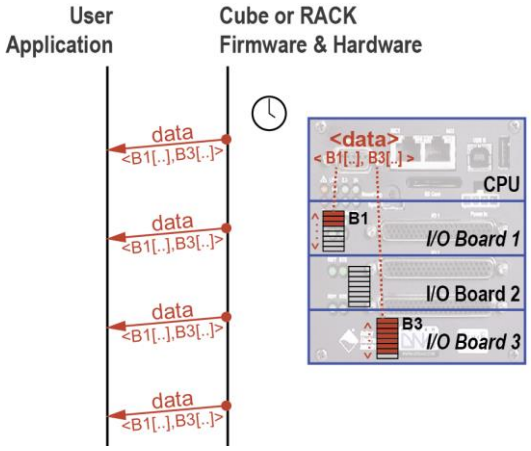

Figure 6. Input Boards Using AVMap Mode

Once the user-programmed system condition occurs (a condition based on a synchronized clock or a FIFO watermark), the UEI system reads however many data samples are available from each configured channel on all input boards defined in the AVMap. The data is assembled and transferred to the user application.

© Copyright 2018
United Electronic Industries, Inc.

www.ueidaq.com
508.921.4600

When should I use AXMap vs. RtXMap?

Please refer to the following table to determine whether the AXMap or RtXMap protocols better fit the requirements of your application.

FEATURE	AXMAP	RTXMAP
Supported I/O boards	Input boards only	A variety of I/O boards
Response time	Improved response time for applications that need immediate output responses for changes in the data (not waiting for next time frame)	Best to use when there is a defined frame rate uniform to all Cube or RACK systems on the network (or a fraction of it)
Synchronization to 1PPS or IEEE-PTP source for transfers between hardware and app	Yes	No
Synchronization to 1PPS or IEEE-PTP source for data acquisition on I/O boards	Yes	Yes
Input/Output data delivery control	Supports input boards only; output boards would need to be configured as RtXMap	Guaranteed that inputs are read first and outputs are written second, so input data is acquired in steady state
Recoverable lost packets[2]	Because AXMap doesn't use a request/reply scheme, it takes longer and is less reliable to recover lost PowerDNA packets. Up to 10 packets can be recovered.	Uses a request/reply scheme, which allows immediate identification of lost PowerDNA packets. Up to 10 packets can be recovered.
Flexibility/Debug	RtXMap is more flexible than AXMap, (e.g., allocates channels and data sizes in run time for RtVMap) RtXMap is easier to debug than AXMap (e.g., control is via user application)	

[2] Note that the lost packet rate on a "typical" network is between 1 in 1,000,00 and 1 in 10,000,000. Loss of packets more frequently than that often indicates a network problem.

© Copyright 2018
United Electronic Industries, Inc.

www.ueidaq.com
508.921.4600

How does Asynchronous mode work?

Asynchronous mode is similar to ADMap, except that the data packets are sent upon a user-programmable asynchronous event.

Users configure which physical channels to acquire data from for the data transfers and program a hardware condition to trigger the transfer.

Once the system condition occurs, the UEI Cube/RACK reads a single piece of channel data for each configured channel. The CPU assembles the data and transfers the packet to the user application.

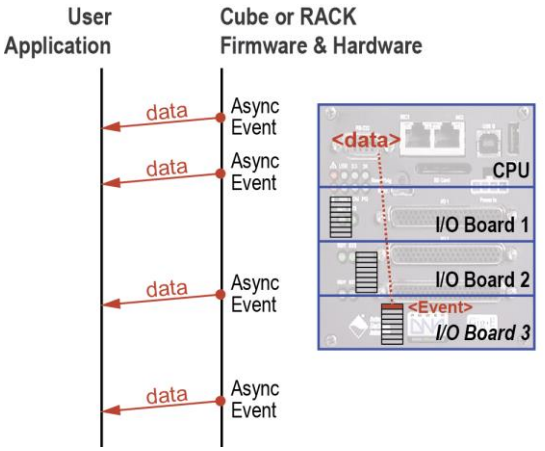

Figure 7. Input Board Using Asynchronous Mode

The following table lists I/O boards that support Asynchronous mode and which types of events they support:

TYPE OF I/O BOARD	TYPE OF ASYNCHRONOUS EVENTS	EXAMPLES OF SUPPORTED BOARDS
Digital I/O Boards	Edge detection/Change of state Events	DNx-DIO-401/3/4/5/6, DNx-DIO-449
Serial I/O Boards	Timeout conditions, TX done, RX done, and pattern detection Events	DNx-SL-501, DNx-SL-508
CAN-Bus Boards	FIFO watermark and bus error Events	DNx-CAN-503
Counter/Timer Boards	Count complete, edge detection on input and/or gate pin Events	DNx-CT-601, DNx-CT-602
IRIG-650 Boards	1PPS and GPS data ready Events	DNx-IRIG-650
Avionics Protocol Boards	Many protocol-specific Events	DNx-1553-553, DNx-AFDX-664

© Copyright 2018
United Electronic Industries, Inc.

www.ueidaq.com
508.921.4600

How do I get started with each of the modes?

Using UEI-provided sample code is the best way to get started when programming each of the DAQ modes.

UEI provides object-oriented Framework samples for programming in C++, C#, .Net, MATLAB, LabView, and many more. Framework is available when using Windows OS and specifically supports Point-by-point and ACB (buffered) modes.

UEI also provides C-based sample code when programming low-level applications (using Windows OS, Linux OS, or VxWorks OS).

All UEI software installations include an extensive set of code samples.

Are all modes available for all products?

Acquisition modes are I/O board-specific. Users can check the examples repository to see which I/O boards have sample programs supporting specific acquisition modes. Point-by-point mode is supported for all I/O boards.

Additionally, note that UEIPAC systems specifically support Point-by-point, RtDMap, RtVMap, AVMap, and ADMap modes (not ACB).

When deciding which acquisition mode to use, UEI support is happy to help guide you in determining which I/O mode is best suited for your application.

For more information:

Please contact UEI support at support@ueidaq.com or call 508.921.4600 with any questions.

© Copyright 2018
United Electronic Industries, Inc.