

Answers to your Questions about Implementing Fieldbus in DeltaV Systems

This whitepaper answers many questions users have asked about implementing Foundation fieldbus systems with DeltaV systems.



Table of Contents

Introduction..... 4

 References..... 4

 Definitions 4

Questions and Answers..... 5

 Question 1. What is “Control in the Field”? 5

 Question 2. What is the interaction between controller execution rate and the Macrocycle when Control in the Field is used? 5

 Question 3. What is “Hybrid Control”? 6

 Question 4. What is the interaction between controller execution rate and the Macrocycle when Hybrid Control is used? 6

 Question 5. How does the Macrocycle schedule multiple control loops on a segment? 7

 Question 6. How does the LAS schedule data transfer?..... 7

 Question 7. How do Scheduled Data Transfers work? 8

 Question 8. How do Unscheduled Data Transfers work? 8

 Question 9. What happens if the controller execution rate is faster than the Macrocycle? 8

 Question 10. What factors influence calculation of the Macrocycle?..... 8

 Question 11. Can the Macrocycle be computed in the design phase? 9

 Question 12. What does MissedViewListScan indicate about system operation?..... 10

 Question 13. What is a NumDIITolkenPass Timeout, and how should this statistic be interpreted? 10

 Question 14. What is the relationship between Function Blocks and Shadow Blocks?..... 10

 Question 15. What is the DeltaV implementation of the FOUNDATION fieldbus standard with regard to data link layer and device timer values? 11

Question 16. What items should be considered to obtain an efficient fieldbus implementation? 11

Question 17. Why are certain Fieldbus Network Parameters pre-set for DeltaV systems? 11

Introduction

Questions often arise when implementing Fieldbus in DeltaV systems, such as “What is the meaning of a particular term”? “How does it apply to DeltaV systems”? “How do I calculate the Macrocycle”? and many more questions. This paper provides answers to common questions, using a question and answer format.

Since questions will continue to arise, this paper will be periodically updated to include additional questions and answers. Please do not hesitate to e-mail or write us your questions. There are probably many other users who need answers to the same questions you are asking.

References

The Fieldbus Foundation provides documentation to help you understand fieldbus principles. It is highly recommended that you obtain copies of their “Technical Overview” and their “System Engineering Guidelines”. These items are available in Adobe pdf format from the foundation website at <http://www.fieldbus.org>.

Information for installing DeltaV fieldbus–interface hardware and fieldbus segments is available in the DeltaV manual, Fieldbus Installations in a DeltaV Automation System. Fieldbus configuration information is found in DeltaV Books Online (BOL).

Definitions

Several terms are used in this paper which you will need to know. These terms are:

- **Compel Data** — a command issued by the Link Active Scheduler (LAS) to a specific device for it to broadcast its scheduled data to all devices configured to receive the data. The broadcast data is primarily used for loop control.
- **Link Active Scheduler (LAS)** — a device on a fieldbus segment which schedules when data can be sent on the segment. In DeltaV systems, the LAS is built into the DeltaV H1 card.
- **LiveList** — a list of all devices that are properly responding to the pass token message.
- **Macrocycle** — the time it takes for a fieldbus segment to execute the schedule of the Link Active Scheduler (LAS) one time. Macrocycle depends upon system configuration and is based on several factors (explained in the questions below).
- **Missed ViewList Scan** — a diagnostic parameter that informs a user when ViewList data is being requested more frequently than it can be sent.
- **Pass Token (PT)** — a signal passed from the Link Active Scheduler (LAS) to a device on the fieldbus segment which gives permission to the device to communicate unscheduled information over the segment.
- **Probe Node** — A request of the Link Active Scheduler which is periodically sent to each device that is not currently on the LiveList asking for an indication that it is able to communicate.
- **Scheduled Data Transfers** — regular, cyclical, exchange of control loop data between devices on a fieldbus segment broadcast upon receiving a Compel Data command.
- **Unscheduled Data Transfers** — user–initiated requests, either manually or by system configuration, for data transfers such as reads and writes (typically setpoint changes, tuning changes, downloads, uploads, and ViewList updates) and event reporting. Unscheduled data transfers are sent by a device when it receives a pass token.
- **Time Distribution** — A Link Active Scheduler message sent to all devices on the segment giving them the current real time so that all segment devices are running by the same clock.
- **ViewList** — fieldbus function block parameters that are provided primarily for operator display. There are four types of ViewLists: 1) dynamic operation parameter values (e.g., measured variable), 2) static operation dynamic parameters values (e.g., set point), 3) all dynamic operation values (e.g., set point value returned from controller), and 4) other static operation parameters.

Questions and Answers

Question 1. What is “Control in the Field”?

Control in the Field is a term given to a control strategy where all function blocks for the strategy are incorporated in field devices. Figure 1 illustrates Control in the Field. An Analog Input block is executing in a fieldbus transmitter and PID and Analog Output blocks are executing in a digital valve controller. All of the control is located in the transmitter (PT-01) and the digital valve controller (FV-01). DeltaV controller execution rate and I/O scan rate do not affect the control performance.

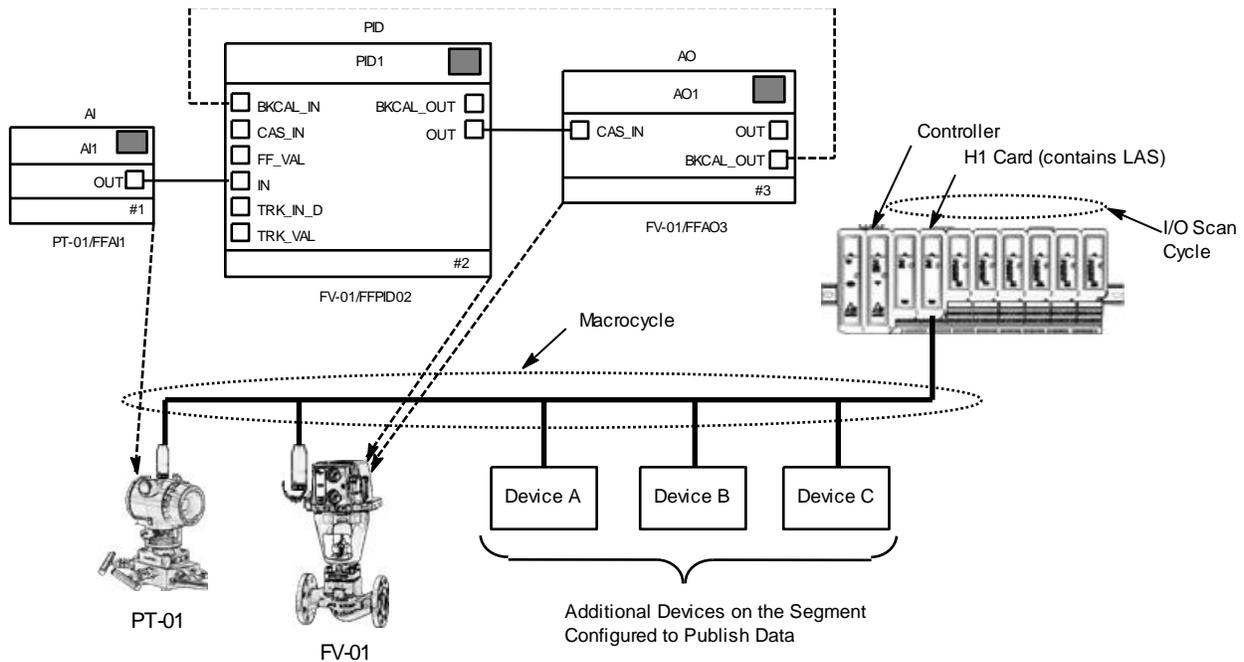


Figure 1 Control in the field

Question 2. What is the interaction between controller execution rate and the Macrocycle when Control in the Field is used?

In Control in the Field, the controller execution rate has no affect on actual loop performance because the scheduling of the function blocks in the device is driven from the Macrocycle and not from the controller execution rate. In Figure 1, the controller is not providing any control to the loop. However, the controller is passing ViewList data for presentation on operator displays. Running the module slower than the Macrocycle is recommended for good data presentation.

Question 3. What is “Hybrid Control”?

Hybrid control is a term given to a control strategy that incorporates function blocks running in both field devices and a DeltaV controller. When more functionality is required than device-based function blocks provide or when function blocks are needed which are not included in devices, DeltaV controllers are used because they contain the needed functionality that devices cannot deliver.

Figure 2 illustrates Hybrid Control. An Analog Input is executing in a fieldbus transmitter, a PID block is running in a DeltaV Controller, and an Analog Output block is executing in a digital valve controller. Both the AI block and the AO block are publishing data to the H1 card, and the H1 card is publishing data to the AO block. The H1 card contains the LAS (one LAS at each port, operating independently) and is transferring the published data per the LAS schedule. Timing of the controller execution rate, I/O scan rate, and Macrocycle scan rate is asynchronous. To obtain acceptable control loop performance and ViewList data presentation, the rates must be coordinated.

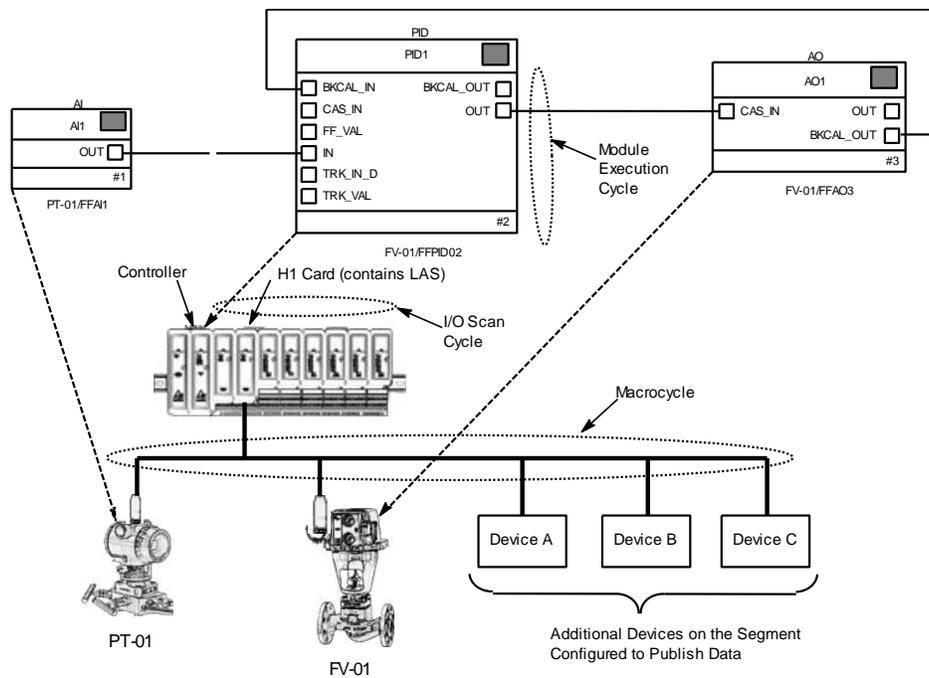


Figure 2 Hybrid control

Question 4. What is the interaction between controller execution rate and the Macrocycle when Hybrid Control is used?

In Hybrid Control, the controller execution rate affects overall loop performance. The execution of the controller function blocks is driven by the controller execution rate. For the blocks to obtain timely operating data, the Macrocycle scan rate needs to be coordinated with controller execution rate.

If the Macrocycle is too fast, AO will execute without receiving the latest data from the controller. AO will regularly see “stale” data. If the Macrocycle is too slow, PID data will be regularly overwritten in the H1 card before it is transmitted (at the LAS schedule).

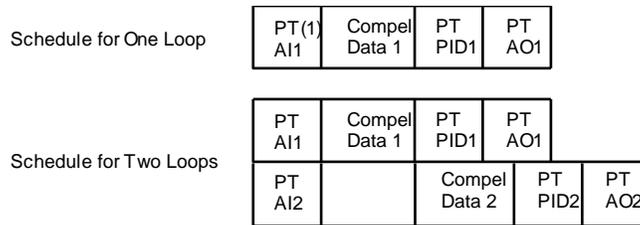
For example, if both the controller execution rate and the Macrocycle are set to one second, scheduled information (upon a Compel Data command) should get to and from the H1 card at precisely 1 second. However, the controller running at one second has an added latency due to scanning the I/O subsystem (which is done asynchronously). The latency is very small, but it can cause the controller's cycle to not match the LAS's Macrocycle.

When a mismatch occurs, the H1 card buffer may not be updated before its contents are transferred (at the LAS schedule). If the buffer is not updated and the data has changed, the controller will not operate on the latest data; it will operate on the same data two times or more, in succession.

A general rule of running the controller execution rate at half the rate of the Macrocycle is recommended as a starting point to make sure that the latest data is transferred. After loop startup, the execution rate can often be adjusted for optimum loop performance.

Question 5. How does the Macrocycle schedule multiple control loops on a segment?

Function blocks in different devices can be scheduled to execute simultaneously. Only the publishing of scheduled data on the bus must be done in sequence. So, running multiple loops on the same segment can be very efficient. See Figure 3.



Note: 1. PT = Pass Token. PT is held by a device when it has permission to send unscheduled data.

Figure 3 Comparison of schedule for running a single loop and running two loops on the same segment

Question 6. How does the LAS schedule data transfer?

At a device download, the LAS provides each device on the related segment with a schedule for Compel Data commands. Between Compel Data commands, each device performs its functions. Upon receiving a Compel Data command, each device broadcasts its scheduled data to all devices which have been configured to receive the data. As devices are added to a segment, the LAS can re-arrange schedules for optimum communication efficiency.

During times when the segment has no scheduled communication, the LAS sends a pass token (PT), among other messages, to a device in the LAS's LiveList which gives permission for the device to send data. The device sends the data and returns the token to the LAS, which in turn, passes the token to the next device on its LiveList. This operation continues until the next Compel Data command is sent by the LAS.

If the LAS determines that the time remaining between the return of the pass token and the next Compel Data command is too short to allow for adequate unscheduled data transfer, the LAS holds the pass token and issues an idle message until the next time that a pass token is sent. At that time, the LAS issues the pass token to the next device in the LiveList. The idle message is used to prevent the backup LAS from taking over.

Question 7. How do Scheduled Data Transfers work?

The Link Active Scheduler (LAS) maintains a schedule, called the Link Schedule or LAS schedule, which is a list of transmit times for all of the data that needs to be cyclically transmitted (Scheduled Data Transfers). When it is time for a fieldbus device to send the data, the LAS issues a Compel Data message to the device. When the device receives the message, it broadcasts or publishes the scheduled data to all devices on the segment configured to receive the data. Such devices are called subscribers.

Question 8. How do Unscheduled Data Transfers work?

When the Link Active Scheduler (LAS) detects a communication gap large enough to send unscheduled data, it sends a PassToken (PT) to a device on the bus with the amount of time the device can hold that token and transmit data.

The LAS rotates the token through the device list in the order of the device data link node address. When a device, including the LAS, receives a PT, the device can send data until it has finished or until the maximum token hold time has expired; whichever is the shorter time.

The time it takes for a token to rotate through all devices on a fieldbus segment is called the Actual Token Rotation Time (ATRT). The ATRT may vary according to the traffic on the bus. The ATRT is not tied to the Macrocycle duration but the Macrocycle scan rate does have an indirect effect on unscheduled data transfers.

As more devices are added to a segment, the more entries there are in the LAS schedule. More entries increases the token rotation time and decreases the unscheduled time available in a Macrocycle. As a result, devices on a segment, including the LAS, are not able to send unscheduled messages as frequently as they could if the segment was lightly loaded. If this occurs, control loop performance can become degraded.

Question 9. What happens if the controller execution rate is faster than the Macrocycle?

ViewList data can only be sent when a device holds the token, and data from only one function block can be sent at a time. For example, in one rotation of the token, the controller sees the latest PID, but does not have new AO data until the next time that the device has the token. This will produce a MissedViewListScan because the module executes too quickly. Whether the data is being used for ViewList only, or for both control and ViewList, the module is providing “stale” ViewList data to the operator display and is operating on “stale” data.

Question 10. What factors influence calculation of the Macrocycle?

Macrocycle calculation depends on several factors in a system configuration. The Macrocycle is built by sequencing all of the linked blocks on the fieldbus segment that require scheduled data transfers. Some linked function blocks reside in the same device and therefore do not need to use the bus.

Also affecting the Macrocycle is data flow between blocks, when and how long each block executes, and the time required to transfer data between blocks. Further, the Macrocycle takes into account reserved spaces for asynchronous data transfers, such as pass token requests, prob nodes, time distribution, and so forth.

Factors affecting macrocycle can include:

- Types of devices

- Types of function blocks running in the devices
- Function block execution order
- Device-to-device links
- Links between function blocks running in Fieldbus devices and function blocks running in the DeltaV controller
- Extra time allocated to unscheduled communication
- Some suppliers function blocks running faster

Question 11. Can the Macrocycle be computed in the design phase?

The Macrocycle can be estimated during the design phase. The actual Macrocycle is calculated by the DeltaV system, using a proprietary algorithm, after the Fieldbus segment has been configured. Currently, there is not a method available to exactly calculate the Macrocycle outside of having the DeltaV system do it.

To estimate a Macrocycle, Schedule Macrocycle, Required Macrocycle, and Actual Macrocycle must be understood. Schedule Macrocycle is the configured rate for one iteration of the Compel Data schedule. It can be configured for future expansion of a Fieldbus segment by selecting a Macrocycle that is larger than the required Macrocycle. However, a larger Schedule Macrocycle will cause the control loop to run slower since function blocks will not execute as often. The Schedule Macrocycle field is a drop-down list in the DeltaV Explorer with five options: 250 ms, 500 ms, 1 sec (default), 2 sec, and 5 sec.

Required Macrocycle is the minimum Macrocycle time while Actual Macrocycle is the time between consecutive iterations of the fieldbus schedule. The Actual Macrocycle is the greater of the Schedule Macrocycle or the Required Macrocycle. The Actual Macrocycle is determined by the factors listed in Question 10.

You might simulate your system operation using the following steps to get a sense of how these factors work together. Refer to Fieldbus Configuration Procedures in DeltaV Books Online if you need help with any of these steps.

- Use DeltaV Explorer to add an H1 card and several devices.
- Use Control Studio to configure a control strategy composed of AI, AO, and PID blocks.
- Assign the blocks to Fieldbus, assign the module to a node, and save the module.
- In the DeltaV Explorer, select the Fieldbus port, select Properties from the Context menu, select the Advanced tab, and read the Required Macrocycle value.
- You can go back to Control Studio, add an external link and connect the link to a Foundation fieldbus function block, and save the module.
- Reread the Required Macrocycle value in Explorer. It should be a higher value.
- To observe effects on the macrocycle, change the device revision, add more function blocks, adjust function block execution rates, and add more links. With each addition or change, check the Required Macrocycle value.

Question 12. What does MissedViewListScan indicate about system operation?

MissedViewListScan is a diagnostic parameter. If it is incrementing, segment communications should not be seen as overloaded. Scheduled data transfers are guaranteed to get through. An incrementing **MissedViewListScan** parameter simply is an indication that **ViewList** data to the controller for operator display is not getting updated as frequently as the controller is requesting it. It is difficult to quantify what is acceptable before it is recommended to slow down the controller execution rate.

As more devices are added to a segment and, more importantly, as more function blocks per device are configured, **MissedViewListScans** can start occurring. The general rule of running the controller execution rate at half the speed of the Macrocycle has been found to eliminate **MissedViewListScans** in most situations.

When **Control in the Field** is used, a controller can execute slowly because the controller is not included in the control loop. Therefore, from the perspective of the loop, there is no need to configure a module such that misses occur, although **MissedViewListScan** may still occur during heavy communication usage, such as for downloads.

In practice, there is practically no functional difference between a controller running at one second with a high number of **MissedViewListScans** compared to the same controller running at two seconds with no **MissedViewListScans**. Controller execution rate can be set for best operator display update time (faster rate) while minimizing use of processor resources (slower rate).

When **Hybrid Control** is incorporated, the most important criterion to consider is control loop performance. It is generally acceptable to expect some **MissedViewListScans** to achieve best control loop performance. Experience has shown that, if **MissedViewListScans** are within 25% of requests sent, display updates are not adversely affected.

Question 13. What is a NumDIITokenPass Timeout, and how should this statistic be interpreted?

A key statistic is **NumDIITokenPass Timeouts**. It is a DeltaV Device Communications Statistic that can be found in DeltaV Diagnostics (by expanding a fieldbus port under the I/O subsystem to expose the connected devices). If this statistic increments, it means that the fieldbus device did not receive the **PassToken (PT)** message that was sent to it, the H1 card did not receive the token pass response, or the device held the token longer than the **TokenHoldTime**. If a token response is not received in three consecutive token rotations to the same device, the LAS removes the device from the **LiveList**.

Ideally, the **NumDIITokenPass Timeouts** parameter does not increment during normal operation. However, as with any communications network, noise on the bus or a sensitive device receiver may cause a token or token response to be missed. These situations cause the **NumDIITokenPass Timeouts** and **DII Retries** parameters to increment. Sources of noise can include electromagnetic radiation from high energy motors and other electrical sources, improper electrical grounding, and out-of-tolerance device transmitters. As a general rule, if this parameter is around 1% or less of **Requests Sent**, further troubleshooting is not normally required unless **NumLiveListAppearances** is incrementing.

Question 14. What is the relationship between Function Blocks and Shadow Blocks?

A fieldbus segment connects to a DeltaV system through an H1 card and is configured from a suite of pre-configured function blocks using Control Studio. When function blocks are assigned and downloaded to a fieldbus device, the function block algorithms execute in the device and the blocks that appear in Control Studio (in On Line mode) are merely a “shadow” of the function blocks. Hence, the term “Shadow Blocks”. Shadow blocks are updated with **ViewList** data at the controller scan rate. Shadow blocks do not need to reside in a single module.

Question 15. What is the DeltaV implementation of the FOUNDATION fieldbus standard with regard to data link layer and device timer values?

The Foundation fieldbus communications profile states that the default value for MaxResponseDelay and InterPDUdelay are 10 and 16 but states that "support for smaller values is recommended". Smaller values are also referred to as "faster" values because they improve performance.

InterPDUdelay defines how closely Protocol Data Units (PDUs) can be spaced on the bus. Closer spacing means that more bus time is available for transferring data. The MaximumResponseDelay defines the maximum time that a device has to respond to a poll for scheduled data (a CompeIData PDU). Smaller MaxResponseDelay and InterPDUdelay means that entries in the LAS schedule can be packed closer together, thus providing more unscheduled time. Devices are configured with these values when they are first detected on the bus by the LAS. They are not allowed to join the bus if they cannot operate within these values.

DeltaV systems follow the Fieldbus Foundation recommendation so that users obtain better performance from their fieldbus system. DeltaV technology has determined these values through device testing.

Question 16. What items should be considered to obtain an efficient fieldbus implementation?

Several rules of thumb are applicable when designing a fieldbus system. Efficient designs accomplish needed functionality while minimizing communication traffic. Helpful rules are:

- Whenever possible, use Control in the Field.
- Use DeltaV controllers (Hybrid Control) only when functions needed for your control system are not available in a smart field device.
- Use one control entity per control module.
- When writing to fieldbus parameters, write only if there is a change in the parameters (exception reporting), and do it sparingly.
- Minimize the number of external references to fieldbus devices. An external reference is normally an input parameter read from one function block to another and requiring communication across the segment or the H1 card.

Question 17. Why are certain Fieldbus Network Parameters pre-set for DeltaV systems?

The Fieldbus Network Parameters listed below have been factory-tested for DeltaV systems. They contain pre-set values, determined by empirical tests, chosen to deliver the most robust H1 implementation, reducing the chances of errors in the field.

■ APCLCKSYNCNTRVL

ClockSynchronizationInterval is the interval at which the primary time master provides System Management clock synchronization messages to devices.

Pre-set value is 10 seconds.

■ DFMNTKNDLGTM

DefaultMinimumTokenDelegationTime is the minimum amount of time to give to a device to transfer data with a single **PassToken**. The LAS cannot send a token to a device if a gap in its schedule cannot accommodate the device holding the token (transferring data) at least this long.

Pre-set value is 96. (One unit is 256 microseconds).

■ DFTKNHLDTM

DefaultTokenHoldTime is the initial amount of time to allocate to a device in one cycle of circulating the token, which may include multiple **PassToken** Protocol Data Units.

Pre-set value is 288. (One unit is 256 microseconds).

■ FRSTUNPLDNDID

FirstUnpolledNode prevents the Link Active Scheduler from wasting time polling nodes that are not used, as provided for by the Fieldbus data link layer. DeltaV systems only use a portion of the total number of Fieldbus Node Addresses.

Pre-set value is 36.

■ INTPDUDLY

MinimumInterProtocolDataUnitDelay is the minimum gap between PDUs. It ensures that more than one device is not attempting to communicate at the same time, and that all devices are capable of recognizing the start of a new PDU after the end of the previous one.

Pre-set value is 12. (One unit is 256 microseconds).

■ LASDBSTTSSPDUPR

DatabaseStatusDistribution is the period with which the Link Active Scheduler broadcasts LAS Database Status on the bus to synchronize the LAS management information with all Backup Link Active Schedulers.

Pre-set value is 1000 milliseconds.

■ LNKMNTTKHLDTM

LinkMaintenanceTokenHoldTime is the initial amount of time to allocate to LAS related maintenance activities in one cycle of the circulating token. These activities include probing for new devices, measuring round-trip delays to devices, and conveying updated schedule information to other Backup Link Active Schedulers in case of possible LAS transfer.

Pre-set value is 0. (One unit is 256 microseconds).

■ LNKTMSYNCLSS

TimeSyncClass determines the accuracy required for FMS (Fieldbus Message Specification) clock synchronization and FAS (Fieldbus Access Sublayer) time distribution.

Pre-set value is 0.

■ MAXRSPDLY

Maximum Response Delay defines how long a device has to respond to a **CompelData** or to start transmitting after the receipt of a token.

Pre-set value is 6. (One unit is one slot time).

■ MCRCYCLEDRTN

This parameter is not used in DeltaV systems.

■ MXNCTCLMLASDLY

MaximumInactivitytoClaimLASDelay is the maximum time of bus inactivity before a device can attempt to claim the LAS. A timer is started in each device capable of claiming the LAS each time it detects that a PDU is sent on the bus.

Pre-set value is 100. (One unit is 32 microseconds).

■ MXNTRCHNSGNLSKW

This parameter is not used in DeltaV systems.

■ MXSCHDLOVRHD

MaximumSchedulingOverhead indicates the maximum scheduling overhead that can be used by the LAS on a link schedule. The overhead is included in the time allotted for each scheduled activity, and so is used only during schedule construction and in the determination of whether a DLE can serve as LAS for an existing schedule. The overhead is a part of the macrocycle.

Pre-set value is 0. (One unit is 256 microseconds).

■ NMUNPLDNDID

NumberOfUnpolledNodes indicates the number of nodes that will not be probed for inclusion onto the **LiveList** starting from **FRSTUNPLDNDID**.

Pre-set value is 196. (Units are device count).

■ OINTEG

This parameter is not used in DeltaV systems.

■ PERDLPDUOVHD

PhysicalLayerInducedDelay is the delay between the end of the last octet of a PDU and the beginning of the first octet of the next PDU.

Pre-set value is 4. (One unit is 256 microseconds).

■ PRMBLXTNSN

PhysicalLayerPreambleExtension is the amount by which the preamble should be extended. However, not all devices can operate with extended preambles. The preamble (10101010) synchronizes the receiver to accept data.

Pre-set value is 0.

■ **PSTTRNSGPXTNSN**

PhysicalLayerPostTransmissionGapExtension is the amount by which the mandatory post-transmission gap, which is the period of non-transmission between successive sequences of Physical PDUs should be extended.

Pre-set value is 0.

■ **SLOTTM**

SlotTime is the basic device time unit used in the LAS recovery algorithm and other DDL timing algorithms.

Pre-set value is 11. (One unit is 256 microseconds).

■ **STATUS**

Status of the port. This parameter is not currently used as part of DeltaV Explorer. To get the actual port status, view this parameter from DeltaV Diagnostics.

■ **T1, T2, T3**

These are timers. Their values are used during the commissioning of Fieldbus devices. The device vendor sets the **T2** timer value. Even though DeltaV shows a **T2** value in Explorer, it is not used because DeltaV systems assign their own H1 addresses locally. **T1** and **T3** values have been selected based on rigorous testing of a large number of devices on loaded and unloaded segments. If these values are increased, commissioning times also increase. If these values are decreased, it may not be possible to commission devices at all.

T1 pre-set value is 240000. T2 pre-set value is 1920000. T3 pre-set value is 1280000. (Unit is 1/32 milliseconds.)

■ **TMDSTPRD**

TimeDistributionPeriod is the period that the Link Active Scheduler uses to transmit time distribution messages. After the expiration of this time period, the LAS will send a time distribution message at its next opportunity.

Pre-set value is 5000 milliseconds.

■ **TRGTTKNRTTM**

TargetRotationTime is the upper bound limit on time required for one full rotation of the PassToken message to all devices on the H1 segment.

Pre-set value is 12288 milliseconds.

This page intentionally left blank.

To locate a sales office near you, visit our website at:

www.EmersonProcess.com/DeltaV

Or call us at:

Asia Pacific: 65.6777.8211

Europe, Middle East: 41.41.768.6111

North America, Latin America: +1 800.833.8314 or

+1 512.832.3774

© Emerson Process Management 2013. All rights reserved. For Emerson Process Management trademarks and service marks, go to:
<http://www.emersonprocess.com/home/news/resources/marks.pdf>.

The contents of this publication are presented for informational purposes only, and while every effort has been made to ensure their accuracy, they are not to be construed as warranties or guarantees, express or implied, regarding the products or services described herein or their use or applicability. All sales are governed by our terms and conditions, which are available on request. We reserve the right to modify or improve the design or specification of such products at any time without notice.



DELTA^V

www.DeltaV.com



EMERSON[™]
Process Management