Chapter 84

Using the AXN/MBM/401

TEC/NOT/090

This paper discusses the following topics:

84.1 Module overview

The AXN/MBM/401 is a 4-channels dual-redundant MIL-STD-1553 bus monitor which can parse and/or packetize each channel at the same time. The module is MIL-STD-1553-B compatible.

You can either use Direct (DC) or Transformer (TC) coupled operation (through wiring).

Screen shots and descriptions of settings shown in this technical note are from DAS Studio 3 software. DAS Studio 3 is used to create a configuration, which contains the various elements that make up your data acquisition system. You then use this configuration file to manage and program these elements.

To see how hardware is represented in the DAS Studio 3 graphical user interface, refer to the *DAS Studio 3 User Manual*.

NOTE: The AXN/MBM/402 is a similar module but with only two channels. The AXN/MBM/40x does not function with an AXN/BCU/401 controller; instead it must be used with an AXN/BCU/402 (or higher) controller.

84.1.1 Key features

- Monitors up to four dual redundant MIL-STD-1553 busses using Direct or Transformer coupled operation
- Coherently parses traffic and tags for up to 4,095 parser ID; parser ID based on 16-bit commands; RT-RT messages treated as one parser ID
- Aperiodic transmission of packetized MIL-STD-1553 messages including tags as iNET-X or IRIG-106 Chapter 10 payload per channel
- Optional recording of messages with time-out status reply

NOTE: For further information, refer to “84.9.3 Recommended reading” on page 15.

84.2 Parser operation

84.2.1 How parsing works

Like other Curtiss-Wright bus monitors, the AXN/MBM/401 uses a triple buffer for parsing. The following figure illustrates the triple buffering of data words (grey) and time message tags (white) used for each redundant bus in the AXN/MBM/401’s parser.

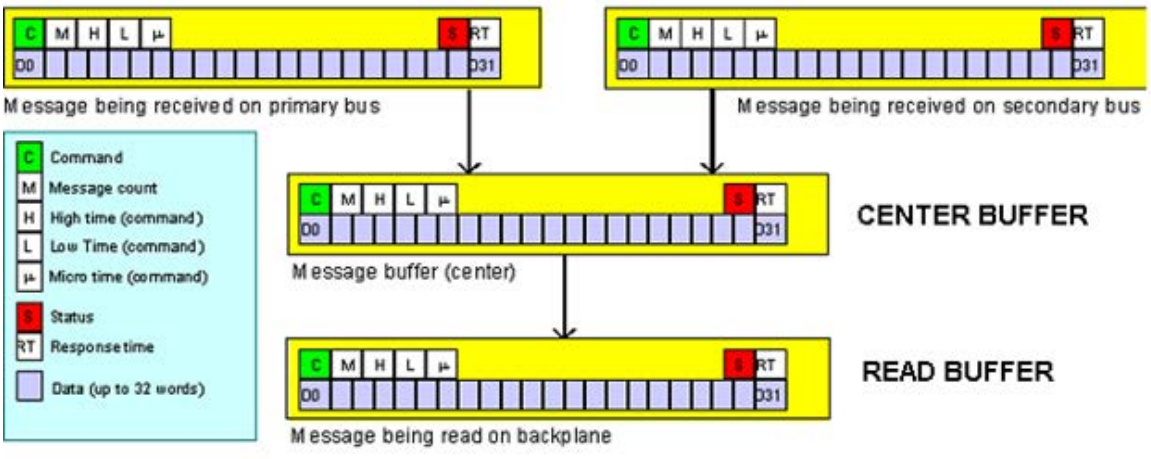


Figure 84-1: Triple buffering of a BC to RT traffic and associated message tags

The received message, made up of the command word(s), status word(s), status response time(s) and up to 32 data words are stored in triple buffered parser memory. A time tag of when the message was received and a count of the number of messages on the received bus, as well a slot info status are also appended to the parser message and can be read as tags. The triple buffer operates in a manner whereby the oldest fresh message is read first, followed by any fresh messages. The parser slot info tag carries the following information about the massage:

- **Empty** indicates that this parser slot has never been written to and no message matching the message definition has been received. All commands (such as status data) return the default fill value when read.
- **Stale** indicates that this message was previously read from the parser slot.
- **Skipped** indicates that this message overwrote a message in this parser slot that was not read on time.
- **Bus** The bus it was received on.

NOTE: It is advisable to always slightly over sample the parser so as to take into account differences between the clock rate of each system. For example if messages are sent 10 times a second and the parser slot is sampled 10 times a second then messages will occasionally be marked stale and skipped. This is due to a beat effect between the relative clocks of the MIL-STD-1553 bus and the Axon Data Acquisition system. Sampling at faster than 10 times a second would always over come this.

The time tags H, L, M correspond to High time, Low time and Micro time, which is the time midway of the first transmitted bit of the message (that is, Mid-way through COMMAND Sync Bit) with a 1-µsec resolution.

The way triple buffering works is as follows:

Time message tags are added to each message received and stored in separate buffers for each of the busses. As soon as a message is received with no errors, the contents of the write buffer is transferred to the center buffer. If the data in the center buffer has not been transferred to a read buffer, a skipped flag is set.

As soon as the last parameter of interest has been read from the buffer being read by the backplane, the contents of the center buffer (if new) are transferred to the read buffer. If no new data word has been received, the stale flag is set. A center and read buffer exist for every message ID (parser slot). Skipped and stale bits can be found in the Message Info register to indicate whether messages are lost or repeated (undersampling or oversampling situations).

Additional tags such as Message Commands, Message Statuses, Message response times, Message Count and Message Info registers are also available as additional information and can be added from DAS Studio's MIL-STD-1553 Builder application as explained in "84.4.2 Variable sized messages" on page 7. For further information regarding these registers, refer to the *AXN/MBM/401* data sheet.

The parser filters the data stream and captures specific words from a specific message. The user has to define precisely what data is being sought. This 'parse what' or message signature content, defined by the user in the bus, monitors a specific RAM address in the Current Value Table (CVT) for each buffer on the triple buffer.

This RAM address is called the Parser ID. The AXN/MBM/401 has 4095 different Parser IDs.

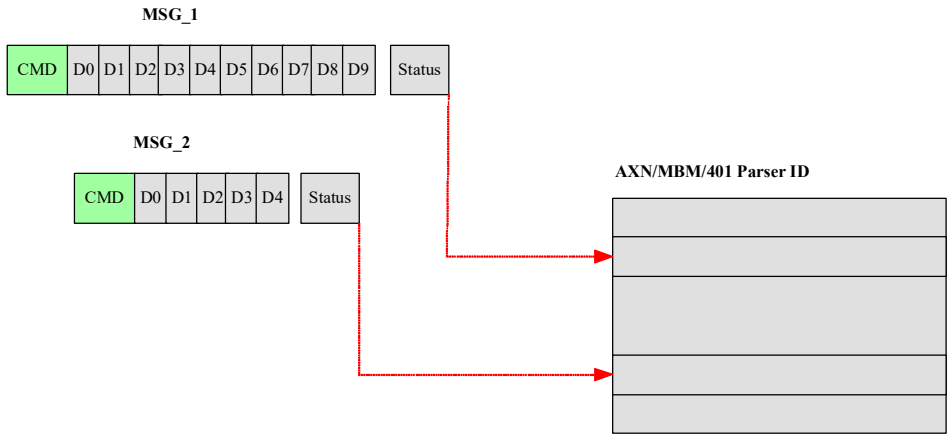


Figure 84-2: Explanation of the Parser ID

84.3 Module Settings tab

84.3.1 Parser settings

In the parser, a total of up to 4095 complete messages are triple buffered so that the message is coherent and the stale/skip indication is message-wide. Each message is tagged to 1 us resolution.

NOTE: To view the screen shots shown in this section in DAS Studio 3, ensure the AXN/MBM/401 module is in context and the Settings tab is selected.

To configure the parser, first you must decide on the parser settings.

You need to specify each field per channel as shown in the following figure.

Source Name	Maximum Response Time	Accept Rx Message With No Status	Accept Tx Message With No Status
MIL-STD-1553-In-A(0)	12	☐	☐
MIL-STD-1553-In-A(1)	12	☐	☐
MIL-STD-1553-In-A(2)	12	☐	☐
MIL-STD-1553-In-A(3)	12	☐	☐

Figure 84-3: Parser settings

Setting	Description
Maximum Response Time	Maximum permitted response time for RT in micro-seconds.
Accept RX Message With No Status	This setting needs to be set to **True** if the bus contains RX messages without status; that is, if any remote terminals monitoring the MIL-STD-1553 bus are likely to be off-line. If set to **No**, then only complete messages, where a remote terminal responds, are captured.
Accept TX Message With No Status	This setting needs to be set to **True** if the bus contains TX messages without status; that is, where the remote terminal does not respond. For this setting to have an effect, AcceptRxMessageWithNoStatus must be set to **Yes**. When set to **No**, the AXN/MBM/401 rejects Tx command messages where the remote terminal does not respond. The effect on the parser is that the last valid message with the last valid data and time stamps remains.

The above settings are important if the bus contains messages with no Status. The following figure shows an example on a bus for which a message BC-RT has no Status but the next message RT-BC. In MIL-STD-1553, the command word (CMD) and status word (STS) have the same signature (see section “3.3 Word definition” in *TEC/NOT/004 - MIL-STD-1553*), this means the CMD and STS can only be differentiated by the context or protocol.

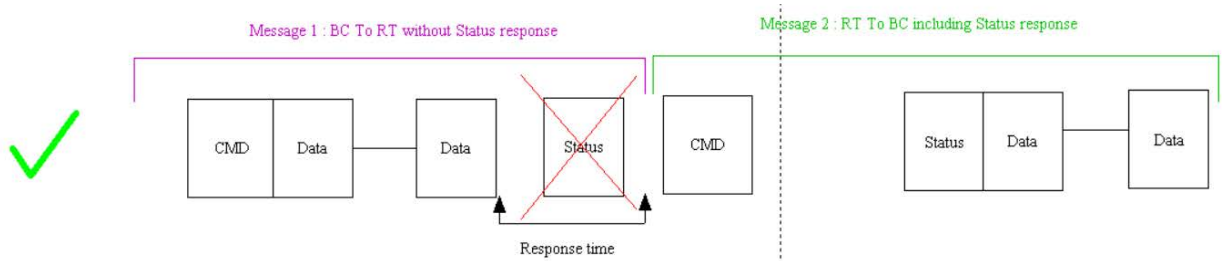


Figure 84-4: Example of a BC-RT without status with the correct response time set

For example, if the response time is set too high, and if there is no response from a particular RT, the next command word may belong to that message as a status word. This may cause issues parsing the next message.

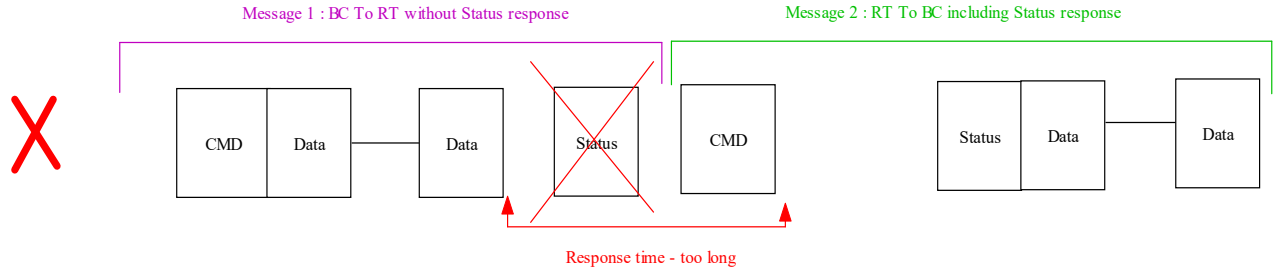


Figure 84-5: Example of a BC-RT without status with response time set too long

If the response time is set too low, then the AXN/MBM/401 may consider that a valid message has timed out due to no response. The status of this message appears as the next command. This may cause issues parsing the next message and so on.

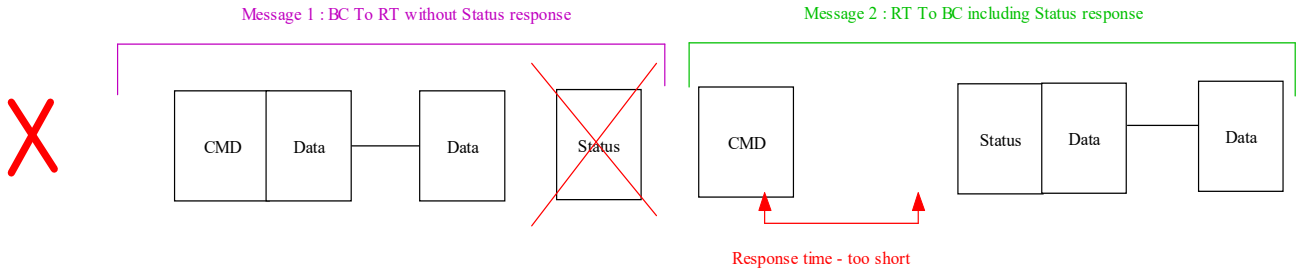


Figure 84-6: Example of a BC-RT without status with a response time set too short causing a time out

To detect when a Tx remote terminal is off-line, set AcceptTxMessageWithNoStatus to **Yes**. The parser status word must be sent to detect this and then be processed in the ground station. If the status word is hex FFFF, the ground station must deal with the invalid data in the parser. Otherwise, the data shows spikes.

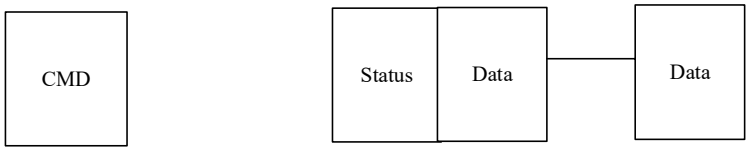


Figure 84-7: Example of a RT-BC with status and data

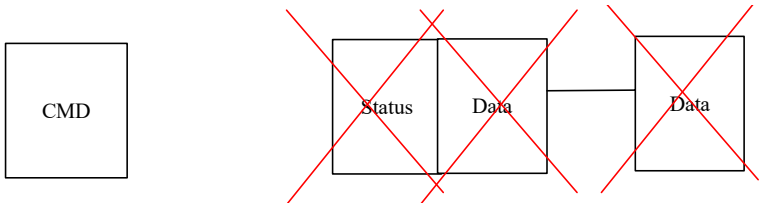


Figure 84-8: Example of a RT-BC without status nor data

You can also set a value on the Fill Value as shown in the following figure.

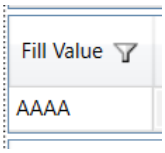


Figure 84-9: Fill Value setting

All parser slot parameters (data and tag) except MessageInfo display this value until a parsed message is received in a channel, the Fill Value indicates no data has been received.

84.3.2 Global parameters settings

Like most bus monitors, the AXN/MBM/401 has global parameters, which can be used for debugging. The main global parameters are Report word and several counters such as message, error and WordsPerSecond counters for each bus and redundant bus as described in the following table.

MyAXN_MBM_401	Report	***P_MyAXN_MBM_401_Report***
MyAXN_MBM_401	BusActive	***P_MyAXN_MBM_401_BusActive***
MIL-STD-1553-In-A(0)	MessageCountA	***P_MyAXN_MBM_401_MIL-STD-1553-In-A(0)_MessageCountA***
MIL-STD-1553-In-A(0)	ErrorCountA	***P_MyAXN_MBM_401_MIL-STD-1553-In-A(0)_ErrorCountA***
MIL-STD-1553-In-A(0)	WordsPerSecondA	***P_MyAXN_MBM_401_MIL-STD-1553-In-A(0)_WordsPerSecondA***

Figure 84-10: Global parameters in DAS Studio 3 Settings tab

Setting	Description
Report	The Report word is a 16-bit register, which provides information regarding errors detected in the card and the bus. The Report word is recommended to be monitored as a debug register when abnormal conditions are detected. Refer to the *AXN/MBM/401* data sheet.
BusActive	Indicate if a valid message from the protocol tracker was received on a channel/bus since last read. Bit 0 corresponds to CH(0)_A, bit 1 = CH(0)_B up to Bit 6 = CH(3)_A, Bit 7 = CH(3)_B.
MessageCountA	Counts valid messages from the protocol tracker received on a channel/bus. There is a separate counter for bus A and redundant bus B.
ErrorCountA	Counts messages with errors. There is a separate counter for bus A and redundant bus B.
WordsPerSecondA	MIL-STS-1553 words received on the bus in the last second. There is a separate counter for bus A and redundant bus B.
ModuleTemperature	Temperature of the AXN/MBM/401 module. Refer to the *AXN/MBM/401* data sheet.

84.3.3 Errors

There are several errors reported by the AXN/MBM/401. As explained in the previous section, the Report word provides an indicator of the different errors detected in any of the busses, and ErrorCount provides an indicator of the number of errors detected on each bus. Additional packetizer headers contains an error flag Er (1 bit) and a 6 bits Error Code indicator.

The supported errors are defined in the *AXN/MBM/401* data sheet.

84.4 MIL-STD-1553 Builder

For the following sections, use the MIL-STD-1553 Builder application in DAS Studio 3. Refer to the "Builder application GUI overview" section of the *DAS Studio 3 User Manual* for a brief overview of navigating the application.

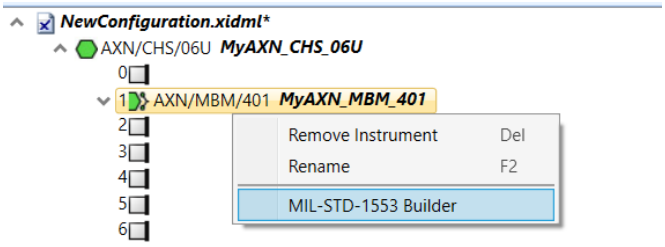
"84.4.1 Defining parsing rules" on page 6

"84.5.2 Adding parameters to the package" on page 9

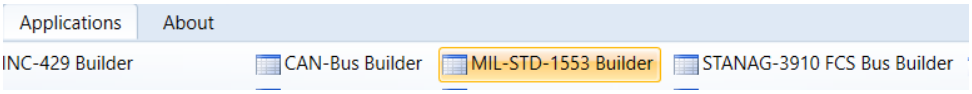
84.4.1 Defining parsing rules

After you have all channel settings configured, refer to the following to define rules to identify messages.

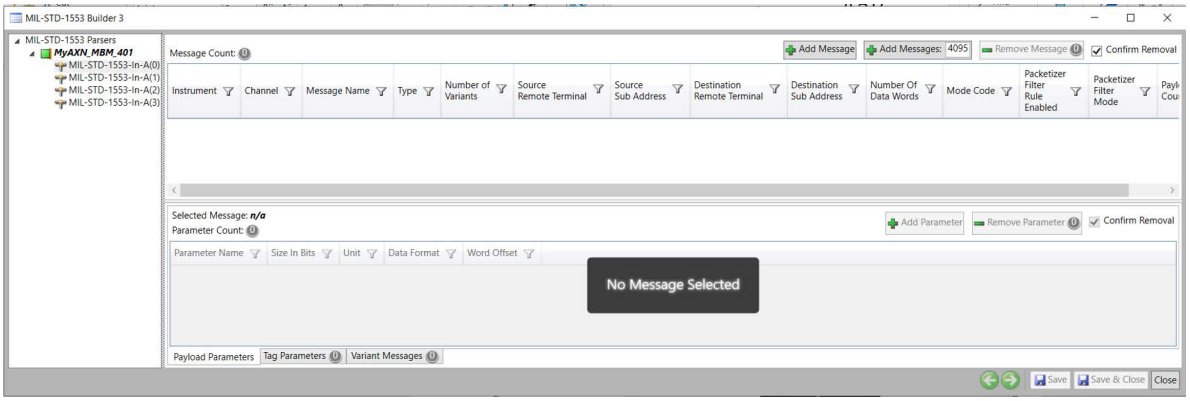
1. Do one of the following.
   - In the Navigator, right-click the AXN/MBM/401 module and then click **MIL-STD-1553 Builder**.



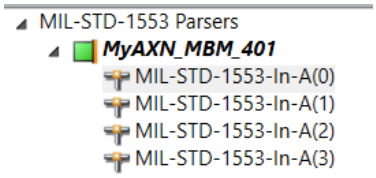
   - On the **Applications** tab click **MIL-STD-1553 Builder**.



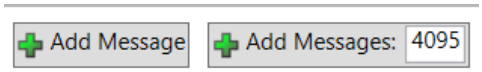
   The **MIL-STD-1553 Builder** application opens.



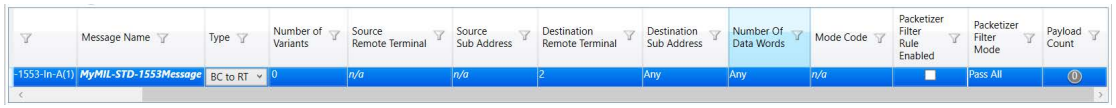
2. In the Navigator (left pane), select the channel on the AXN/MBM/401 that you want to parse data off.



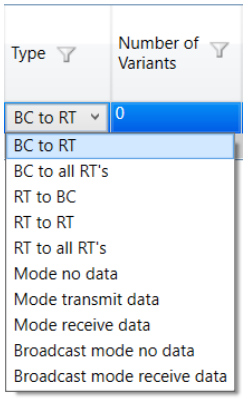
3. Click the **Add Message** button to add a single package. To add multiple packages (up to 1024), click the **Add Messages** button (typing the number of packages in the field).



Now you must define the rules to identify or parse the desired message.



4. The first rule to define is **Type**. In the **Type** field, click the drop-down list.



Depending on the type, the Source/Destination Remote Terminal, SubAddress, Number Of Data Words, and Mode Code are accessible.

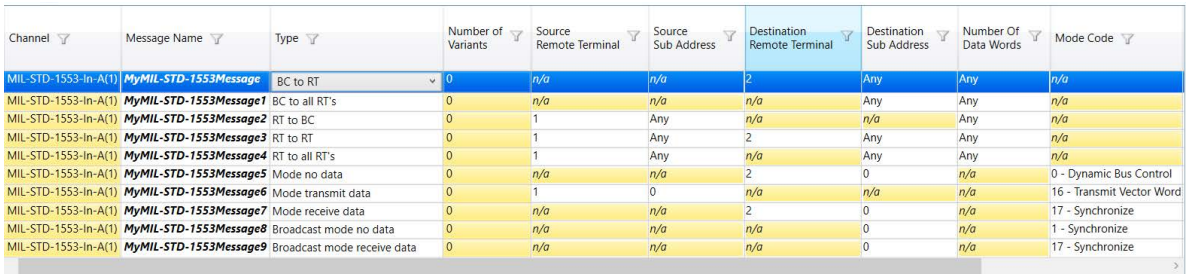


Figure 84-11: Example of the setup of each type available in the AXN/MBM/401

84.4.2 Variable sized messages

By default, the AXN/MBM/401 is configured to parse any number of data words for a particular message (see Number of Data Words in previous figure).

The Number Of Data Words is defined in the CMD word (refer to “84.1 Module overview” on page 1), which enables the module to parse messages of particular sizes. In cases where there are variable sized messages and when **Number Of Data Words** is set to **Any**, care must be taken when processing the data.

The parser only updates the data words that appear on the MIL-STD-1553 bus. The other words are not valid. This means that the ground station or post processing must use the number of data words indicated in the command word to identify which data words are valid. The other option is to parse particular message sizes individually.

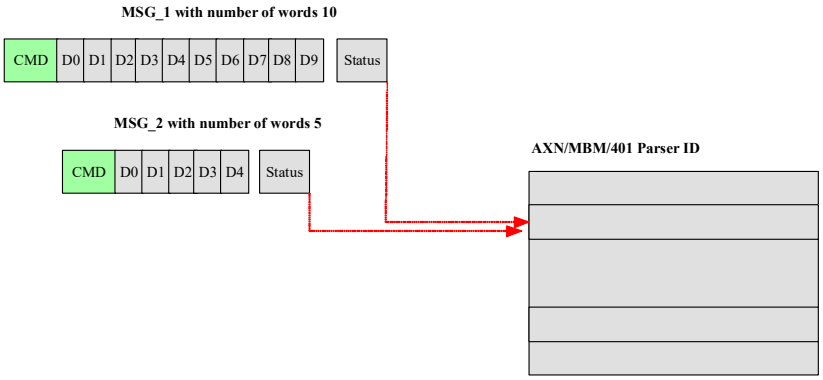


Figure 84-12: Parser slot example using Any for number of data words on two messages with the same type/RT/SubAddress but with different number of words

84.5 Packetizer rule per message

The module supports filtered packetizer per channel. To configure the messages you want to filter, you must configure the following two fields: **Packetizer Filter Rule Enabled** and **Packetizer Filter Mode**

Packetizer Filter Rule Enabled must be selected to allow the message to be either **Blocked** or **Passed** in the packetizer (depending on the channel **Packetizer Filter Mode** set in the setting tab) when they meet the packetizer filtering condition referenced by this process.

NOTE: **Packetizer Filter Rule Enabled** and **Packetizer Filter Mode** are only configurable when the **Packetization Enabled** check box is selected.

Packetizer Filter Mode can then be specified.

Block By Rule means that messages that match defined filter rules are blocked and all other messages are passed through in the packetizer.

Pass By Rule means that messages that match defined filter rules are passed through and all other messages are blocked in the packetizer.

Pass All means that all messages are passed through the packetizer.

NOTE: **Packetizer Filter Rule Enabled** and **Packetizer Filter Mode** are only configurable when the **Packetization Enabled** check box is selected.

Packetizer Filter Mode is a channel setting; this field must be set to the same rules in both the **Settings** tab and **MIL-STD-1553 Builder**. When there is a discrepancy between MIL-STD-1553 Builder and the channel settings, DAS Studio automatically changes **Packetizer Filter Mode** for all messages in the same channel and in the Settings tab as shown in the following figure. This prevents errors in a configuration.

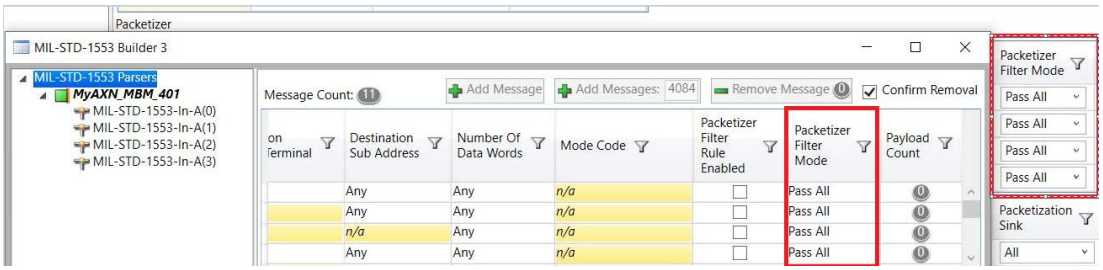


Figure 84-13: Packetizer Filter Mode settings propagation

84.5.1 Adding variants

Variant messages are messages stored in the same memory as an existing message. When a message has a payload parameter, it can have variant messages attached to it. These variant messages cannot be existing messages and they cannot have payload parameters. When a variant message is received, it is stored in the same memory as its parent message, that is, the message with payload parameters.

For the following settings it means the messages RT-BC from RT 2 and RT 3 are using the same parser ID.

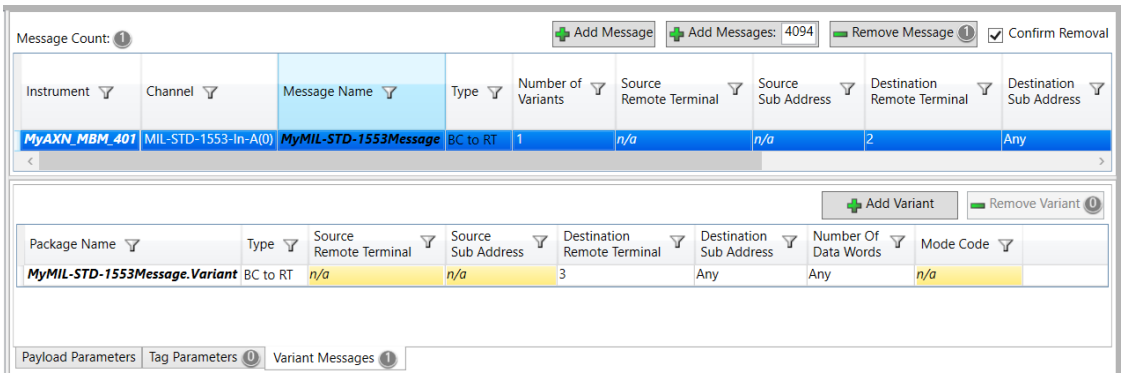


Figure 84-14: Example of a BC-RT RT=2 and its variant RT=3

Any defined parameters are the same for the message and the variant because they share the same Parser ID.

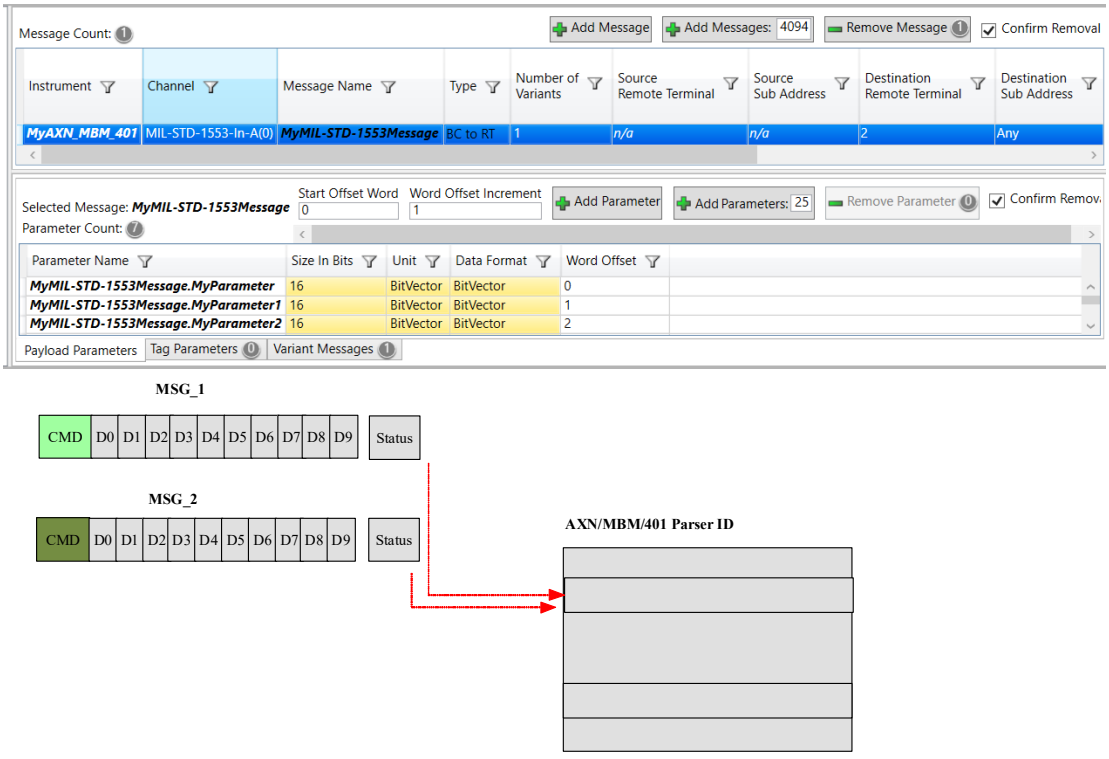


Figure 84-15: Messages with a different parser sharing the same Parser ID

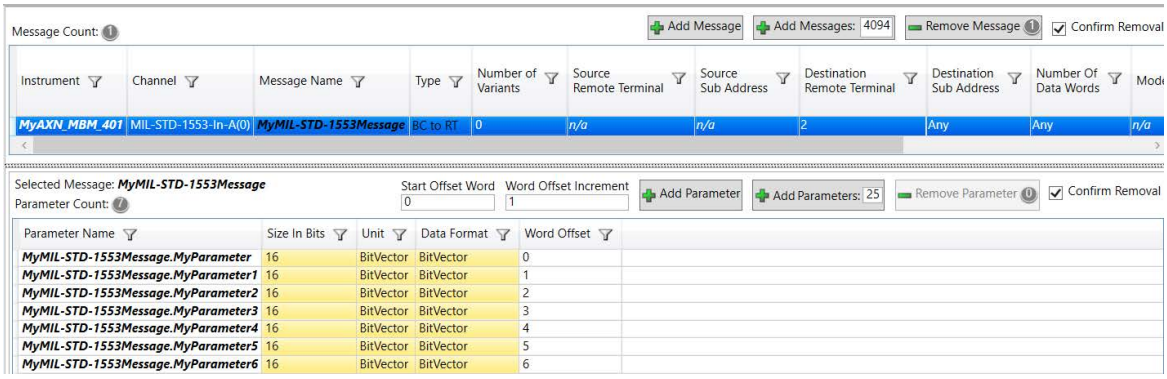
The application is for sensor redundancy: if the bus controller sees an issue with a sensor, it can dynamically switch to the redundant sensor.

84.5.2 Adding parameters to the package

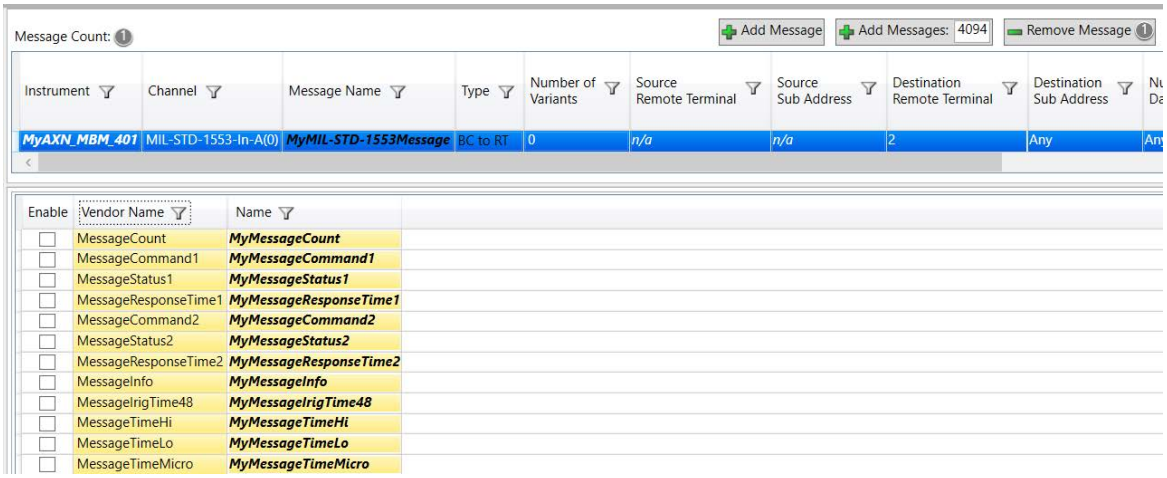
After you have defined rules to identify a message, refer to the following to select the number of bytes required to be parsed.

1. Click **Add Parameter** to add a single parameter. To add multiple parameters, click **Add Parameters** (typing the number of parameters in the field). Up to 32, 16-bit parameters can be defined for every message.

In the following example, six parameters are added.
Message with six data words starting from offset 0 and increasing by 1 position.



2. To tag a message, select the message and then click the **Tag Parameters** tab.



The tags associated with the message are described as follows:

Setting	Description
MessageIrigTime48	MessageIrigTime48 is a 48-bit register consisting of three 16-bit time registers: TimeHi, TimeLo, and TimeMicro. Represents the time stamp of a valid parsed message.
TimeHi, TimeLo and TimeMicro	Same information as MessageIrigTime48 but split in three 16-bit registers. Note: These registers are implemented to provide compatibility with legacy systems.
MessageCommand1 and 2	Command word parsed. MessageCommand2 is for the second RT, for example for RT-RT type.
MessageStatus1 and 2	Status word parsed. MessageStatus2 is for the second RT, for example for RT-RT type.
MessageResponseTime1 and 2	Status word's response time in microseconds. MessageResponseTime2 is for the second RT, for example for RT-RT type.
MessageCount	Counter of the received messages.
MessageInfo	Stale/skipped indication for each parsed message. These flags indicate whether messages are repeated or lost, that is, oversampling or undersampling situations respectively.

Refer to the *AXN/MBM/401* data sheet for further information regarding these additional tags.

3. To save your changes and close **MIL-STD-1553 Builder**, click **Save & Close**.

When package building in MIL-STD-1553 Builder is complete, these parameters become available to be placed, for example in a PCM stream.

84.6 Packetizer operation

Independently of the parser, when **Packetization Enabled** is selected, an iNET-X or IRIG-106 Chapter 10 packet stream is generated for each channel. These parser-aligned packets may be transmitted aperiodically to optimize network bandwidth utilization and memory usage when recording 1553 traffic.

Click the Packetization drop-down menu to see available settings to configure or tune packetizer behavior.

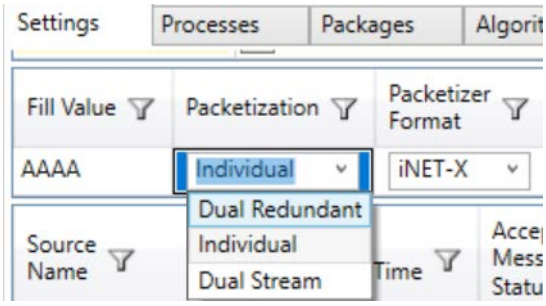


Figure 84-16: Packetization settings on Setting tab

84.6.1 Packetization settings

Individual means that a packetizer stream is generated per bus using the Packetizer settings on bus MIL-STD-1553-In-A and on bus MIL-STD-1553-In-B respectively.

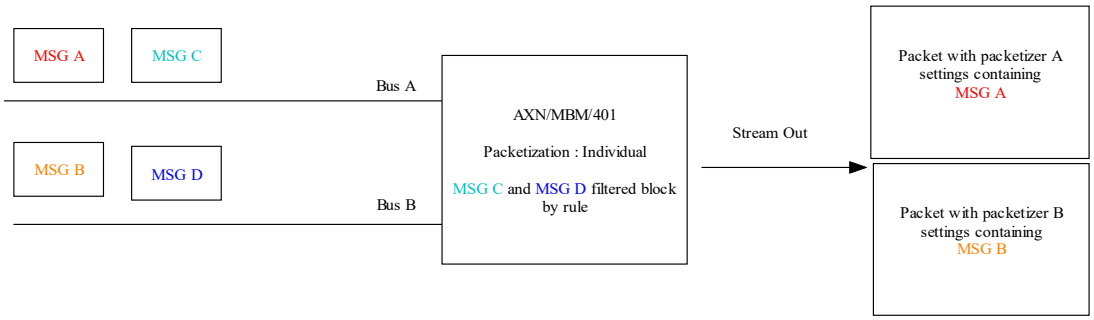


Figure 84-17: Example of Individual packetization

Dual Redundant means that all traffic from bus A and bus B are packetized in a single stream using the packetizer settings from bus MIL-STD-1553-In-A.

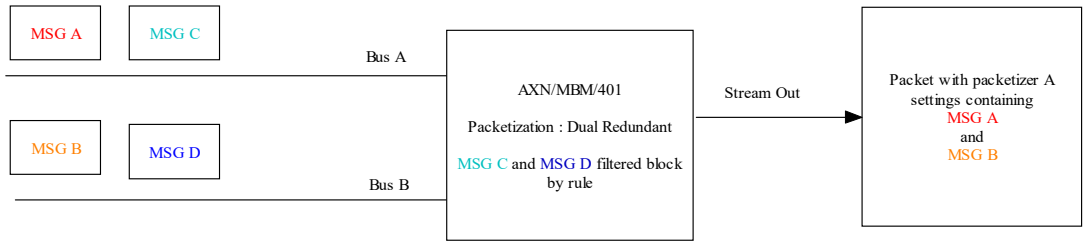


Figure 84-18: Example of Dual Redundant packetization

Dual Stream means that two streams are created with each stream containing all traffic from bus A and bus B.

For **Dual Stream** mode the primary stream uses **Packetizer** settings from bus MIL-STD-1553-In-A while the secondary stream uses **Packetizer** settings from bus MIL-STD-1553-In-B. The secondary stream gathers all data on the bus and ignores any packetizer filtering rules.

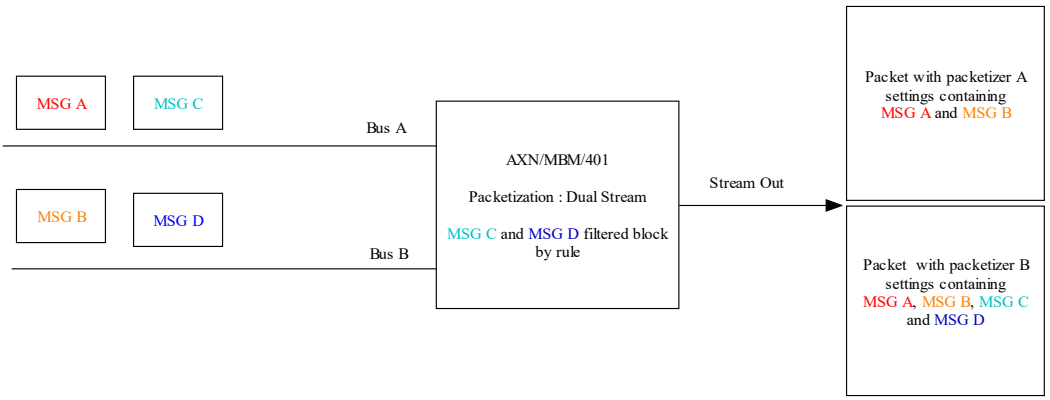


Figure 84-19: Example of Dual Stream packetization

84.6.2 Settings when Packetizer Format is iNET-X

The **Packetizer** settings are divided in two tables: bus Primary and bus Secondary. When **Packetizer Format** is set to **iNET-X**, the following settings are available.

Packetizer

Source Name	Stream Id	Channel Id	UDP Transfer Header Format	Source Id	Packetization Enabled	Packet Timeout	Max Packet Payload Size	Packetizer Filter Mode	Packetization Sink
MIL-STD-1553-In-A(0)	FFFFFFFF	FFFF	N/A	FF	☐	50	511	Pass All	All
MIL-STD-1553-In-A(1)	FFFFFFFF	FFFF	N/A	FF	☐	50	511	Pass All	All
MIL-STD-1553-In-A(2)	FFFFFFFF	FFFF	N/A	FF	☐	50	511	Pass All	All
MIL-STD-1553-In-A(3)	FFFFFFFF	FFFF	N/A	FF	☐	50	511	Pass All	All

Source Name	Stream Id	Channel Id	UDP Transfer Header Format	Source Id	Packetization Enabled	Max Packet Payload Size	Packet Timeout	Packetization Sink
MIL-STD-1553-In-B(0)	FFFFFFFF	FFFF	N/A	FF	☐	511	50	All
MIL-STD-1553-In-B(1)	FFFFFFFF	FFFF	N/A	FF	☐	511	50	All
MIL-STD-1553-In-B(2)	FFFFFFFF	FFFF	N/A	FF	☐	511	50	All
MIL-STD-1553-In-B(3)	FFFFFFFF	FFFF	N/A	FF	☐	511	50	All

Figure 84-20: Packetizer settings (iNET-X)

Setting	Description
Source Name	Channel to packetize.
Stream ID	iNET-X stream identifier for selected channel if a packet is generated via the assertion of Packetization Enabled. This is a conditional setting and is only active when Packetizer Format is set to iNET-X.
Packetization Enabled	Enables the generation of a stream of packets containing messages from this channel. DAS Studio 3 automatically creates a packetizer packet after verification/programming.
Packet Timeout	The timeout in milliseconds before a packet is generated if insufficient messages have been received to reach the Packet Size. Packets generated due to Packet Timeout are tagged in the iNET-X header. The Packet Timeout ranges from 10 ms to 999 ms (default value is 50 ms). Reducing this value results in more frequent and generally smaller packets. Increasing the value results in less frequent, but generally bigger packets.
Max Packet Payload Size	The number of words in the packet buffer, ranges from 200 words to 511 words. The default value is 511 words; reducing this value results in smaller and therefore generally more frequent packets.

Setting	Description
Packetization Sink	Selects which modules the packetizer package is sent to for transmission or storage. The choices are **Controller Only**, **All** slots or **Slot #** in which a sink module that supports packetizer logging.

For further information regarding iNET-X Placed packets used by the packetizer, refer to *TEC/NOT/067 - IENA and iNET-X packet payload formats*. Additionally, the AXN/MBM/401 data sheet provides an example of packetizer parser blocks.

84.6.3 Settings when Packetizer Format is Chapter 10

The **Packetizer** settings are divided in two tables: bus Primary and bus Secondary. When **Packetizer Format** is set to **Chapter 10**, the following settings are available.

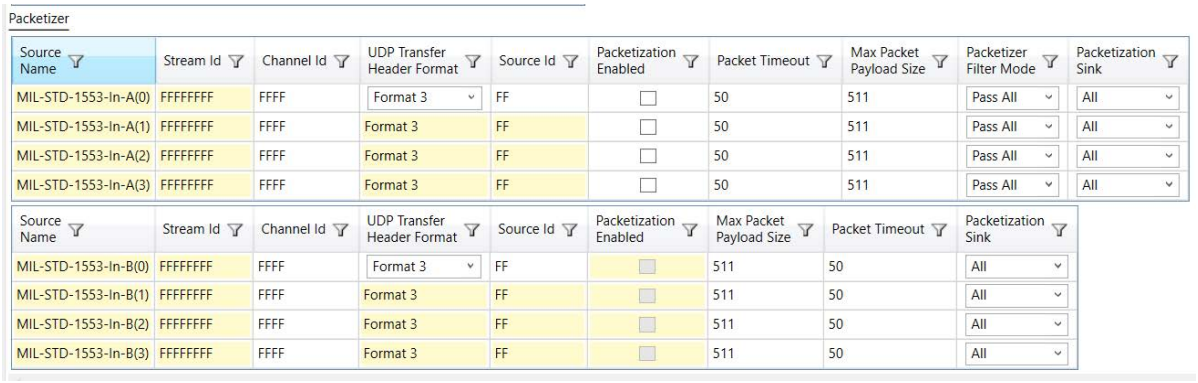


Figure 84-21: Packetizer settings (Chapter 10)

Setting	Description
Source Name	Channel to packetize.
Channel ID	Chapter 10 channel ID for selected channel if a packet is generated via the assertion of Packetization Enabled. This is a conditional setting and is only active when Packetizer Format is set to Chapter 10.
UDP Transfer Header Format	UDP transfer header format used to wrap Chapter 10 packets for streaming.
Source Id	Source Id used when streaming Chapter 10 packets in UDP transfer header format 3.
Packetization Enabled	Enables the generation of a stream of packets containing messages from this channel. DAS Studio 3 automatically creates a packetizer packet after verification/programming.
Packet Timeout	The timeout in milliseconds before a packet is generated if insufficient messages have been received to reach the Packet Size. The Packet Timeout ranges from 10 ms to 999 ms (default value is 50 ms). Reducing this value results in more frequent and generally smaller packets. Increasing the value results in less frequent, but generally bigger packets.
Max Packet Payload Size	The number of words in the packet buffer, ranges from 200 words to 511 words. The default value is 511 words; reducing this value results in smaller and therefore generally more frequent packets.
Packetization Filter Mode	Specifies the filtering mode for the channel (only applies to the primary bus). **Block By Rule** means that messages that match defined filter rules are blocked and all other messages are passed through. **Pass By Rule** means that messages that match defined filter rules are passed through and all other messages are blocked. **Pass All** means that all messages are passed through. This is only applicable when **Packetization Enabled** is selected.

Setting	Description
Packetization Sink	Selects which modules the packetizer package is sent to for transmission or storage. The choices are **Controller Only**, **All** slots or **Slot #** in which a sink module that supports packetizer logging.

Refer to the *AXN/MBM/401* data sheet for clarification of the Chapter 10 packetizer format generated by the module.

NOTE: IADS does not currently support this Chapter 10 format.

84.7 Enabling packetizer

To turn on packetizer operation on any channel, define a unique stream ID for that channel and then select the **Packetization Enabled** check box for that channel as shown in the following figure. The packetizer is enabled the next time the module is programmed.

Packetizer

Source Name	Stream Id	Channel Id	UDP Transfer Header Format	Source Id	Packetization Enabled	Packet Timeout	Max Packet Payload Size	Packetizer Filter Mode	Packetization Sink
MIL-STD-1553-In-A(0)	1	FFFF	N/A	FF	☑	50	511	Pass All	All
MIL-STD-1553-In-A(1)	2	FFFF	N/A	FF	☑	50	511	Pass All	All
MIL-STD-1553-In-A(2)	3	FFFF	N/A	FF	☑	50	511	Pass All	All
MIL-STD-1553-In-A(3)	4	FFFF	N/A	FF	☑	50	511	Pass All	All

Source Name	Stream Id	Channel Id	UDP Transfer Header Format	Source Id	Packetization Enabled	Max Packet Payload Size	Packet Timeout	Packetization Sink
MIL-STD-1553-In-B(0)	FFFFFFFF	FFFF	N/A	FF	☐	511	50	All
MIL-STD-1553-In-B(1)	FFFFFFFF	FFFF	N/A	FF	☐	511	50	All
MIL-STD-1553-In-B(2)	FFFFFFFF	FFFF	N/A	FF	☐	511	50	All
MIL-STD-1553-In-B(3)	FFFFFFFF	FFFF	N/A	FF	☐	511	50	All

Figure 84-22: Packetization Enabled setting when packetization is dual redundant

NOTE: DAS Studio automatically creates packetizer packets on the aperiodic transmitter (such as the AXN/BCU/402). The packet rate is always 1 Hz; this value is only required for XidML. This value is not used by the Axon hardware.

Settings | Processes | Packages | Algorithms | Documentation

Channels

Instrument Name	Channel Name	Bit Rate	Connection Name	Connected Instrument	Connected Channel	Package Count
MyAXN_BCU_402_C	Ethernet(1)	n/a	MyAXN_CHS_06U_Packetizing			4
MyAXN_BCU_402_C	Ethernet(2)	n/a	MyAXN_CHS_06U_1_Packetizing			4

Package Properties

Name	Rate (Hz)	Type	Sub Type	Stream ID	Source IPA	Source UDP Port	Destination MAC	Destination IPA	Destination UDP Port	DataType
AXNMBM401_1	1	INet-X	Parser aligned	1	192.168.28.1	1023	01-00-5E-00-01-64	235.0.1.100	8010	MIL-STD-1553
AXNMBM401_2	1	INet-X	Parser aligned	2	192.168.28.1	1023	01-00-5E-00-01-65	235.0.1.101	8010	MIL-STD-1553
AXNMBM401_3	1	INet-X	Parser aligned	3	192.168.28.1	1023	01-00-5E-00-01-66	235.0.1.102	8010	MIL-STD-1553
AXNMBM401_4	1	INet-X	Parser aligned	4	192.168.28.1	1023	01-00-5E-00-01-67	235.0.1.103	8010	MIL-STD-1553

Figure 84-23: AXN/BCU/402 packages showing a packetizer created by DAS Studio after verification/programming

84.8 Packetizer packet format (parser aligned)

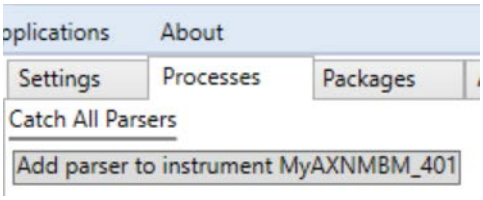
Refer to the *AXN/MBM/40x* data sheet for details on the iNET-X parser-aligned packet structure for MIL-STD-1553 bus monitoring.

84.9 Appendix

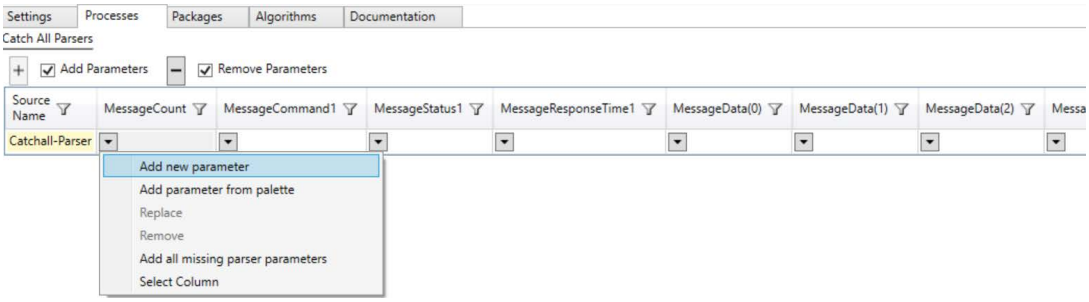
84.9.1 Catch All Parsers setting

The module supports catch all; that is, all messages except for the ones defined in the parser of the card go to this parser ID; similar function to a snarfer (implementation of IRIG-106 Chapter 8). Using catch all effectively requires sampling this register at the maximum frequency of the incoming messages, which is an inefficient approach as it requires a very high sampling rate. Catch all can be used to check if the bus is getting data other than the fill value.

To enable **Catch All Parsers**, go to the **Processes** tab of the AXN/MBM/401 and then click **Add parser to instrument MyAXNxxx**.



The following fields become available to allow parameters to be added. For further information, refer to “Chapter 15 - Processes tab” in the *DAS Studio 3 User Manual*.



84.9.2 Messages burst

The AXN/MBM/401 parser is triple buffered. If there is a burst of more than three messages between parser sampling intervals, the triple buffer can be saturated. Therefore, if it is necessary that all messages get captured, the parser data must be sampled at the speed of the burst if this is above the aggregate rate.

A message burst could also cause the stale and skip bits to be set on the MessageInfo parameter.

84.9.3 Recommended reading

To better understand this paper, read the following documents.

Table 84-1: Data sheets

Document	Description
AXN/MBM/401	MIL-STD-1553 bus monitor parser/packetizer - 4ch dual redundant
AXN/MBM/402	MIL-STD-1553 bus monitor parser/packetizer - 2ch dual redundant

Table 84-2: Data sheets

Document	Description
TEC/NOT/004	MIL-STD-1553
TEC/NOT/063	Grounding and shielding of the Axon and Acra KAM-500
TEC/NOT/067	IENA and iNET-X packet payload formats

Table 84-3: User manual

Document	Description
DOC/MAN/030	DAS Studio 3 User Manual

MIL-STD-1553 Designer's Guide

ILC Data Device Corporation®

105 Wilbur Place

Bohemia

New York 11715-2482

Military standards:

MIL-STD-1553 A

MIL-STD-1553 B

MIL-STD-1553 Multiplex Applications Handbook