

# Using DAS Studio 3 to configure the AXN/ENC/402

TEC/NOT/089

**CURTISS -  
WRIGHT**

This paper discusses the following topics:

- “59.1 Module overview” on page 1
- “59.2 Ethernet-to-PCM bridge overview” on page 2
- “59.3 AXN/ENC/402 DAS Studio 3 settings” on page 3
- “59.4 Using DAS Studio 3 to set up configuration scenarios” on page 4
- “59.5 Scenario 1: using the AXN/ENC/402 to act as an Ethernet to PCM bridge” on page 4
- “59.6 Scenario 2: using the AXN/ENC/402 as a generic Ethernet parser to be transmitted over PCM” on page 6
- “59.7 Scenario 3: using the AXN/ENC/402 to transmit Chapter 7” on page 10
- “59.8 Appendix” on page 14

## 59.1 Module overview

The AXN/ENC/402 is used to encode data in an IRIG-106 Chapter 7 (2017) PCM stream. IRIG-106 Chapter 7 defines the method for transporting variable-length, well-defined data formats in a Chapter 4 pulse code modulation (PCM) stream.

The following can be encoded:

- Parameters from any Axon module in a chassis can be placed in the Ch.7 PCM stream.
- Parameters from other Axon chassis or Ethernet sources can also be placed in the Ch.7 PCM stream.
- Ethernet frames generated in the Axon chassis can be wrapped and placed in the Ch.7 PCM stream.
- Ethernet frames from other Axon chassis or Ethernet sources can also be wrapped and placed in the Ch.7 PCM stream.

The following figure shows an example of two Axon chassis connected in a daisy-chain configuration. The AXN/BCU/402/B (or later) can transmit parameters from the remote Axon chassis to the AXN/ENC/402. It can parse a specific 3rd party Ethernet packet to transmit a specific parameter from it. It can transmit selected 3rd party Ethernet packets and transmit packetizer packets from local and remote chassis over Ch.7.

---

**NOTE:** Daisy-chaining is only supported from the AXN/BCU/402/B (or later). Instances of AXN/BCU/402 in the following figure are considered as AXN/BCU/402/B (or later).

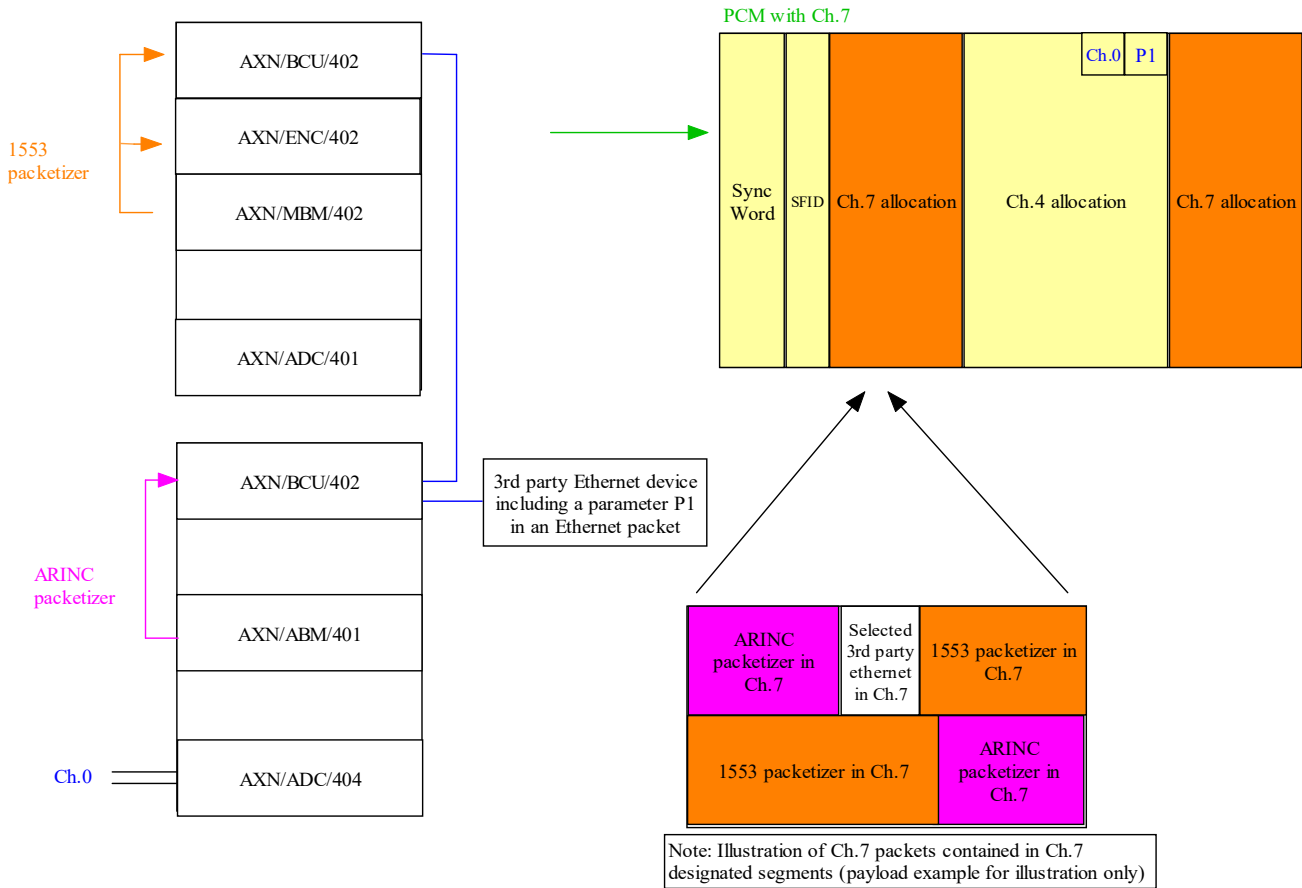


Figure 59-1: Example Ch.7 transmission

**NOTE:** The remote Axon chassis could also be an Acra KAM-500 chassis with an Ethernet controller.

The AXN/ENC/402 has multiple outputs such as TTL, RS422, and PMF with different PCM codes, however, only one PCM can be defined.

**NOTE:** The AXN/ENC/401 is similar to the AXN/ENC/402 in terms of set up. However, it can only perform IRIG-106 Chapter 4 PCM.

## 59.2 Ethernet-to-PCM bridge overview

A Multi Chassis Scheduler (MCS) is system scheduling software that enables you to transparently transmit parameters from any available Data Acquisition Unit (DAU) in a networked system by means of an Ethernet bus monitoring module located in an AXN/ENC/402 in the chassis. This system is referred to as Scenario 1 throughout this document.

The MCS software—which is integrated within the compiler of DAS Studio 3—is in charge of automatically creating intra-chassis Ethernet packets that are captured by the AXN/ENC/402.

The following diagram shows a typical MCS-over-Ethernet scenario where PCM is used for real-time telemetry in an Ethernet system.

In this example, parameter P1 comes from a module placed on a remote DAU 1 and is being transmitted over the network, then P1 gets parsed and transmitted over PCM by the AXN/ENC/402.

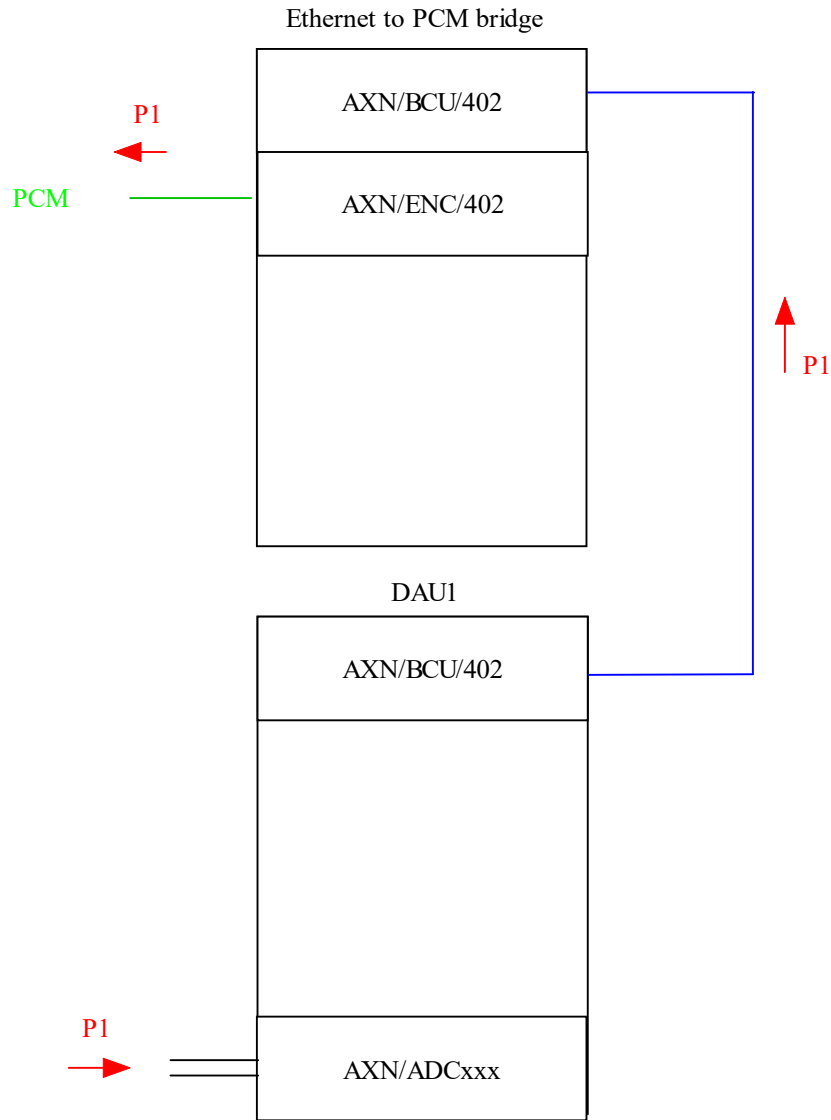


Figure 59-2: Example of MCS traffic showing parameter P1 from remote DAU 1

### 59.3 AXN/ENC/402 DAS Studio 3 settings

This technical note describes some scenarios where DAS Studio 3 can be used to configure the AXN/ENC/402. Depending on the scenario, some of the default settings shown in the following table need to be modified.

Backplane Ethernet				
Source Name	Traffic Priority	VLAN Support	Traffic Type	
BackplaneEthernet	iNET-X	<input type="checkbox"/>	iNET-X	

Figure 59-3: DAS Studio 3 Settings tab for AXN/ENC/402

Table 59-1: AXN/ENC/402 settings in DAS Studio 3

Setting	Notes
Traffic Type	<p>This setting defines the type of packet expected when configuring the AXN/ENC/402 as a bus monitor in different scenarios. In an Ethernet-to-PCM stream scenario (MCS), select <b>iNET-X</b> or <b>IENA</b>. (Scenario 1)                      In a generic Ethernet bus monitor scenario, select <b>Generic</b>. (Scenario 2 and 3)                      Select <b>All</b> when parsing both MCS packets and generic Ethernet packets.</p> <hr/> <p><b>NOTE:</b> Select either iNET-X or IENA. Do not mix traffic types; if you decide to use iNET-X packets for all other traffic, then the transport packets must also be iNET-X.</p>
Traffic Priority	<p>This advanced option is only available when parsing both MCS packets and generic Ethernet packets (Scenario 3). It defines the processing priority for package types when filtering rules overlap.</p>
VLAN Support	<p>This is an advanced option. Enabling VLAN Support allows parser slots—when present—to skip over VLAN packet headers, allowing you to classify traffic on the basis of IP and UDP/TCP packet header fields. By default, this check box is cleared.</p>

For further information, refer to the latest *AXN/ENC/402* data sheet.

## 59.4 Using DAS Studio 3 to set up configuration scenarios

DAS Studio 3 is used to create a configuration, which contains the various elements, which 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, see Figure 1 in the *DAS Studio 3 User Manual*.

## 59.5 Scenario 1: using the AXN/ENC/402 to act as an Ethernet to PCM bridge

As described in “59.2 Ethernet-to-PCM bridge overview” on page 2, in Ethernet systems where every DAU has an Ethernet controller and transmits data to a central switch, the AXN/ENC/402 can act as a bridge module for transmitting Ethernet-captured data into a Pulse Code Modulation (PCM) stream.

In this scenario, the AXN/ENC/402 is connected to the output of an aggregating switch, through which data from all other DAUs is available to the PCM DAU.

This scenario is illustrated in the following figure where 3 DAUs (DAU\_0, DAU\_1, and PCM\_DAU) are connected to a network switch; connecting the system in this way makes all DAU parameters available to the PCM stream transmitted by the AXN/ENC/402.

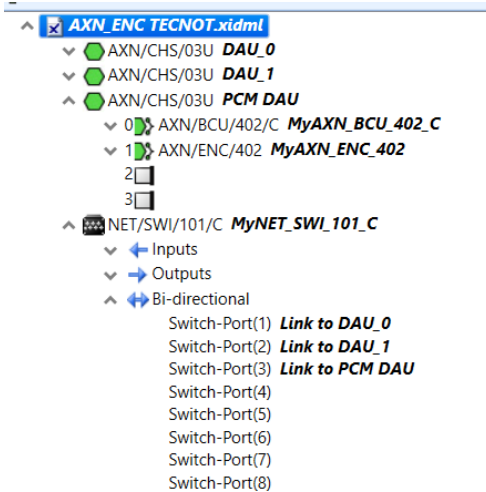


Figure 59-4: Scenario 1 – Ethernet to PCM bridge (MCS)

**NOTE:** This example assumes the network switch is correctly configured to route packets from a remote DAU to the AXN/ENC/402.

In the following figure, Traffic Type has to be configured to IENA or iNET-X to act as a bridge module for transmitting Ethernet data and Traffic Type is set to iNET-X or IENA.

Backplane Ethernet			
Source Name	Traffic Priority	VLAN Support	Traffic Type
BackplaneEthernet	iNET-X	<input type="checkbox"/>	iNET-X

Figure 59-5: DAS Studio 3 settings when configuring the AXN/ENC/402 for MCS

The following figure shows the read only fields, which are automatically set showing what Ethernet field the module uses to parse the MCS packet.

iNET-X Parsing		
Source Name	Source IP Address (iNet-X)	Stream ID (iNet-X)
BackplaneEthernet	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 59-6: DAS Studio 3 read only settings when configuring the AXN/ENC/402 for MCS

You can proceed to build a PCM frame once the routing of the switch (for example NET/SWI/101/C) and the module settings are configured correctly in DAS Studio 3. Once the PCM is created, click Verify; MCS packets are automatically built by DAS Studio 3 at the data source DAUs. This ensures that the data words to be sent over real-time PCM arrive at the data selector chassis in time for transmission.

Refer to the *DAS Studio 3 User Manual* to build an IRIG-106 PCM using Transmission Assistant.

## 59.6 Scenario 2: using the AXN/ENC/402 as a generic Ethernet parser to be transmitted over PCM

The AXN/ENC/402 can also be used as an Ethernet bus monitor to parse traffic from external sources, thereby making it available to the AXN/ENC/402 PCM. In this scenario, all traffic to be parsed by the AXN/ENC/402 is considered generic, therefore you must use the Ethernet Builder application to parse Ethernet traffic.

Under **Backplane Routing**, set both **Internal Ethernet Placed Packets** and **External Ethernet Packets** fields on the AXN/BCU/402/B (or later) to indicate where the generic packets are destined.

Backplane Routing	Slot 2
Internal Ethernet Placed Packets	●
External Ethernet Packets	●

Figure 59-7: Example of the AXN/BCU/40x routing to the AXN/ENC/402 in Slot 2

Selecting **External Ethernet Packets** (to Backplane) routes all the external packets going through the internal switch of the AXN/BCU/402/x to Slot 2 of the AXN/ENC/402.

Selecting **Internal Ethernet Placed Packets** (to Backplane) routes all the packets created in the AXN/BCU/402/x to Slot 2 of the AXN/ENC/402.

For the AXN/ENC/402 to parse generic Ethernet traffic, you must set the Traffic Type field to Generic as shown in the following figure.

Backplane Ethernet			
Source Name	Traffic Priority	VLAN Support	Traffic Type
BackplaneEthernet	Generic	<input type="checkbox"/>	Generic

Figure 59-8: Scenario 2 - AXN/ENC/402 acting as a generic Ethernet parser in DAS Studio 3

The **Ethernet Builder** application adds and configures Ethernet messages and parameters on Ethernet bus monitor modules.

This section features a worked example of how to use Ethernet Builder to configure the AXN/ENC/40x to parse an Ethernet packet with a source IP of 10.11.12.13, a destination IP of 233.10.11.45, and a destination port of 1024.

In this worked example, words 100, 102, 104, and 106 are extracted from the parsed packet.

Knowing that the Word Offset is defined as per the example in the following table, Word Offset 0 is the start of the Ethernet Frame and is therefore the first 16 bits of the Destination MAC Address.

Table 59-2: iNET-X, IENA, and VLAN word offset

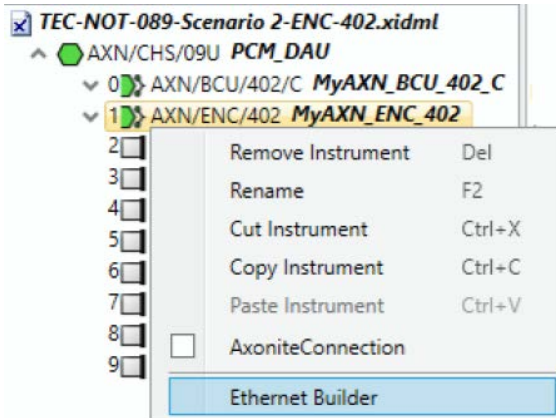
Field name	Values of offset index words	
	Offset If VLAN disabled	Offset If VLAN enabled
<b>MAC header</b>		
Destination MAC address, word 0 (MSW)	0	0
Destination MAC address, word 1	1	1
Destination MAC address, word 2 (LSW)	2	2
Source MAC address, word 0 (MSW)	3	3
Source MAC address, word 1	4	4
Source MAC address, word 2 (LSW)	5	5

Table 59-2: iNET-X, IENA, and VLAN word offset (continued)

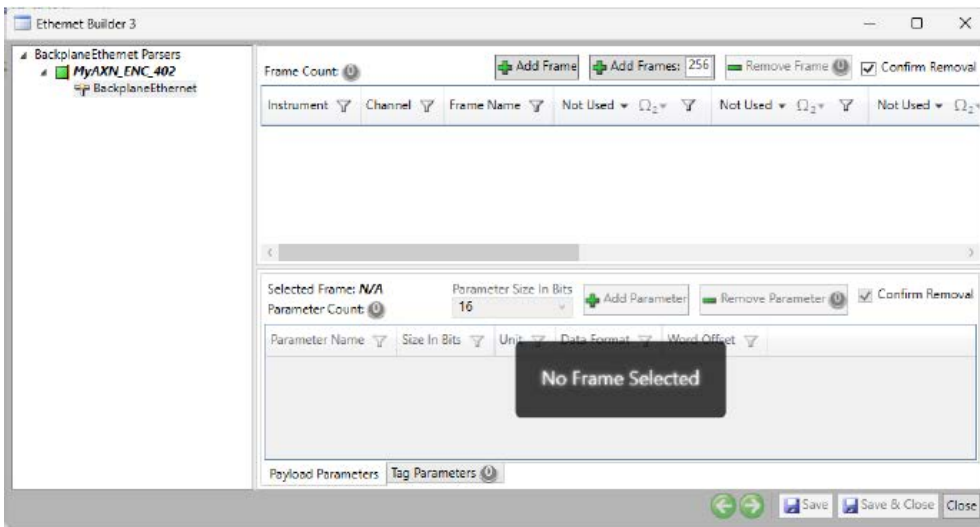
Field name	Values of offset index words	
	Offset If VLAN disabled	Offset If VLAN enabled
Frame/Protocol type; 0x0*0 e.g. IP = 0800h	6	8
<b>IP header</b>		
VLAN priority/ID	N/A	7
IP version/IHL/ToS	7	9
IP packet size	8	10
IP ID	9	11
IP flags and fragment offset	10	12
IP TTL and Protocol; xx11h for UDP and xx06h for TCP	11	13
IP header checksum	12	14
Source IP address, word 0 (MSW)	13	15
Source IP address, word 1 (LSW)	14	16
Destination IP address, word 0 (MSW)	15	17
Destination IP address, word 1 (LSW)	16	18
Source port no.	17	19
Destination port no.	18	20
UDP Length; TCP seq word 0	19	21
UDP CSum; TCP seq word 1	20	22
<b>Data payload</b>		
IENA key or iNET Control Field 0 // TCP Ack word 0	21	23
IENA size or iNET Control Field 1 // TCP Ack word 1	22	24
iNET-X Stream ID word 0 or IENA date 0 // TCP offset/flags	23	25
iNET-X Stream ID word 1 or IENA date 1// TCP window	24	26
iNET-X Seq Number 0 or IENA date 2 // TCP Csum	25	27
iNET-X Seq Number 1 or IENA status // TCP urgent Ptr	26	28
iNET-X Pckt Length 0 or IENA Seq // TCP Options (or TCP Data 0)	27	29

In the following example, an AXN/ENC/402 is used. The AXN/ENC/401 follows the same concept as the AXN/ENC/402, as the IRIG-106 Chapter 4 PCM is the same on both encoders.

1. Right-click the AXN/ENC/402 module in the Navigator and then click **Ethernet Builder**. (You can also click the **Applications** menu and then click **Ethernet Builder**.)

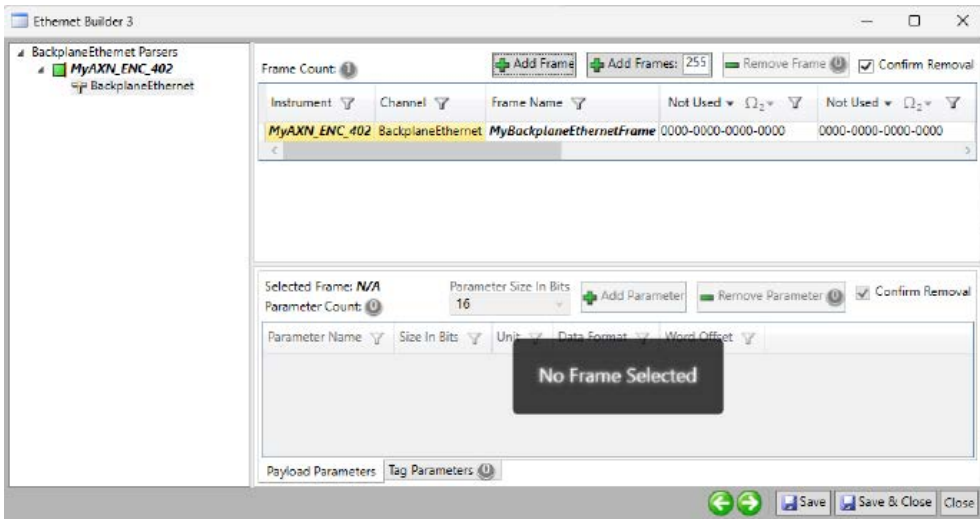


The **Ethernet Builder 3** dialog box opens.



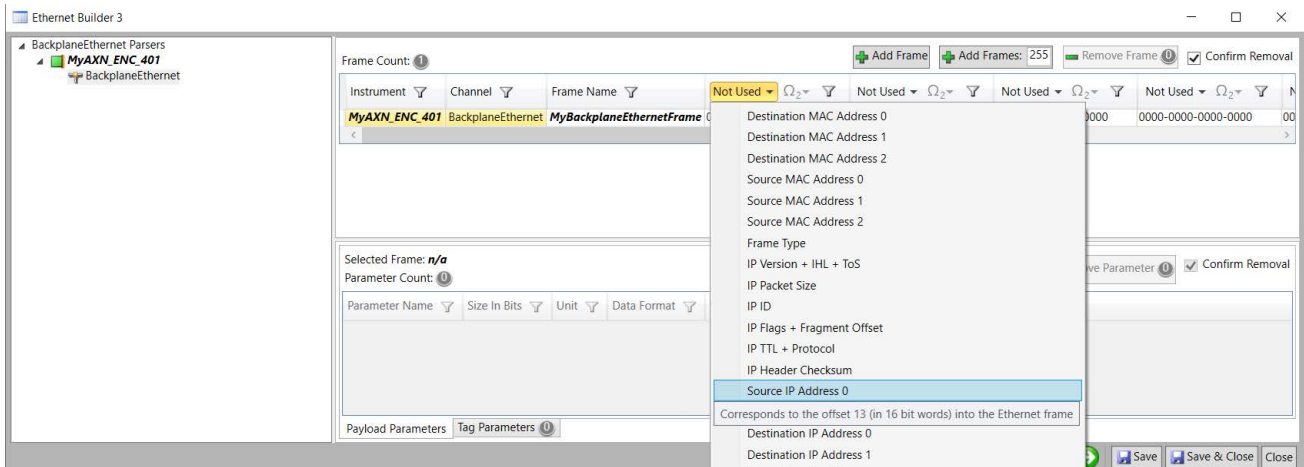
For dialog box navigation, see “Builder application GUI overview” in the *DAS Studio 3 User Manual*.

2. Add a single frame to the Ethernet link by selecting Ethernet in the Navigator pane and clicking **Add Frame**.

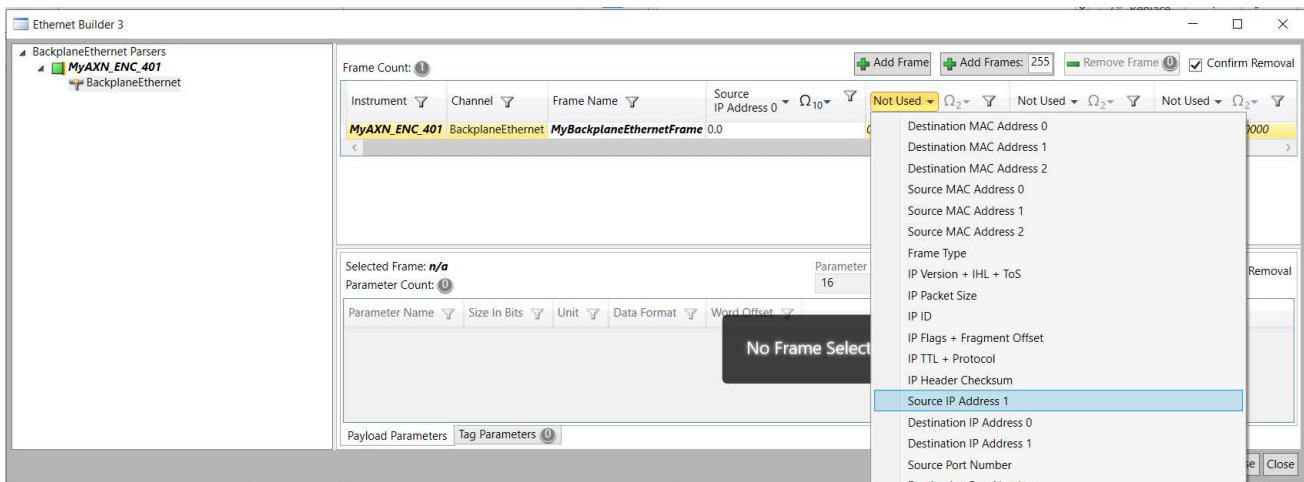


**NOTE:** In the **Frame Count** pane, there are eight 16-bit user-definable classifier fields (by default, these fields are labeled **Not Used**), which can be used to create the rules which help the AXN/ENC/40x identify Ethernet packets. As the fields to be used for source IP and destination IP are both two 16-bit fields, two classifier rules for both the source IP and the destination IP are required.

- In the first classifier field, click the drop-down arrow beside **Not Used** and select **Source IP Address 0**.

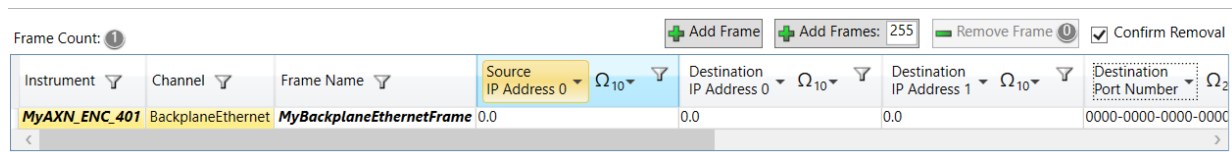


- In the second classifier field, click the drop-down arrow beside **Not Used** and select **Source IP Address 1**.



**NOTE:** When Source IP Address 0 is in use, it is not available in the drop-down list.

- Set the remaining classifier fields by selecting **Destination IP** for the third and fourth classifier fields, and selecting **Destination Port Number** for the fifth classifier field. Definitions for the required classifier fields are now set.



**NOTE:** Values for these classifier fields can be formatted using Decimal, Hex, Octal, or Binary format. To change a format, click the drop-down arrow by the  $\Omega$  symbol for each classifier field and choose the desired format. In our working example, Decimal format suffices.

- In the **Source IP Address 0** field, type **10.11**.
- In the **Source IP Address 1** field, type **12.13**.
- In the **Destination IP Address 0** field, type **233.10**.
- In the **Destination IP Address 1** field, type **11.45**.

- In the **Destination Port** field, type **1024**.  
Values for the classifiers fields are now set.

Channel	Frame Name	Source IP Address 0	Source IP Address 1	Destination IP Address 0	Destination IP Address 1	Destination Port Number
Ethernet	MyEthernetFrame	10.11	12.13	233.10	11.45	1024

**NOTE:** Now that parsing classifiers have been defined, add the parameters to be extracted from the packet; specifically, words 100, 102, 104, and 106.

- In the **Start Offset Word** field, type **100**.
- In the **Word Offset Increment** field, type **2**.
- In the field beside **Add Parameters**, type **4**.

Selected Frame: **MyEthernetFrame**  
 Parameter Count: 0  
 Start Offset Word: 100  
 Word Offset Increment: 2  
 Add Parameter  
 Add Parameters: 4

Parameter Name	Size In Bits	Unit	Data Format	Word Offset

- Click **Add Parameters**.  
Four parameters are now created at offsets **100, 102, 104, and 106**.

Selected Frame: **MyEthernetFrame**  
 Parameter Count: 4  
 Start Offset Word: 100  
 Word Offset Increment: 2  
 Add Parameter  
 Add Parameters: 329

Parameter Name	Size In Bits	Unit	Data Format	Word Offset
MyEthernetFrame.MyParameter	16	BitVector	BitVector	100
MyEthernetFrame.MyParameter1	16	BitVector	BitVector	102
MyEthernetFrame.MyParameter2	16	BitVector	BitVector	104
MyEthernetFrame.MyParameter3	16	BitVector	BitVector	106

- To save changes, click **Save & Close**.

For further information on Ethernet Builder, refer to the *DAS Studio 3 User Manual*.

MyEthernetFrame.MyParameter created are now available to be placed in the PCM. Refer to the DAS Studio manual to build an IRIG-106 PCM using Transmission Assistant.

**NOTE:** The parameters parsed by the AXN/ENC/402 are not available to any other modules in the system.

The AXN/ENC/402 can simultaneously operate as Generic and Ethernet-to-PCM bridge by setting **Traffic Type** to **All** as shown in the following figure.

Source Name	Traffic Priority	VLAN Support	Traffic Type
BackplaneEthernet	Generic then IENA then iNET-X	<input type="checkbox"/>	All

Figure 59-9: Backplane Ethernet with Traffic Type set to All

## 59.7 Scenario 3: using the AXN/ENC/402 to transmit Chapter 7

IRIG-106 Chapter 7 defines the method for transporting variable-length, well-defined data formats in a Chapter 4 pulse code modulation (PCM) stream.

Any Ethernet frames (even generated in the Axon chassis such as packetizer or any remote chassis) can be wrapped and placed in the Ch.7 PCM stream as shown in Figure 59-1 on page 2. AXN/ENC/402 only supports PT Raw Ethernet Media Access Control (MAC) Frame Packet (type 0x4) type packets.

Both high latency and low latency packet definitions are supported.

### 59.7.1 Transmitting generic packets over Ch.7

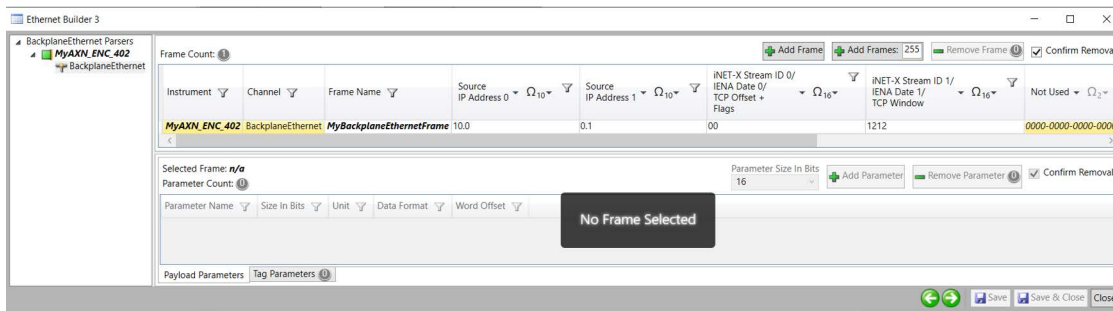
If the packet is coming internally, set the field **Internal Ethernet Placed Packets** on the AXN/BCU/40x to indicate where the generic packets are destined.

Backplane Routing	Slot 2
Internal Ethernet Placed Packets	<input checked="" type="checkbox"/>
External Ethernet Packets	<input checked="" type="checkbox"/>

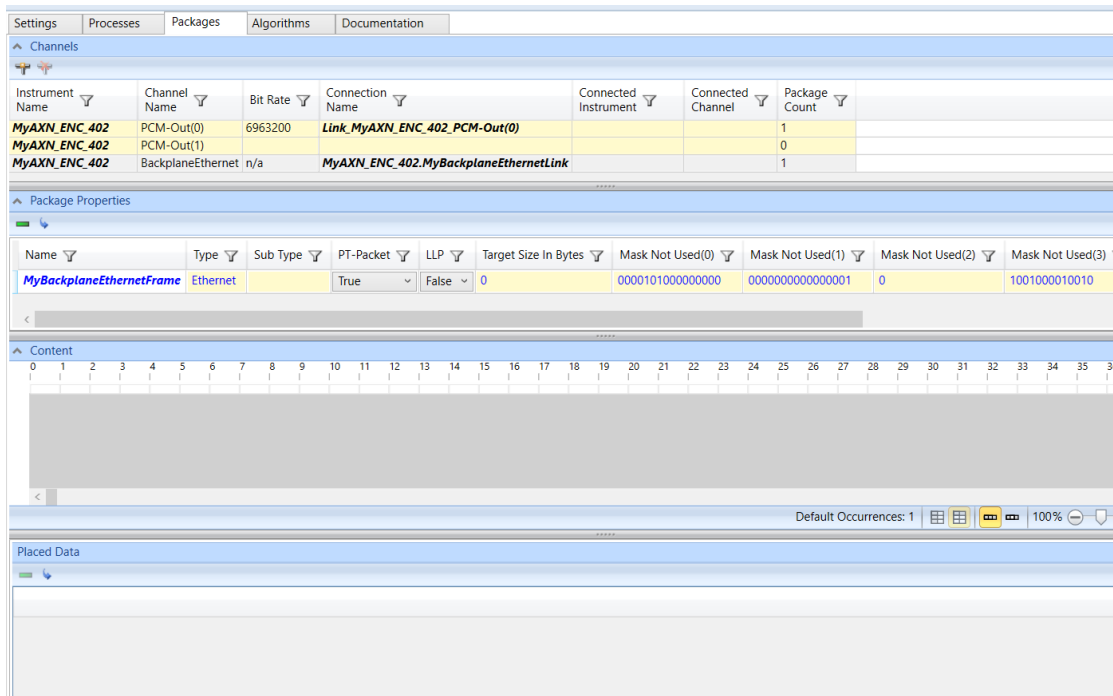
Figure 59-10: Example of AXN/BCU/40x routing to the AXN/ENC/402 in slot 2

Use **Ethernet Builder** to define the frame you want to select.

1. Right-click the AXN/ENC/402 module in the Navigator and then click **Ethernet Builder**. (You can also click the **Applications** menu and then click **Ethernet Builder**.)  
The settings here are configured to select a packet with source IP 10.0.0.1 and Stream ID 1212h. (This packet could be a packetizer from an AXN/ABM/401 for example.)



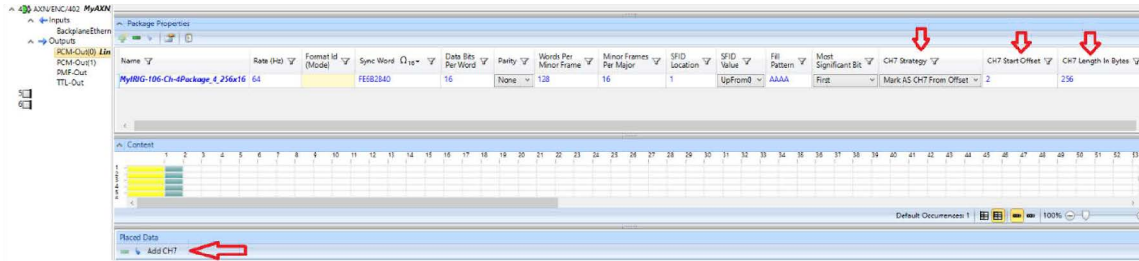
2. Click **Save & Close**.
3. On the **Packages** tab of the AXN/ENC/402, select **BackplaneEthernet** and then set **PT-Packet** to **True**.  
LLP (Low Latency Packet) is set to False by default. If set to True, a target payload size in bytes must be provided.



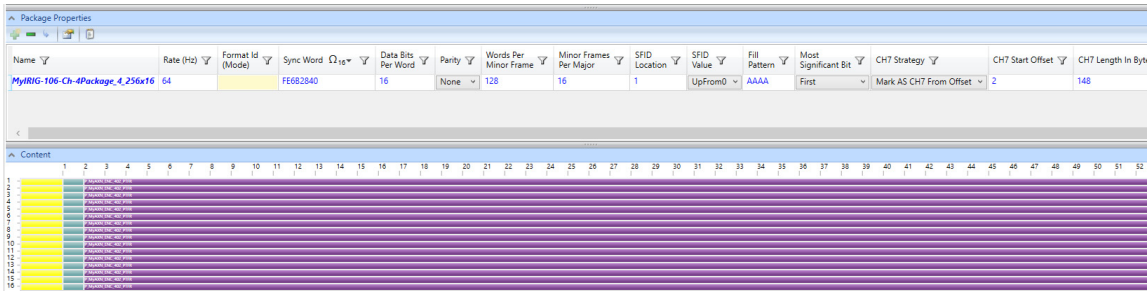
### 59.7.1.1 Allocate Ch.7 space in the frame

When marking packets as PT-Packets, an area of the PCM frame must be reserved for transmission of these packets. In the example below, a PCM frame has been created.

- Unused space must be allocated in the minor frame starting from a particular offset. The reserved space must be the same length and start at the same offset in each minor frame. For that, set the CH7 Strategy as “Mark as CH7 From Offset” as 2 and “CH7 Start Length in Bytes” as 148.



- Click the **Add CH7** button. The **Content** pane is updated.



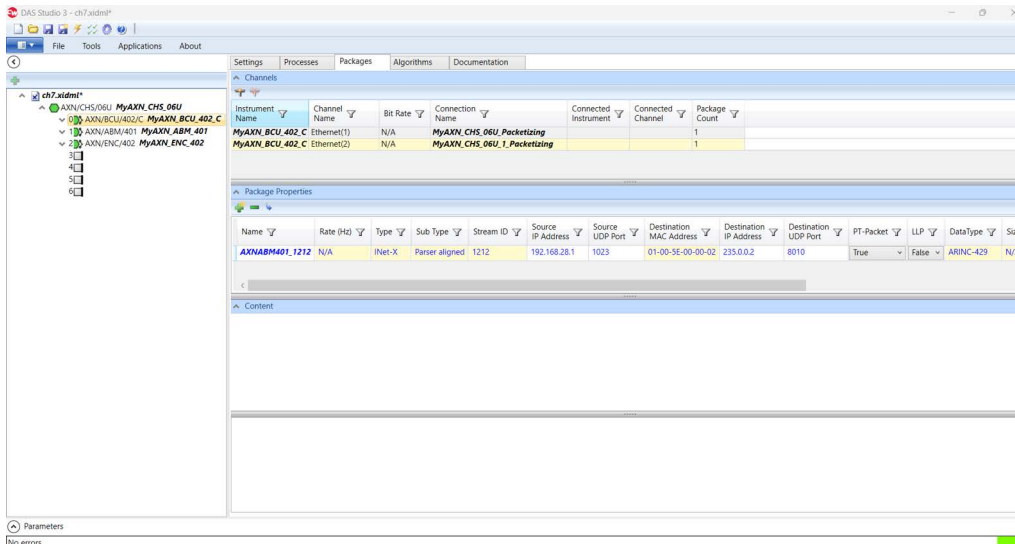
Each minor frame carries a payload of a single transport packet, which may be segmented. This means that Ch.7 data may be interleaved with non-Ch.7 data.

### 59.7.1.2 Transmitting packetizers over Ch.7

IENA and INET-X packetizers can be transmitted over Ch.7. If a packetizer packet is available in the chassis (Verify has to be run in order for the MCS to create the packet), the AXN/ENC/402 automatically makes the packet available in the AXN/BCU/402 module **Packages** tab.

The following example shows a packetizer packet over Ch.7 with stream ID 0x1212 from an AXN/ABM/401 module placed in the same chassis as the AXN/ENC/402.

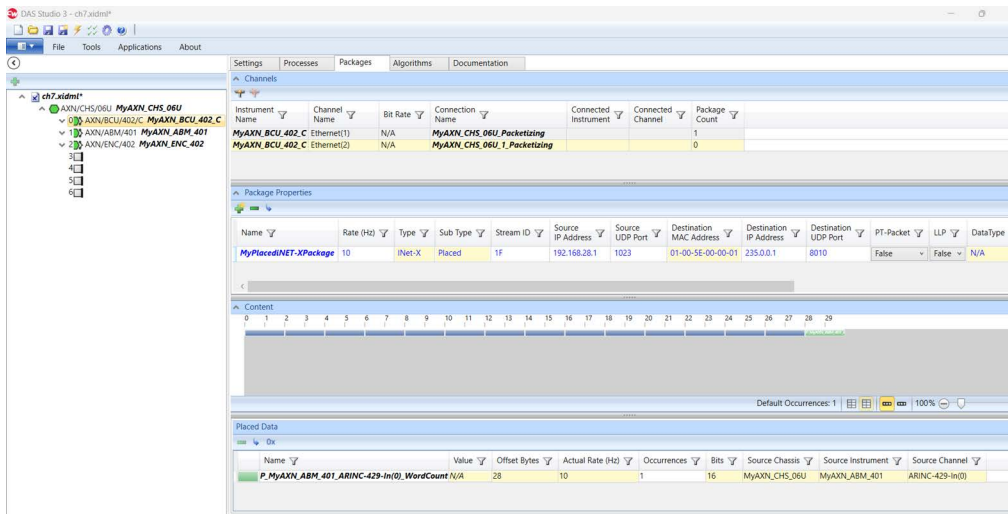
- To transmit the package over Ch.7, select the packetizer packet and then set **PT-Packet** to **True**. **LLP** (Low Latency Packet) is set to **False** by default on the AXN//BCU/402.



### 59.7.1.3 Transmitting placed packets over Ch.7

iENA and iNET-X placed packets can be transmitted over Ch.7, if a placed packet is available on the AXN/BCU/402,

- Select the placed packet and then set **PT-Packet** to **True**.



- To transmit it over Ch.7, select the packetizer and set **PT-Packet** to **True**. **LLP** (Low Latency Packet) is set to **False** by default.

For details on building Ch.7 streams, refer to the *DAS Studio 3 User Manual*. For additional information, refer to the IRIG-106 Chapter 7 specification and the AXN/ENC/402 data sheet.

**NOTE:** You cannot transmit an Ethernet packet over Ch.7 and parse it in Generic mode simultaneously. IADS doesn't support Chapter 7.

## 59.8 Appendix

### 59.8.1 Buffer depth to handle burst packet

The AXN/ENC/402 has the capability to set the buffer depth to allow parsing of 3rd party bursty Ethernet packets. Increasing the buffer depth, decreases the number of available parser slots.

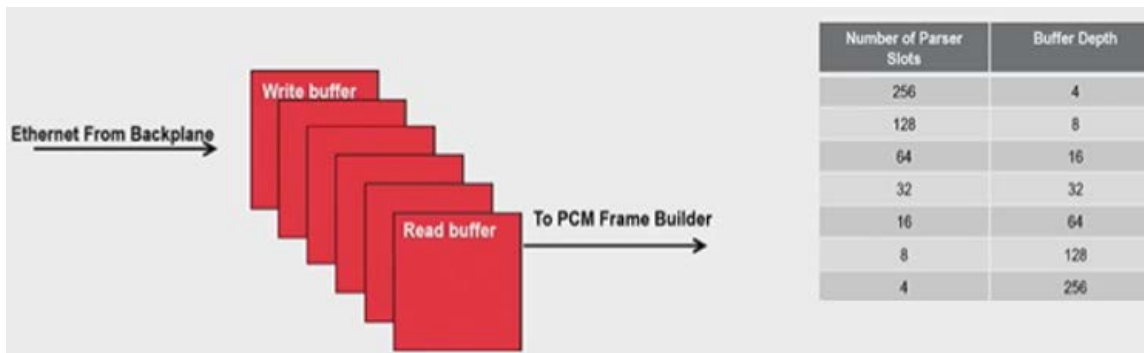


Figure 59-11: Deep buffer versus parser slot

**NOTE:** Buffer depth is not used for parsing MCS packets.

### 59.8.2 Bit-rate

The bit-rate is automatically calculated by DAS Studio according to the major frame rate and total bits. The AXN/ENC/402 is capable of generating whatever bit-rate value that results from the defined frame size and rate, as long as the rate is a reasonable float number and in the min/max limits specified in the data sheet.

### 59.8.3 Parsing non fixed length Ethernet generic packet

Because the packet length is not fixed, you will need to get the maximum data words (that is 758, which is the maximum for an Ethernet packet). You can decrease it if you know the maximum output from the device you are trying to monitor.

You must discard the last data words if the packet length is not at its maximum. The data word 8 reports the IP size packet. Therefore, you can use it to discard them after post-processing on the ground.

Field name	Values of offset index words	
	Offset If VLAN disabled	Offset If VLAN enabled
Destination MAC address, word 0 (MSW)	0	0
Destination MAC address, word 1	1	1
Destination MAC address, word 2 (LSW)	2	2
Source MAC address, word 0 (MSW)	3	3
Source MAC address, word 1	4	4
Source MAC address, word 2 (LSW)	5	5
Frame/Protocol type; 0x0*0 e.g. IP = 0800h	6	8
VLAN priority/ID	N/A	7
IP version/IHL/ToS	7	9
IP packet size	8	10

For example, if the packet is 700 data words, discard the last 58 data words because they will be rubbish values (RAM).

### 59.8.4 Parsing parameters with multiple occurrences in an Ethernet generic packet

The AXN/ENC/402 parses Ethernet packets into 16-bits words; regardless of whether parameters are composed of multiple words/occurrences.

The following figure shows an example of a parameter P1 composed of two words in the same packet, which is a unique type of packet coming from an Ethernet device source IP 10.0.0.1. The first instance 1-P1 is at word offset 30; the second instance is at word offset 33.

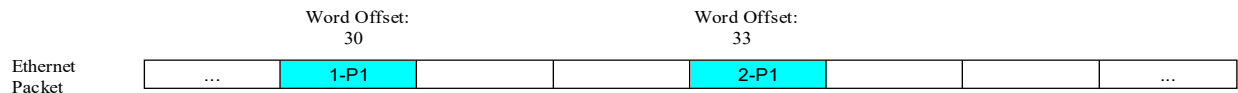


Figure 59-12: Example of a parameter P1 composed of two words in the same packet

Two distinct data words need to be set up in Ethernet Builder as shown in the following figure.

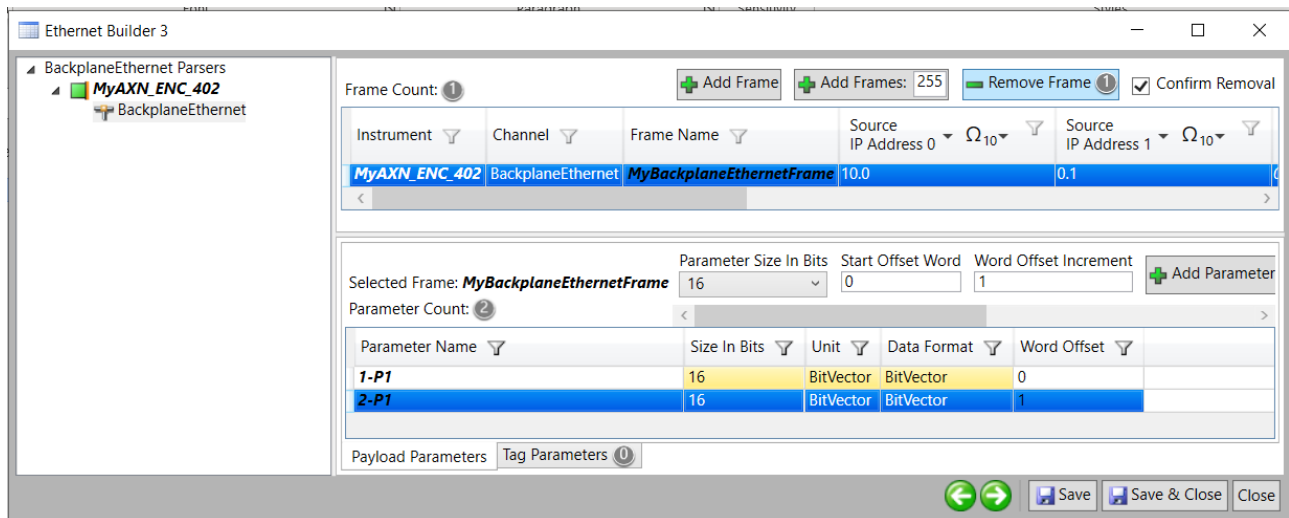


Figure 59-13: Defining distinct data words in Ethernet Builder

If the parameter needs to be placed in PCM for example, the two samples composing the parameter P1 need to be commutated accordingly to match the PCM frame. In our example below it is set at 1:2

SFID						Syncword
SFID				1-P1		Syncword
SFID						Syncword
SFID				2 -P1		Syncword

Figure 59-14: PCM showing the two samples composing the parameter P1

When decommutating the PCM using software such as IADS, a mimic PCM is required with one single parameter at 1:2. This signals the decom software that the two samples are coming from a single parameter.

SFID						Syncword
SFID				P1		Syncword
SFID						Syncword
SFID				P1		Syncword

Figure 59-15: Mimic PCM indicating to decom software that two samples belong to the same parameter P1

## 59.8.5 VLAN

### 59.8.5.1

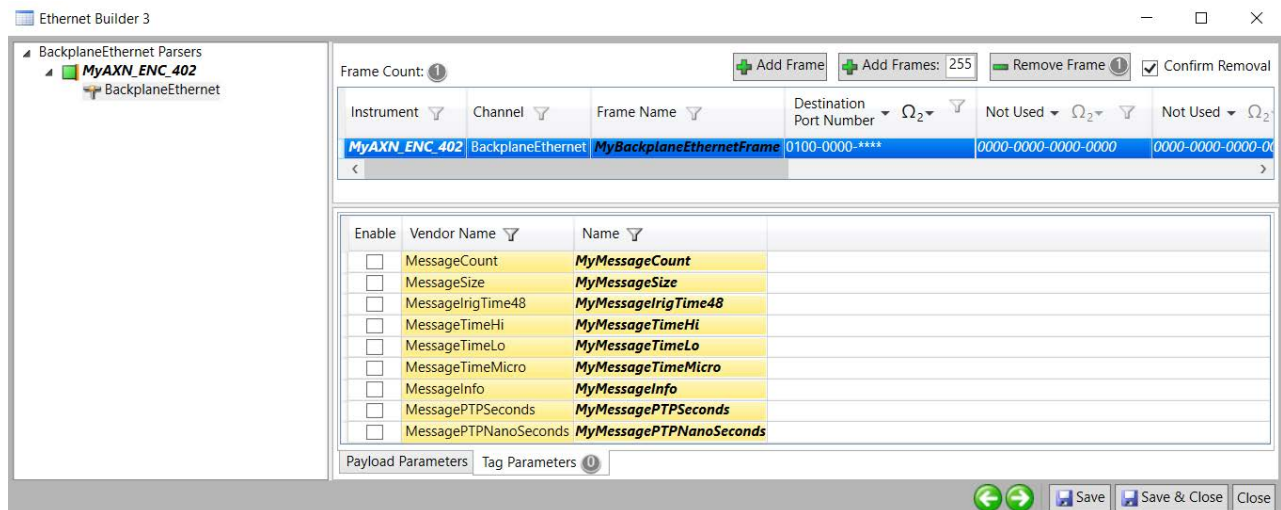
VLAN Tagging, also known as Frame Tagging, is a method developed by Cisco to help identify packets traveling through trunk links. When an Ethernet frame traverses a trunk link, a special VLAN tag is added to the frame and sent across the trunk link. A commercial switch maybe able to identify packets from different VLANs according to the information contained in its VLAN tags. IEEE 802.1Q adds a 4-byte VLAN tag between the Source/Destination MAC address and Length/Type fields of an Ethernet frame to identify the VLAN to which the frame belongs

### 59.8.6 Using wildcards and tags in Ethernet Builder

On occasion, data may be required from a packet on the same link with the same source and destination IP addresses (but whose destination port may change). This data can be captured by using wildcards in the classifier settings. Wildcards are expressed as asterisks in the Ethernet builder, and can only be used when Binary format has been set for values in the classifier fields.

To add another packet with the exact same settings as the packet above, do the following:

1. Open **Ethernet Builder**.
1. Click **Add Frame**.
2. Select the **Destination Port Number** field in the new packet, and click the drop-down arrow by the Ω symbol to change its format from **Decimal** to **Binary**.  
The 1024 value in the **Destination Port Number** field now reads **0100-0000-0000**.
3. To represent varying port numbers, insert wildcards (asterisks) where required. This forces the AXN/ENC/402 to capture this packet regardless of the destination port number.
4. For example, 0100-0000-\*\*\*\* parses incoming packets with the same source and destination IP from port 1024 to 1039.  
**NOTE:** Wildcarding is only supported in binary mode. At this point, if the Binary format is changed back to Decimal (using the drop-down arrow beside an Ω symbol), the value for the Destination Port Number field (edited using wildcards) does not change format.



5. Add parameters to the new packet as required.
6. Click the **Tag Parameters** tab to add additional message tag parameters such as **MessageTimeStamp** (timestamp of arrival of the message), **MessageCount**, **MessageSize** (bytes), **MessageStatus**, **MessageInfo** (stale / skipped message).
7. To save changes, click **Save & Close**.

### 59.8.7 Related documentation

To better understand this paper, read the following documents.

Table 59-3: Data sheets

Document	Description
AXN/ENC/401	IRIG-106 PCM encoder (PMF output)
AXN/ENC/402	IRIG-106 PCM encoder (PMF output) with Chapter 7

Table 59-4: Data sheets

Document	Description
TEC/NOT/024	Evolution of Pulse Code Modulation (PCM)
TEC/NOT/027	IRIG 106-96 Chapter 4
TEC/NOT/067	IENA and iNET-X packet payload formats

Table 59-5: User manual

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

This page is intentionally blank