Using the KAD/UAR/102

The KAD/UAR/102 parses (coherently extracts specific bytes) and snarfs (sends all data to a FIFO) up to four RS-232/422/485 channels. This paper introduces the RS-232/422/485 channels and outlines how to configure the KAD/UAR/102 module. This paper discusses the following topics:

35.1 RS-232/RS-422/RS-485 overview

This section introduces RS-232/422/485, focusing on the physical layer and the bit definition.

35.1.1 Physical layer

RS-232 is single ended, so the difference voltage is relative to ground. RS-422 and RS-485 are differential ended, so the difference voltage is between the positive and negative terminals. RS-422 and RS-485 are nominally independent of ground, but if the ground potential differs by too much between end nodes, then no data is received.

The RS-232, RS-422 and RS-485 logic 0 is less than -200 mV. Logic 1 is more than 200 mV (see the following figure).
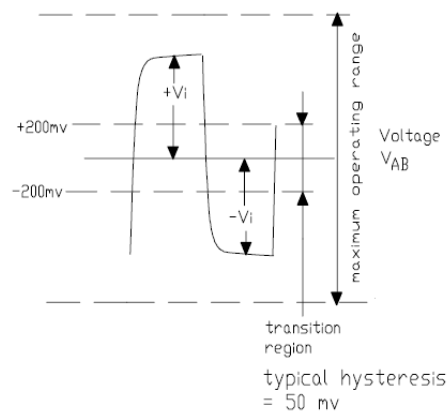

Figure 35-1: Differential input

RS-422 and RS-485 require a termination at the end of the transmission line (see the following figure).

Figure 35-2: Termination differential input

35.1.2 Bit definition

A data word can be either seven or eight bits in length. A parity bit can be placed at the end of each data word (see the following figure).
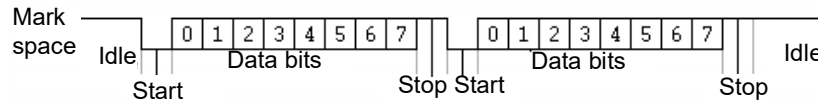

Figure 35-3: Serial bit definition

35.2 KAD/UAR/102 and the serial link

This section outlines the key features of the KAD/UAR/102 with respect to the serial link monitoring. Like other Curtiss-Wright bus monitors, the KAD/UAR/102 uses a triple buffer for parsing.

35.2.1 Parsing

The following figure illustrates the triple buffering of data words (green) and time message tags (white) used for each bus in the KAD/UAR/102's parser. These data words and message tags are described in Table 35-1 on page 2.
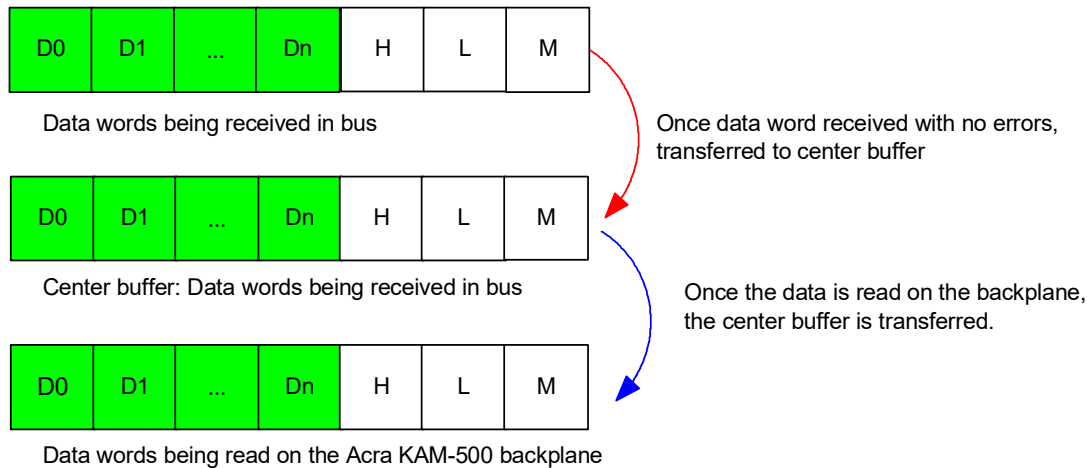

Figure 35-4: Triple buffering of traffic and associated message tags

Table 35-1: Data word and message tag descriptions

Data word/message tag	Description
D0, D1, D2, Dn	Traffic with n<512.
H	High time.
L	Low time.
M	Micro time.

Time message tags are added to each message received and stored in separate buffers for each of the four busses. As soon as a message is received with no errors, the complete 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 MsgInfo register. Refer to the *KAD/UAR/102* data sheet**.**

35.3 Overview of the KAD/UAR/102

The KAD/UAR/102 is an RS-232, RS-422, or RS-485 universal asynchronous bus monitor, a coherent message parser and an error-detection function on a single module. The KAD/UAR/102 also has snarfer capability for each of the four busses.

In the parser, a total of up to 126 complete messages are triple buffered so that the stale indication is message-wide. Each message can be up to 512 characters (bytes) long (including start and stop characters). Each message is tagged to 0.1 ms resolution; a message is considered found when up to eight specific bytes are received. The end of a message is determined by a user-defined stop character or specific number of data words. A message is not updated if any sequence is incorrect. The snarfer stores all data from each bus in a separate FIFO 2K data words deep.

35.3.1 Key features

- Four independent input channels
- Bit-rates from 300 bps to 1,000,000 bps
- 7/8 bits per data word with odd/even or no parity
- Programmable start sequence (one to eight characters)
- Programmable stop sequence (one character or by fixed length)
- Locks on idle time
- Parses up to 126 strings (messages) per module
- Up to 512 characters (bytes) per message; minimum of nine characters (bytes) per message
- Time tagging (±0.1 ms)
- Message wide stale (skipped) indication
- FIFO is 2K data words deep per channel
- Two bits for FIFO full/empty

35.4 Setting up the KAD/UAR/102 using KSM-500

You may use kSetup software to configure the KAD/UAR/102. kSetup is included in the KSM-500 suite of tools. Some configuration features that are needed to set up the KAD/UAR/102 include configuring channels, setting messages and adding data words to messages. This information is summarized in the "Setting up packages" and "Setting up datalinks" section of the *KAD/UAR/102* data sheet.

35.4.1 Setting parameters

The Parameters tab (see the following figure) displays all the parameters available from the module. To set parameters for the KAD/UAR/102, select the module in the Task Explorer pane of kSetup. On the Parameters tab, complete the fields described in the following table. For more information on using kSetup, contact Curtiss-Wright support (acra-support@curtisswright.com).

Figure 35-5: Parameters tab in kSetup

Table 35-2: Parameters tab settings

Field name	Description
Parameter Name	Name of the parameter.
Register	The Report register is useful to transmit when monitoring message problems; for details of register definitions, see the *KAD/UAR/102* data sheet.

Table 35-2: Parameters tab settings (continued)

Field name	Description
Protocols	When the parameter is placed in a PCM frame, click on the Package button to display transmission details.
Comment	User-defined text relating to the parameter.

35.4.2 Configuring channels

You must configure channels for both parsing and snarfing. On the Channels tab (see the following figure), complete the fields described in the following table.

Figure 35-6: Channels tab in kSetup

Table 35-3: Channels tab settings

Field name	Description
Channel	Read-only field; the bus for which the physical definition must be configured.
Type	RS-232, RS-422 or RS-485.
BaudRate	Baud rates supported by the module.
Bits	Bits per data word: 7 or 8.
Parity	Parity of the data word: None, Even or Odd.
Sync Interval	Idle time between consecutive characters required before starting new message; 2 to 63ms.

NOTE: When setting Type, the pinout for RS-232 is different than the pinout for RS-422/RS-485.
When setting BaudRate, a channel can also use a non-standard baud rate. You can define non-standard baud rates (tolerance of ±2%) provided the relationship between the highest non-standard baud rate (f1), and the other non-standard baud rate (fNS), obeys the following: fNS = f1 / N (where N is an integer greater than or equal to 2 and less than 4095).
When setting Sync Interval, idle time can be used when there's no unique start sequence on a message. For more information on Sync Interval, see "35.6 Common errors and scenarios for configuring messages" on page 9.

35.4.3 Setting messages

It is necessary to configure messages if you want to use the KAD/UAR/102 to parse data. On the Messages tab (see the following figure), complete the fields described in the following table.

Message Name	Channel	Interpret Sequence As	Start Sequence	Stop Sequence	Bytes	End of Message
*	*	*	*	*	*	*
UAR_0_J8_Msg1	0	ASCII	SEQ	\I	9	Sequence
UAR_0_J8_Msg2	1	ASCII	SEQ	\I	9	Sequence
UAR_0_J8_Msg3	2	ASCII	SEQ	\I	9	Sequence
UAR_0_J8_Msg4	3	ASCII	SEQ	\I	9	Sequence

Figure 35-7: Messages tab in kSetup

Table 35-4: Messages tab settings

Field name	Description
Message Name	Label for the message name. Select the message name and type the new message name.
Channel	Channel to parse. Select the channel from 0 to 3.
Interpret Sequence As	The module can parse hexadecimal or ASCII data stream. Select between ASCII and Hex.
Start Sequence	Start sequence to parse the data stream. If you right-click a message and select Edit Start Sequence, the Match Sequence dialog box displays. 0 to 8 bytes can be defined in the Start Sequence. If 0 byte is used, Sync Interval on the Channels tab has to be configured. For more information, see "35.7 References" on page 14.
Stop Sequence	Parses the data stream. Right-click on the message and select Edit Stop Sequence. Stop Sequence is only used when End of Message is set to Sequence. Stop Sequence can only be set up with one byte.
Bytes	Number of bytes in the full message including the start sequence. Complete this field only when End of Message is set to Length. For more information, see "35.6 Common errors and scenarios for configuring messages" on page 9.
End of Message	You can parse on the message length or on a stop sequence. For more information, see "35.6 Common errors and scenarios for configuring messages" on page 9.

35.4.3.1 Adding data words

To add data words on the parser, double-click a message on the Messages tab and then click Add. For examples of data words, see the following figure. On the Data tab, complete the fields described in the following table.
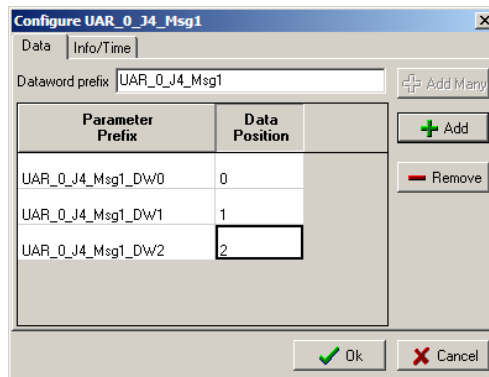

Figure 35-8: Adding data words

Table 35-5: Data word settings

Field name	Description
Parameter Prefix	Parameter name.
Data Position	Position of the data word in the message. If the data position is an even number, the data word is read as: Data N (even) R[15:8] The 7/8 bits of byte M in serial stream. R(15) is the Last Bit Transmitted (LBT). Data N (even) R[7:0] The 7/8 bits of byte M+1 in serial stream. R(15) is the LBT. If the data position is an odd number, the data word is read as: Data N (odd) R[15:8] The 7/8 bits of byte M in serial stream. R(15) is the LBT. Data N (odd) R[7:0] The 7/8 bits of byte M-1 in serial stream. R(15) is the LBT. See Figure 35-12 on page 10 and Figure 35-15 on page 10 for an example of data position.

NOTE: If the start sequence is defined, the first data words contain the start sequence. These specific odd/even registers are used to differentiate between a little/big endian. The odd/even registers facilitate the decommutation depending on whether UNIX or Microsoft is used on the ground station.

The KAD/UAR/102 can parse up to 512 bytes per message per bus. This length includes the bytes used for the start sequence and the end sequence (if it is used).

Data words are only updated when the sequence is correct. Data words are not refreshed until a valid message appears. At power-up, data words have a random value until a valid message arrives. If Length is selected as the End Sequence, add a number of data words less than or equal to the Length, otherwise kProgram generates an error.

For a data word example, see the previous figure.

35.4.3.2 Adding Info/Time message tags

Because the serial stream is asynchronous to the Acra KAM-500, some message tags are available (see the following figure). For bit definitions of these registers, see the *KAD/UAR/102* data sheet.
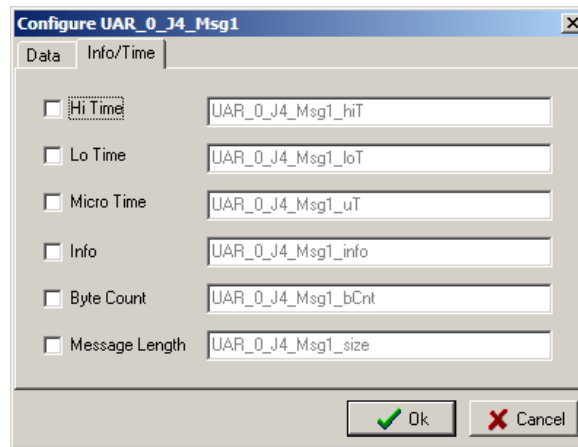

Figure 35-9: Available Info/Time message tags

The serial stream is asynchronous to the Acra KAM-500 so it does not pick up the time and message info automatically. These settings can be added to the stream as extra time/info about the messages.

35.4.4 Reviewing Setup tab requirements

Time Server: Legacy. Must remain at default setting of Slave.

35.4.5 Defining snarfer settings

If configuring the KAD/UAR/102 to snarf as well as parse, you must define settings on the 100% Bus Traffic Monitoring tab as well as the Channels tab. For information on Channel tab settings, see "35.4.2 Configuring channels" on page 4.

To define snarfer settings on the 100% Bus Traffic Monitoring tab (see the following figure), complete the fields described in Table 35-6 on page 8.

Figure 35-10: 100% Bus Traffic Monitoring tab in kSetup

Table 35-6: 100% Bus Traffic Monitoring tab setting

Field name	Description
Parameter Name	You can edit any parameter name.
Active	If active, all snarfer parameters from the module are made available for transmission.
#Words	Allows an irregular PCM frame placement to ensure you get the best possible bandwidth from the PCM.
Mode	Read-only field. For bit definition, see the *KAD/UAR/102* data sheet.
Bus	Read-only field. The bus to be snarfed.
Comment	User-defined text relating to the parameter.

35.5 Setting up the KAD/UAR/102 using DAS Studio 3

You may use DAS Studio 3 software to configure the KAD/UAR/102. Some configuration features that are needed to set up the KAD/UAR/102 include configuring channels, configuring messages, and adding data words to messages. This information is summarized in the "Setting up packages" and "Setting up datalinks" section of the *KAD/UAR/102* data sheet.

35.5.1 About the Settings tab

The Settings tab (see the following figure) displays settings available for configuring the KAD/UAR/102. On the Settings tab, complete the fields described in Table 35-7 on page 8. For information on using the other tabs, that is Processes, Packages, Algorithms, and Documentation, see the *DAS Studio 3 User Manual*.
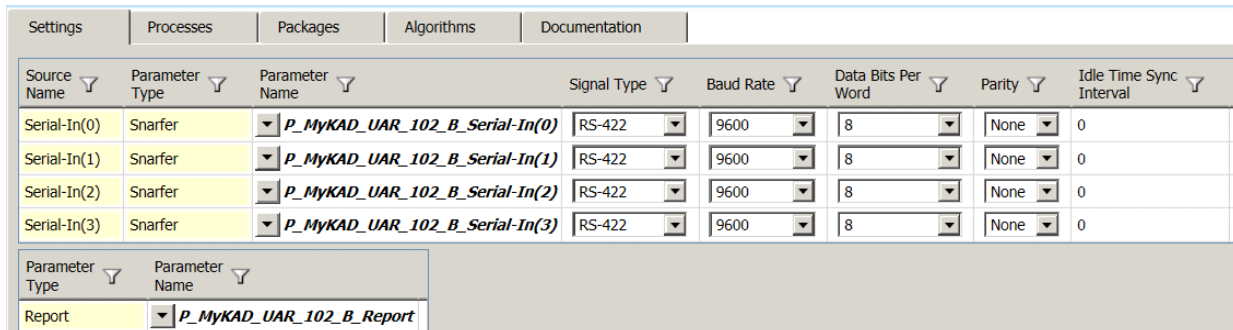

Figure 35-11: Settings tab in DAS Studio 3

Table 35-7: Settings tab features

Field name	Description
Source Name	Read-only field; displays the bus for which the physical definition must be configured.
Parameter Type	Read-only field; displays the snarfer function being configured.
Parameter Name	Snarfer parameter name.
Signal Type	For information, see the "Instrument Settings" section of the *KAD/UAR/102* data sheet. **NOTE:** This field is applicable for snarfing and parsing.
Baud Rate	For information, see the "Instrument Settings" section of the *KAD/UAR/102* data sheet. **NOTE:** This field is applicable for snarfing and parsing.
Data Bits Per Word	For information, see the "Instrument Settings" section of the *KAD/UAR/102* data sheet. **NOTE:** This field is applicable for snarfing and parsing.
Parity	For information, see the "Instrument Settings" section of the *KAD/UAR/102* data sheet. **NOTE:** This field is applicable for snarfing and parsing.

Table 35-7: Settings tab features (continued)

Field name	Description
Idle Time Interval	For information, see the "Instrument Settings" section of the *KAD/UAR/102* data sheet. **NOTE:** This field is applicable for snarfing and parsing.
Parameter Type	Read-only field; the Report register is useful to transmit when monitoring message problems. For details of register definitions, see the *KAD/UAR/102* data sheet.
Parameter Name	Name of the parameter to be transmitted.

NOTE: When setting Signal Type, the pinout for RS-232 is different than the pinout for RS-422/RS-485.

When setting Baud Rate, a channel can also use a non-standard baud rate. You can define non-standard baud rates (with a tolerance of ±2%) provided the relationship between the highest non-standard baud rate (f1), and the other non-standard baud rate (fNS), obeys the following: fNS = f1 / N (where N is an integer greater than or equal to 2 and less than 4095).

When setting Sync Interval, idle time can be used when there is no unique start sequence in a message. For more information, see "35.6.6 Parsing messages without a unique start sequence" on page 13.

35.5.2 Serial Builder applications

The Serial Builder application is used to add and configure messages and parameters on RS-232, RS-422 and RS-485 serial bus monitor modules for the mode parser of a module. For more information, see the "Serial Builder" section in the *DAS Studio 3 User Manual*. For information on package tags, see the "Processes Tab - Process Register Editor and Viewer" section in the *DAS Studio 3 User Manual*.

35.6 Common errors and scenarios for configuring messages

35.6.1 Common errors

If the bus +/- is physically inverted, swap the connections referring to the pinout of the data sheet and retransmit.

If the wrong bus is being parsed, compare the channel on the Channels tab to the physical connection, referring to the pinout on the data sheet.

If the baud rate is set incorrectly, change the setting (see "35.4.2 Configuring channels" on page 4).

If the Serial link ground is not connected to the Acra KAM-500 ground (GND), refer to the pinout on the data sheet and make the connection.

If termination is not present or double termination is incorrectly used, refer to the pinout of the data sheet and make the connection.

35.6.2 Parsing messages with less than nine bytes

To parse a message, the message must have at least nine bytes. The snarfer can parse any data stream size in bytes.

If a message is five bytes on the bus and the user programs the KAD/UAR/102 with a length of nine bytes, the module parses the message but it also parses the next message. The KAD/UAR/102 continues parsing until it receives the full message length that it has been configured with. This may affect the next message on the bus, because as soon as the parser identifies a valid start sequence received, it ignores everything else after it.

NOTE: If using a KAD/UAR/102/C, the minimum message length is 4 bytes (rather than 9 bytes).

35.6.3 Optimizing parsing

Parsing can be likened to performing a search on a text or binary file. To optimize parsing, you must configure the Start Sequence and Stop Sequence fields with the most information possible as described in the following example.

Two different messages are defined in the bus: Message A (MSGA) - ABCDEFGHI\I; and Message B (MSGB) - ABTUVWXYZ\I. If for one message, the Start Sequence is defined as AB, the KAD/UAR/102 uses the same ID (parser slot) to store the message (see Figure 35-12 on page 10). In this case, the data words for the module return the following:
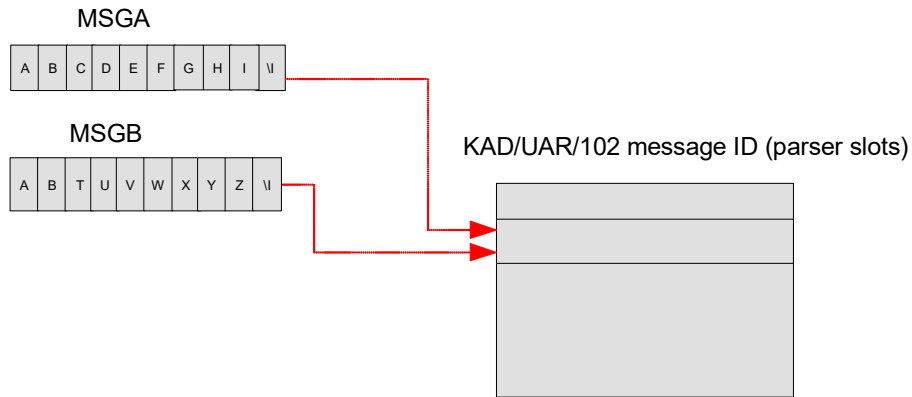
DW0 = AB

DW2 = CD or TU

DW4= EF or VW

MSGA

| A | B | C | D | E | F | G | H | I | \I |

MSGB

| A | B | T | U | V | W | X | Y | Z | \I |

KAD/UAR/102 message ID (parser slots)

Figure 35-12: Using the same message ID (same parser slot for two different messages)

For the following example, the Start Sequence for both messages is defined:

MSGA has a Start Sequence defined as ABC. MSGB has a Start Sequence defined as ABT. Both messages have a Stop Sequence defined as \I (see the following figure).

Parameters	Channels	Messages	Setup	100% Bus Traffic Monitoring

Message Name	Channel	Interpret Sequence As	Start Sequence	Stop Sequence	Bytes	End of Message
*	*	*	*	*	*	*
MSGA	0	ASCII	ABC	\I	9	Sequence
MSGB	0	ASCII	ABT	\I	9	Sequence

Figure 35-13: Defining Start Sequence and Stop Sequence fields with KSM-500

Instrument	Channel	Package	Parsing Mode	Bytes Per Package	Start Sequence Format	Start Sequence	Stop Sequence Format	Stop Sequence
MyKAD_UAR_102_B	Serial-In(0)	MSGA	Start/Stop Sequence	n/a	ASCII	ABC	ASCII	\I
MyKAD_UAR_102_B	Serial-In(0)	MSGB	Start/Stop Sequence	n/a	ASCII	ABT	ASCII	\I

Figure 35-14: Defining Start Sequence and Stop Sequence fields with DAS Studio 3

In this example, the KAD/UAR/102 uses a unique message ID to store each message (see the following figure).

MSGA

| A | B | C | D | E | F | G | H | I | \I |

MSGB

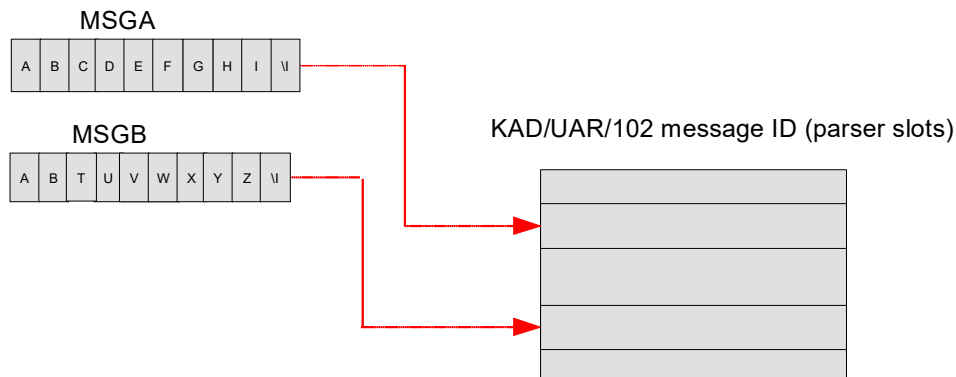| A | B | T | U | V | W | X | Y | Z | \I |

KAD/UAR/102 message ID (parser slots)

Figure 35-15: Using a unique message ID (different parser slots for two different messages)

In this case, the data words for the module return the following:

MSGA

DW0 = AB

DW2 = CD

DW4 = EF

MSGB

DW0 = AB

DW2 = TU

DW4 = VW

35.6.4 Using End of Message values to stop parsing

Parsing of the string can be stopped by using the Length and Sequence values in the End of Message field (see the following figure and Figure 35-17 on page 11). Length and Sequence values are not used by the module to determine the parser slot (Message ID).

In the following example, MSGA and MSGB both use the same Start Sequence value but with a different length (defined in the Bytes field); both messages use the same parser slot (Message ID) in the module.

MSGA: ABCDEFGHI

MSGB: ABCDVWXYZ

WARNING: The module may inadvertently put data words from MSGA into MSGB or vice versa.

Message Name	Channel	Interpret Sequence As	Start Sequence	Stop Sequence	Bytes	End of Message
×	×	×	×	×		×
MSGA	0	ASCII	ABCD	\I	6	Length
MSGB	0	ASCII	ABCD	\I	7	Length

Figure 35-16: Setup of two messages with different lengths using KSM-500

In the following example, MSGA and MSGB both use the same Start Sequence value but with a different Stop Sequence value; both messages use the same parser slot (Message ID) in the module.

MSGA: ABCDEGHIF

MSGB: ABCDWXYZE

WARNING: The module may inadvertently put data words from MSGA into MSGB or vice versa.

Message Name	Channel	Interpret Sequence As	Start Sequence	Stop Sequence	Bytes	End of Message
×	×	×	×	×		×
MSGA	0	ASCII	ABCD	F	9	Sequence
MSGB	0	ASCII	ABCD	E	9	Sequence

Figure 35-17: Setup of two messages with different Stop Sequence values using KSM-500

Instrument	Channel	Package	Parsing Mode	Bytes Per Package	Start Sequence Format	Start Sequence	Stop Sequence Format	Stop Sequence
MyKAD_UAR_102	Serial-In(0)	MSGA	Start/Stop Sequence	n/a	ASCII	ABCD	ASCII	F
MyKAD_UAR_102	Serial-In(0)	MSGB	Start/Stop Sequence	n/a	ASCII	ABCD	ASCII	E

Figure 35-18: Setup of two messages with different Stop Sequence values using DAS Studio 3

35.6.5 Setting variable lengths

Some messages have a variable length. As shown in the following example, the parser cannot differentiate between different message lengths that use the same stop sequence.

MSGA instance 1: ABXYZxywzF

MSGA instance 2: ABXYxywzF

Message Name	Channel	Interpret Sequence As	Start Sequence	Stop Sequence	Bytes	End of Message
*	*	*	*	*	*	*
MSGA	0	ASCII	AB	F	9	Sequence

Figure 35-19: Incorrect setup of variable length (as seen in KSM-500)

Instrument	Channel	Package	Parsing Mode	Bytes Per Package	Start Sequence Format	Start Sequence	Stop Sequence Format	Stop Sequence
MyKAD_UAR_102	Serial-In(0)	MSGA	Start/Stop Sequence	n/a	ASCII	AB	ASCII	F

Figure 35-20: Incorrect setup of variable length (as seen in DAS Studio 3)

If the message is defined as shown in Figure 35-19 on page 12, then the data words for the module return the following:

MSGA:

DW0: AB

DW2: XY

DW4: Zx or xy

DW6: yw or wz

DW8: zF or Fr (with r being a random byte)

WARNING: This type of message cannot be parsed properly. For this case, the snarfer must be used by setting up 100% Bus Traffic Monitoring (see the following figure).

Parameter Name	Active	#Words	Mode	Bus	Comment
*	*	*	*	*	*
UAR_0_J5_Bus_0_AllPass	Yes	1	SNARFER	0	
UAR_0_J5_Bus_1_AllPass	Yes	1	SNARFER	1	
UAR_0_J5_Bus_2_AllPass	Yes	1	SNARFER	2	
UAR_0_J5_Bus_3_AllPass	Yes	1	SNARFER	3	

Figure 35-21: Active field set to Yes on 100% Bus Traffic Monitoring tab in kSetup (KSM-500)

Figure 35-22: Snarfer parameters set using DAS Studio 3

35.6.6 Parsing messages without a unique start sequence

Some messages don't have a unique Start Sequence value defined (see the following figure).

Message Name	Channel	Interpret Sequence As	Start Sequence	Stop Sequence	Bytes	End of Message
×	×	×	×	×	×	×
MSGA	0	ASCII		F	9	Sequence

Figure 35-23: No Start Sequence

The Sync Interval is used when there is no Start Sequence value defined (see the following figure) because the messages in the physical bus do not have a unique start sequence. The Sync Interval can also be used if the value of the Start Sequence is also defined in the message payload.
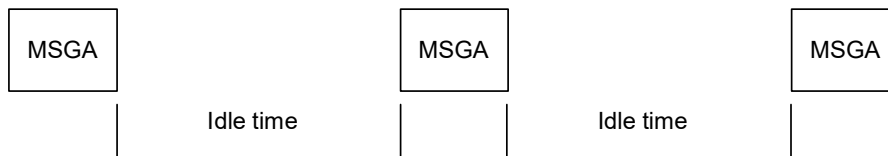

Figure 35-24: Different instances of a message with a defined Sync Interval (idle time)

The Sync Interval can be set to a specific time (see the following figure).

Channel	Type	BaudRate	Bits	Parity	Sync Interval
×	×	×	×	×	×
0	RS-422	9600	8	NONE	12

Figure 35-25: Channel tab setup with a Sync Interval of 12 ms using KSM-500

Source Name	Parameter Type	Parameter Name	Signal Type	Baud Rate	Data Bits Per Word	Parity	Idle Time Sync Interval
Serial-In(0)	Snarfer	P_MyKAD_UAR_102_Serial-In(0)	RS-422	9600	8	None	12

Figure 35-26: Settings tab with a Sync Interval of 12 ms using DAS Studio 3

35.6.7 Using wildcards in Start Sequence

The KAD/UAR/102 can parse messages with a wilcard in Start Sequence. This feature is supported in DAS Studio's Serial Builder using the Start Sequence Format to Hex (see the following figure). Wilcard is for a whole byte, therefore two wildcards are necessary '**'
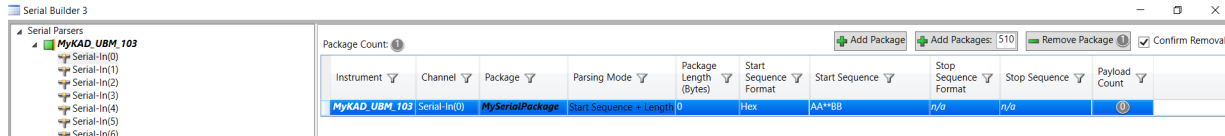

Figure 35-27: Wildcard in Start Sequence example in DAS Studio

In KSM-500, right-click a message, then click **Edit start sequence**, and then select the appropriate byte to wildcard check boxes.
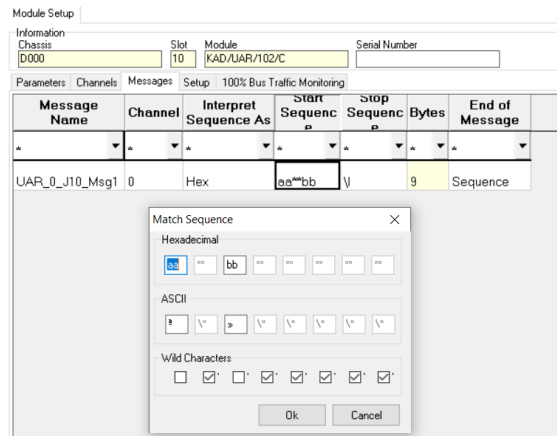

Figure 35-28: Wildcard in Start Sequence example in KSM-500

35.6.8 Data word examples

Consider a stream with ABCDEFGHI:

Data position 0: DW0 is AB

Data position 1: DW1 is BA

Data position 2: DW2 is CD

Data position 3: DW3 is DC

Data position 4: DW4 is EF

Data position 5: DW5 is FE (and so on)

35.7 References

ASCII code reference can be found under http://nemesis.lonestar.org/reference/telecom/codes/ascii.html.