

Document n. 054578-10

Published 7/8/2005

ANNECY ELECTRONIQUE Z.A. Les Marais 74410 St JORIOZ Phone : +33 (0) 450 68 90 65 Fax: +33 (0) 450 68 58 93

INDEX

1	Aim of this document	3
	1.1 Aim of this document	3
	1.2 Reference documents	3
2	Software installation	5
	2.1 Minimum configuration requirements	5
	2.2 Installation	5
3	MuxTrace Software	8
	3.1 Presentation	8
	3.2 Configuration of the CAN network 3.2.1 General configuration of the CAN network 3.2.2 Advanced configuration of the CAN network 3.2.3 CAN network filters	 9 9 10 12
	3.3 Configuration of the NWC network (Diag On Can) 3.3.1 General configuration of the NWC network (Diag On Can)	13 13
	3.4 Configuration of the VAN network . 3.4.1 General configuration of the VAN network 3.4.2 Advanced configuration of the VAN network 3.4.3 VAN network filters	15 16 17
	3.5 Configuration of the LIN network	18 18 18
	3.6 Configuration of the ISO9141 network 3.6.1 General configuration of the ISO9141 network 3.6.2 ISO9141 network advanced configuration 3.6.3 Initialization settings of the ISO9141 network	20 20 22 23
	3.7 Configuration of the J1587 network3.7.1 General configuration of the J1587 network3.7.2 Advanced configuration of the J1587 network	25 25 26
	 3.8 Project configuration 3.8.1 Adding a data base. 3.8.2 Creating a logging file. 3.8.3 - Replay a logging file. 3.8.4 Creating a CAN message. 3.8.5 Creating a VAN message. 3.8.6 Creating a LIN message. 3.8.7 Creating an ISO message. 3.8.8 Creating an NWC message. 3.8.9 Creating a J1587 message. 	27 28 29 30 31 32 34 35 37
	3.9 Interactive generator.3.9.1 Posting of the interactive generators windows3.9.2 Interactive generator window.	39 39 39

3.	.10 Replay asc files 3.10.1 Configuration of replay functionality 3.10.2 Tool bar for the replay	40 40 42
3.	.11 Display of signals. 3.11.1 Creating a list of signals 3.11.2 Visualizing the signals	43 43 44
3.	.12 Graph display of signals 3.12.1 Creating a list of signals 3.12.2 Visualizing the signals	45 45 46
3.	.13 Digital Inputs	47
3.	.14 Digital Outputs	48 48 49
3.	.15 Analog inputs (ANA)	50
3.	.16 Programming module 3.16.1 Starter Development Kit 3.16.2 Library entry points 3.16.3 Accessible functions since the library	51 51 52 56
3.	.17 Access mode to the cards. 3.17.1 Exclusive mode 3.17.2 Shared mode	60 60 60
3.	.18 Expert mode 3.18.1 Expert mode Single license 3.18.2 Expert mode Multiple license	61 61 61
3.	.19 Execution 3.19.1 Visualization parameters	62 63 64 64 64 64
List	t of versions	66

1 Aim of this document

1.1 Aim of this document

The aim of this document is to give the required information for using the MuxTrace programme, which allows the user to control communication channels CAN, CAN fault tolerant, LIN, VAN, ISO9141, Diagnostics On CAN and NMEA0183 and to visualize the state of bus errors thanks to a user friendly graphic interface.

The available functions are:

- Simultaneous management of multi channel and multi protocols.
- Independent channel configuration and triggering.
- Setting *Transmission* function (Period, transmission conditions with key)
- Setting *Reception* function (Acceptance filter, spy mode)
- Permanent indication of frame characteristics and their content in hexadecimal
- Setting acquisition modes
- Recording of measurements configuration

This program is available with all EXXOTEST's network access boards.

1.2 <u>Reference documents</u>

ISO11898 : Road vehicles -- Interchange of digital information -- Controller area network (CAN) for high-speed communication

ISO 11519-2 : Road vehicles -- Low-speed serial data communication -- Part 2: Low-speed controller area network (CAN)

LIN V1.2 : Specifications package

ISO 11519-3 Road vehicles -- Low-speed serial data communication -- Part 3: Vehicle area network (VAN)

ISO 9141 : Road vehicles - Diagnostic System – Characteristics of the numerical data exchange

ISO 9141-2 : Road vehicles - Diagnostic System – Characteristics CARB of the numerical data exchange

ISO 14230 - 1 Physical layer (Keyword Protocol 2000 - Part 1).

ISO 14230 - 2 Data Link layer (Keyword Protocol 2000 - Part 2).

ISO 14230 - 3 Application layer (Keyword Protocol 2000 - Part 3).

ISO 15765-1 Road vehicles – diagnostics on CAN – Part 1 : General information

ISO 15765-2 Road vehicles – diagnostics on CAN – Part 2 : Network layer services

ISO 15765-3 Road vehicles – diagnostics on CAN – Part 2 : Application layer

ISO 15765-4 Road vehicles – diagnostics on CAN – Part 4 : Requirements for emission related systems

OSEK/VDX Network management V 2.5

2 Software installation

2.1 Minimum configuration requirements

Operating system

- Windows 95
- Windows 98
- Windows NT
- Windows 2000/Me/XP

The minimum recommended configuration is the following:

- PC type computer equipped with a Pentium microprocessor (PIII 600 or over recommended) with CD ROM reader

The performance of the MuxTrace program depends on type of PC used and it might be altered depending on the computer configuration

- Another application working simultaneously with the MuxTrace program
- Screen saver
- Antivirus program
-

2.2 Installation

Go to the MuxTrace directory in the CDROM and double click on SETUP.EXE



Choose directory for files to be installed



Click Next

🚰 Setup - MPX Data Tracking v4.30		X
MPX Data Trac	king v4.30	
5	tup	X
	Select Start Menu Folder Where shou'd Setup place the program's shortcuts?	
	Select the Staft Menu folder in which you would like Setup to create the progra shortcuts, then click Next.	ante
	MPS Deta Tracking	
	Accessories Addah Addah Bioadiwe Clev-C++ Bothen C+-Baken 4 CM-Make CAMe addo 2.0.0 CAMe addo 2.0.0 CAMe addo 2.0.0 Compadia	•
	Del Modem Dr.Hold DI. Partiti Doleko device	<u>×</u>
	CBack Next>	Cancel

Click Finish



Installation is completed



3 MuxTrace Software

3.1 Presentation

The MuxTrace program is organized according to a project which consists of :

- General configuration of the project,
- Configuration of each network,
- Possibility to add an user's program,
- Configuration of signals to be visualized,
- Configuration of digital inputs
- The project configuration allows the user to define both the networks that shall be displayed and messages to send.
- The configuration of the networks allows the user to define the characteristics of each network CAN, LIN, ISO, J1587, VAN or NWC, baud rate, spy mode, sample points and other settings specific to each network, as well as the list of messages sent.
- The programming makes it possible in the form of a DLL associated with MuxTrace to carry out for example complex scenarios or to start recordings on particular conditions.
- The configuration of the signals allows the user to define a classification and a list of the signals found in the data bases. The signals coming from the networks will have to be decoded.
- The configuration of digital inputs allows the user to define the surveillance of these inputs.
- The configuration of digital outputs allows the user to define their remote triggering.

Each project can be logged to a file (*.MTP) and like that they can be used later.

MuxTrace features advanced functions:

- managing data bases,
- displaying signals present in the data bases,
- surveillance of digital inputs,
- managing the communication layer Diag On Can (*Iso15765-2*),
- saving messages into text files

All these functions are available on the Expert Mode of the MuxTrace (cf *Expert mode*).

3.2 Configuration of the CAN network

This configuration depends on the type of board installed in the PC. Up to 4 CAN networks can be set up. This configuration will allow the user to choose the configuration of the different parameters related to the CAN bus.

3.2.1 General configuration of the CAN network

Leavenced Lines	* I					
Main configuration						
Bus name						_
2. J. A. (11.2.1.)	250.000		11 - 42	1.00	0	
Saudrate (KDII/S)	1200.000	San	ipie poir	nt (~6)		81
GJW (Synchronisation)	1 -		Spy moo	le		
Network and the fact	1000			🛷 Auto di	ataction	
statistics refresh (ms)	11000	1.	10	C Adio di	election	_
Bus tune						
Bus type		7				
Bustype CAN type High	Speed 💌	Edg	e	S	ope	¥
Bus type CAN type High	Speed 💌	Edg	e	s	ope	
Bustype CAN type High Available setup	Speed 🗾	Edg	e	s	ope	×
Bus type CAN type High Available setup Sample point (%)	Speed	Edg	e SPL	SI	ope TSEG2	
Bus type CAN type High Available setup Sample point (%) 50	Speed	Edg	e SPL	S <u>TSEG1</u> 7	ope TSEG2 8	•
Bus type CAN type High Available setup Sample point (%) 50 50	Speed _	Edg BRP 2 4	e SPL 1 1	51 TSEG1 7 3	ope TSEG2 8 4	
Bus type CAN type High Available setup Sample point (%) 50 50 56	Speed SJW	Edg BRP 2 4 2	e SPL 1 1 1	5 TSEG1 7 3 8	ope TSEG2 8 4 7	
Bus type CAN type High Available setup Sample point (%) 50 50 56 62	Speed SJW	Edg BRP 2 4 2 2	e SPL 1 1 1 1	5 TSEG1 7 3 8 9	ope TSEG2 8 4 7 6	
Bus type CAN type High Available setup Sample point (%) 50 50 50 56 62 62	Speed	Edg BRP 2 4 2 2 4 2 4	e SPL 1 1 1 1 1 1	SI 7 3 8 9 4	ope TSEG2 8 4 7 6 3	
Bus type CAN type High Available setup Sample point (%) 50 50 50 56 62 62 68	Speed	Edg BRP 2 4 2 2 4 2 4 2	e SPL 1 1 1 1 1 1 1	SI 7 3 8 9 4 10	ope TSEG2 8 4 7 6 3 5	
Bus type CAN type High Available setup Sample point (%) 50 50 56 62 62 62 63 75	Speed	Edg BRP 2 4 2 2 4 2 2 4 2 2 2 2	e SPL 1 1 1 1 1 1 1	5 7 3 8 9 4 10 11	ope TSEG2 8 4 7 6 3 5 4	
Bus type CAN type High Available setup Sample point (%) 50 50 56 62 62 62 62 68 75 75	Speed	Edg BRP 2 4 2 4 2 2 4 2 2 4 2 4	e SPL 1 1 1 1 1 1 1 1 1	TSEG1 7 3 8 9 4 10 11 5	ope TSEG2 8 4 7 6 3 5 4 2	*
Bus type CAN type High Available setup 5 50 50 50 56 62 62 62 62 62 63 75 81	Speed	Edg BRP 2 4 2 4 2 4 2 2 4 2 2 4 2 2 4 2 2 4 2 2 4 2 2 4 2 2 4 2 2 4 2 2 4 2 2 4 2 2 4 2 2 4 2 2 4 4 2 2 4 4 2 2 4 4 2 2 4 4 2 2 4 4 2 2 4 4 2 2 4 4 2 2 4 4 2 2 4 4 2 2 4 4 2 2 4 4 2 2 2 4 4 2 2 2 4 4 2 2 2 4 4 2 2 2 4 4 2 2 4 4 2 2 4 4 2 2 2 4 4 2 2 4 4 2 2 2 4 4 2 2 2 4 4 2 2 2 4 4 2	e SPL 1 1 1 1 1 1 1 1 1 1 1	SI 7 3 8 9 4 10 11 5 12	TSEG2 8 4 7 6 3 5 4 2 3	

Bus name Baud rate	Logical name given to the network. It is shown while this network is in use. Network baud rate expressed in Kbit/sec
SJW	Synchronization jump width
Spy mode	Not selected: the board behaves like a CAN station, active on the network. It can send messages as well as generate acknowledgement and frame errors. Selected: the board is totally inactive on the network. It is impossible to either send messages or generate acknowledgement or frame errors.
Auto detection	Seek automatically the bus bit rate.
Statistics refresh	Shows the period of statistics refresh in the bus. Value 0 deactivates the statistics.

Bus type	Choice between bus CAN high speed and CAN low speed – fault tolerant. This choice depends on the type of board used. It is carried out either by the programme or by a staple on the board.
Edge	It selects for the bus CAN high speed the transition slope on the CANH and CANL lines.
Available setup	The sample point is introduced so as to select, depending on the baud rate, the different possible configurations of the TSEG1, TSEG2 and BRP parameters.

3.2.2 Advanced configuration of the CAN network

BRP (Divider) SPL (Sample)	Implication Implication <t< th=""><th></th></t<>	
SJW (Synchron Baudrate (kbit/ Clock (kHZ)	nisation) 1 式 s) 500.000 Sample point (%) 8000	75
Bus Off Automatic n Acceptance fill Filter (Binary)	estart after bus off ter T Extended (29 bits) Ident 0x 000	
Bus Off Automatic r Acceptance fill Filter (Binary)	estart after bus off ter Extended (29 bits) Ident 0x 000 CX Mask 0x 000)))
Bus Off Automatic r Acceptance fill Filter (Binary)	estart after bus off ter Extended (29 bits) Ident 0x 000 X Mask 0x 000 CANLS)

BRP	Clock divider. The divider allows the user to define the time basis of the CAN protocol controller from its clock. This time basis is expressed in quantum and it is used as a reference for the TSEG1, TSEG2 and SJW settings.
SJW	Synchronization jump width (in quantum)
TSEG1	Time previous to sample point (in quantum)
TSEG2	Time post sample point (in quantum)
SPL	Number of sample points
Bus off	This parameter enables the beginning of communication after the CAN bus controller goes into disconnected mode « bus

off ».

parameters.

- Acceptance filter The acceptance filter allows the user to decrease the number of messages received by the PC by placing a reception filter on the messages the user does not want to deal with.
- **Extended** Filter of identifiers, standard (11 bits) or extended (29 bits)
- Binary filterBit-to-bit filter of identifiers that the user wishes to filter.
0 : Filter which lets through identifiers with bit 0
1 : Filter which lets through identifiers with bit 1
X : No filterThe filter can also be obtained with Ident and Mask
- **Pull up resistor** This setting allows the user to dynamically configure the value of pull-up and pull-down resistors in the CAN low speed. The global impedance of the CAN low speed network depends on the number of ECUs present in it. Note: This setting is only used with certain type of material.

3.2.3 CAN network filters

The filters allow displaying only the messages to analyze. Filtering is defined by a standard or wide identifier.

Filters	-		
ype of filter	Refuse frames		-
💑 🛓	215 Modify		<u>Suppress</u>
Name		Ident	
RepRegimeMot		UU4110F1	1

Type of filter

Filter definitions:

<u>Accept all frames</u>: No active filters <u>Refuse frames</u> : Only the specified frame are not displayed <u>Accept only frames</u> : Only the specified frame are displayed

3.3 Configuration of the NWC network (Diag On Can)

This configuration depends on the type of board installed in the PC, the number of NWC networks depends on the number of CAN networks found on the board, which will allow the user to choose the configuration of the different parameters related to the CAN bus.

3.3.1 General configuration of the NWC network (Diag On Can)

wain coniguration	r	- 22
Bus name		
Communication setup		
Adressing mode	Physical 💌	
Adressing format	Normal	
Communication	Full duplex	
Communication mode Show the detail of Display the First Fr	Variables frames 💽 the communication rame	
Communication mode Show the detail of Display the First Fr Spy mode setup Use the spy mode	Variables frames	
Communication mode Show the detail of Display the First Fr Spy mode setup Use the spy mode Request ident	Variables frames	
Communication mode Show the detail of Display the First Fr Spy mode setup Use the spy mode Request ident Response ident	Variables frames	
Communication mode Show the detail of Display the First Fr Spy mode setup Use the spy mode Request ident Response ident Mask	Variables frames	
Communication mode Show the detail of Display the First Fr Spy mode setup Use the spy mode Request ident Response ident Mask Communication mode	Variables frames	

Bus name	Logical name given to the network. Shown during execution.
Addressing mode	Physical or functional addressing
Addressing format	Address normal, normal fixed, extended or mixed
Communication	Half Duplex or Full Duplex
Communication mode	Transmission of variable frames or of fixed frames of 8 bytes.

Show the detail of the communication	Shows the detail of the segmentation in the CAN network visualization window associated to the NWC network.
Display the First Frame	Displays <i>First Frame</i> frames on the network in the visualization window.
Use the spy mode	Activates the bus's spy mode.
Request ident	Request identifier or FC response
Response ident	Response identifier or FC request
Mask	Mask that allows user to define 2 groups of identifiers to be interpreted.
Communication mode	Transmission of variable frames or of fixed frames of 8 bytes.
Show the detail of the communication	Displays details of the segmentation in the CAN network visualization window associated to the NWC network.
Display the First Frame	Displays <i>First Frame</i> frames on the network in the visualization window.

3.4 Configuration of the VAN network

This configuration depends on the type of board installed in the PC. Up to 4 VAN networks can be set up. It allows the user to choose the configuration of the different parameters related to the VAN bus.

3.4.1 General configuration of the VAN network

ain Advanced Filters	
Main configuration	
Bus name	
Baudrate kTS/s 125	👻 🔽 Spy mode
Statistics refresh (ms)	1000 👋 Auto detection
Reception mode	
C RxD0 (Différential)	C RxD2 (DataB)
C RxD1 (Data)	 Automatic
Line diagnosis	
TIP (Transmission In Proc	gress)
SDC (Diagnostic clock)	
SDC time (ms)	00
on a data (the)	

Bus name	Logical name given to the network. Shown during execution.
Baud rate	Network baud rate expressed in kilotime slot/sec
Auto detection Spy mode	 Seek automatically the bus bit rate. Not selected: the board accepts all messages transiting the network. Selected: the board is totally inactive in the network. No acknowledgement when the frame is received. However, it is possible to send messages and to respond in the frame.
Statistics refresh	Shows the period of statistics refresh in the bus. The value 0 deactivates the statistics.
Reception mode	 Reception line of protocol controller RXD0 : Forced reception in differential mode RXD1 : Forced reception on the data line RXD2 : Forced reception on the datab line Automatic: the choice of reception line is carried out according to a protocol controller's internal algorithm. Going from one line to the other is done automatically.

TIP	Diagnosis being sent. This parameter depends on the desired application. It is generally used to detect a line opening.
SDC	Validation of the diagnostic clock. When an error, on the line, is detected, the SDC clock allows the user to monitor communication in both lines, so as to indicate the fact of being back in differential mode if the error disappears. Caution: This parameter depends on the charge and on the bus baud rate.
Clock	Value of the diagnostic clock

3.4.2 Advanced configuration of the VAN network

Lines Inverted Rx Inverted Tx Acceptance filter Filter (binary) Ident 0x 000 Mask 0x 000	Module type Autonomous Synchronous	Coding Manchester C Pulsed
Acceptance filter Filter (binary) Ident 0x 000 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	Lines Inverted Rx Inverted Tx	Transmitting retries
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	Acceptance filter Filter (binary)	Ident 0x 000
	x0000000000	Mask 0x 000

Module type	Autonomous: When sending out a message, the board can generate a SOF (Start of frame). The message is sent out immediately.
	Synchronous: When sending out a message, the VAN protocol controller cannot generate a SOF and monitoring the network. When a SOF of a message from another station transits onto the network, the message is transmitted and clashed with the current message.
Coding	Manchester : By default, coding used by the VAN interface line found on the boards.

	Pulsed : Coding that might be used by an external line interface (optical fiber for example).
Inverted Rx line Inverted Tx line	Possibility to invert the logical status of recessive and dominant levels (for external line interface only)
Number of retries	Number of retries during a transmission in the event of error.
Acceptance filter	The acceptance filter allows the user to decrease the number of messages received by the PC by placing a reception filter on the messages the user does not want to deal with.
Binary filter	Bit-to-bit filter of identifiers that the user wishes to filter. 0 : Filter which lets through identifiers with bit 0 1 : Filter which lets through identifiers with bit 1 X : No filter
	The filter can also be obtained with Ident and Mask parameters.

3.4.3 VAN network filters

The filters allow displaying only the messages to analyze. Filtering is defined by a standard or wide identifier.

Type of filter	Refuse frames		
Name MonMessage		ldent 559	
09-0-0-0-0-0-0-0-0-0		ng	19-19-19-19-19-19-19-19-19-1

Type of filter

Filter definitions: <u>Accept all frames</u>: No active filters <u>Refuse frames</u> : Only the specified frame are not displayed <u>Accept only frames</u> : Only the specified frame are displayed

3.5 Configuration of the LIN network

This configuration depends on the type of board installed in the PC. Up to 2 LIN networks can be set up. It will allow the user to choose the configuration of the different parameters related to the LIN bus.

3.5.1 General configuration of the LIN network

1	IN n°1 bus configuration
	Main Advanced
	Main configuration Bus name Baud rate kbit/s 19200 Statistics refresh (ms) 1000 Warning : Same baud rate for LIN1 and LIN2 (PCI board)
	LIN Revision Version 1.X Version 2.X Pull-up resistor Slave (30 Ko) Master (1 Ko)
-	QkCancel
Bus name	Logical name given to the network. Shown during execution.
Baud rate	Network baud rate expressed in kbit/sec
Statistics refresh	Shows the period of statistics refresh in the bus. Value 0 deactivates the statistics.
Revision LIN	Version 1.X : The calculation of the CRC is in conformity with the revision LIN 1.0, 1.2 and 1.3

Version 2.X : The calculation of the CRC is in conformity with the revision LIN 2.0.

Pull up resistorThis setting allows the user to dynamically configure the value
of pull-up resistors.

3.5.2 Advanced configuration of the LIN network

Advanced configuration Configuration Default delay before bus idle New value (ms)	1302
Baud rate kbit/s	19200

Default delay before bus idle By default, the LIN standard specifies that the delay before the detection of communication loss equals the duration of 25000 bits. For example, for a baud rate of 19200 kBit/sec (1 bit=52 μ Sec) the duration of the detection of communication loss is 52 μ Sec*25000, i.e. 1302 ms. For a unitary test, not included in the context of total integration, it is possible to adjust this setting according to the configuration. Note : 0 means an endless time out.

July 2005

3.6 Configuration of the ISO9141 network

This configuration depends on the type of board installed in the PC. Up to 2 ISO9141 networks may be set up. It will allow the user to choose the configuration of the different parameters related to the ISO9141 (K & L) bus.

3.6.1 General configuration of the ISO9141 network

Bus name	
Baud rate kbit/s	10400 💌 🔽 Spy mode
Statistics refresh (ms)	1000
Header type	10LLLLLL Physical addressing
Warning: Same baud r	ate for ISO1 and ISO2 (USB board)
VP1/4 : Inter byte time Software analysis	out for request or response (ms) [10
VP1/4 : Inter byte time Software analysis Communication parame	e out for request or response (ms) 10 eters (tester mode) x F1 Address of target 0x 33
VP1/4 : Inter byte time Software analysis Communication parame Address of source 0 VP1 : Inter byte time ou	e out for request or response (ms) 10 eters (tester mode) x F1 Address of target 0x 33 ut for ECU response (ms) 20
VP1/4 : Inter byte time Software analysis Communication parame Address of source 0 VP1 : Inter byte time ou VP2 : Time between te	e out for request or response (ms) 10 eters (tester mode) k F1 Address of target 0x 33 ut for ECU response (ms) 20 ster request and ECU response(s) (ms) 1000
VP1/4 : Inter byte time Software analysis Communication parame Address of source 0 VP1 : Inter byte time ou VP2 : Time between te VP3 : Time between E	e out for request or response (ms) 10 eters (tester mode) x F1 Address of target 0x 33 ut for ECU response (ms) 20 ster request and ECU response(s) (ms) 1000 CU response and tester request (ms) 55

General configuration

Bus name Logical name given to the network. Shown during execution.

- Baud rate Network baud rate expressed in kbit/sec
- Spy modeSelected : Spy mode.Not selected : Tester modeSpy mode :The MuxTrace program analyses the
communication between a tester tool and an ECU.Tester mode:The MuxTrace program simulates the presence
of a Tester tool and allows the user to send an ECU diagnosis
request.

Statistics	Shows the period of statistics refresh in the bus.
refresh	Value 0 deactivates the statistics.
Header type	This parameter selects the coding type of a header byte of an ISO9141 or ISO 14230 message. 10LLLLL Physical addressing 11LLLLLL Functional addressing 01LLLLLL Exception mode CARB 00LLLLLL No address information

Communication parameters (spy mode)

WP1/4 Maximum value of inter-byte delay of a request or a response. This parameter is used for detecting the end of a request or a response.

Software analysis If this option is not ticked, the end of frame detection is done through WP1/4 timeout. If this option is ticked, the end of frame detection is done through WP1/4 timeout but also by analyzing the first header bytes that comprise the frame's length

<u>Note</u>: The end of a request or a response in the bus is detected when the WP1/4 times are exceeded. In order to ensure the proper functioning of the program, it is important that the times between a request and a response (WP2) or between a response and a new request (WP3) are above the inter byte times WP1/4.

Communication parameters (tester mode)

Address of source Address of target WP1	Source address in hexadecimal (tester's address) Target ECU's address in hexadecimal. Inter-byte time-out of the ECU's response in ms
WP2	Time-out between a tester request and the ECU's response in
WP3	ms Delay between a response from the ECU and a new request from the tester in ms
WP4	Inter-byte delay of the tester's response.

3.6.2 ISO9141 network advanced configuration.

511-1	bus coning	uration		
Main	Advanced	Initialization settin	gs	
Ons	tart communi	cation		
₽	Send an initia	alization sequence		
Filter	-			
Г	Disable displa	ay of Tester Present	message	
				2

On start communication

On start communication	Selected : Sends an initialization sequence Not selected : No sending When the analysis starts, validating this parameter allows the user to send an initialization sequence by means of the information Source address and Target address. The start of request is of the « init fast » or « 5 bauds init » type, followed by a Start Communication request (code 0x81).
Filter	
Tester Present	Selected : The request or response Tester Present is not displayed Not selected : The request or response Tester Present is displayed The request Tester Present is used for keeping communication but has no other applicable functions. It can be filtered so that the user does not have too much

information.

3.6.3 Initialization settings of the ISO9141 network

LATE AT 1	
Fast initialization G 5 bauds initialization	
Fast initialization	
TiniL : Time for Wake up Pattern - bus low (ms)	25
TWup : Time of total Wake up Pattern (ms)	50
TIdle : Time before sending Wake up Pattern (ms)	300
5 bauds initialization	
5 bauds address 0x 33 Parity No	o change 🖉 💌
W1 : Time out before synchronisation pattern (ms)	60
W2 : Time out before key byte 1 (ms)	20
W3 : Time out before key byte 2 (ms)	20
W4a : Time between key byte 2 and inv. key byte 2 (ms)	25
W4b : Time out before inverted address (ms)	50
W5 : Idle time before sending 5 bauds address (ms)	300
P0 : Time before first start communication request (ms)	5
	1192

Initialization type

Initialization mode	Fast initialization or 5 bauds initialization
Fast initialization	
TiniL	Time for Wake up pattern in ms
Тwup	Duration in ms before sending the first request. This duration consists of the low and high levels of the fast initialization sequence.
Tidle	Inactivity time of the bus before sending the « wake up pattern »

5 bauds initialization

5 bauds address	Target address in hexadecimal sent at 5 bauds
Parity	 None : 5 bauds address bauds sent without changes Even : 5 bauds address sent with an even parity Odd : 5 bauds address sent with an odd parity
W1	Time between the end of the address byte and the beginning of the synchronization pattern
W2	Time between the end of the synchronization pattern and the beginning of key byte 1
W3	Time between key byte 1 and key byte 2
W4a	Time between key byte 2 (coming from the UCE) and its inversion, carried out by the MuxTrace
W4b	Time between inverted key byte 2 and the inverted address coming from the UCE
P0	Time between reception of the inverted address and the beginning of the emission of the StartCommunication request

3.7 Configuration of the J1587 network

This configuration depends on the type of board installed in the PC. Up to 4 J1587 networks can be set up. It allows the user to choose the configuration of the different parameters related to the J1587 bus.

3.7.1 General configuration of the J1587 network

Busname	0000	
Statistics refresh (ms)	1000	

Bus name Logical name given to the network. Shown during execution.

Baud rate Network baud rate expressed in kbit/sec

StatisticsShows the period of statistics refresh in the bus.refreshValues 0 deactivate the statistics.

3.7.2 Advanced configuration of the J1587 network

J158	7 n°1 bus confi	guration		_ 🗆 ×
Main	Advanced			
Adv	vanced configural	ion		
Dela	Delay before bus idle		5000	
- MIE) Acceptance filte	n- 00	 Filter by pass 	
Ma	ask.	0x 00	C Filter by block	
				X Lancel

Delay before bus
idleDelay before detection of loss of communication.
Note : value 0 means an endless time out.
Parameter which allows the user to add acceptance filter
(filter by pass / filter by block) to a message identifier or a
family of message identifiers so as not to overcharge the
display coming from the network.

3.8 Project configuration

The project configuration establishes how the Muxtrace will work :

- Visualization parameters
- List of used buses,
- Sending messages
- Choice of data bases
- Logging to a file
- Replay a logging file

Project name Board USB_t	MPX_DIAG (2 CAN, 4 ISO)	Cyclic up 3141) board	date (ms) ize (frames)	100 1024
CAN C	Common Vused bus Database Log to file	500,000 kbit/s C:\Cédric\Présentations\Demo	75 %	V27.DBC
LIN 1 LIN 2 LIN 3 LIN 4 LIN 4 LIN 4 LIN 4 LIN 4	Replay Transmit list	tor: Modify Sup	oress	nteractive generator DN
ISO 2 ISO 3 ISO 4 ISO 4 ISO 4 ISO 4 ISO 4 ISO 4 ISO 4 ISO 4 ISO 4 ISO 7 ISO 7 INTERACTOR	IS_Dyn_BV_389	Service Transmit message Transmit message Transmit message Transmit message Transmit message	Ident 889 001 002 005 005	Send '12ms' '100ms' '100ms' '2222ms' '2222ms'

Project name	Logical name given to the project. Shown when executed.
Cyclic update	In the event of a fixed visualization, this parameter allows the user to periodically update the visualization.
Memory size	In the event of a sequential visualization, this parameter indicates the name of the messages saved in the memory to be visualized.
Used bus	Selection of buses used during execution. A display window per bus is created.
Database	Selection of a base in .DBC or .DBV format.
Log to file	Selection of a file and its format in order to save the

messages transiting the network.

ReplaySelection of file in order to replay messages contained.Transmit listCreation, suppression and modification of messages to be
sent by the MuxTrace

Performance

The performance of the MuxTrace program depends on the type of PC used, as well as on the frequency in which the information coming from the bus is displayed. In order to optimize this performance, we recommend to :

- Increase cyclic update to a maximum level
- Decrease memory size to a minimum level

3.8.1 Adding a data base

A data base associated to a network will allow the user to work with physical values of signals. The data bases supported by the MuxTrace program are in .DBC, .DBV, DBL or DBx format.

3.8.2 Creating a logging file

The messages received can be logged onto a text file for subsequent analysis.

Main configuration			
Filename			
Output file name	 Overwrite 	C Increment	Q
Trigger condition			
Continously	C	🖹 On start	
🔿 On stop	0	🖹 On trigger programm	
Pre Trigger 0	F	Posttrigger 0	
Identifier format		Data format	
Hex	6	🗈 Hex	
C Decimal	0	🕽 Decimal	
🗖 Database			
TimeStamps	F	Format of time	
Absolute	C	HH:MN:SS:MS	
C Relative	0	SS:MS	
TimeStamps	C A	Format of time THH:MN:SS:MS SS:MS	

Enable logging Enables logging messages transiting the network to a file

Filename	Name of file out
Trigger conditions	<u>Continually</u> : The recording is carried out between the starting and the stop of measurement. <u>On start</u> : The recording is carried out between the starting of measurement and stops n messages after (defined by post trigger) <u>On stop</u> : The recording is carried out between n messages (defined by pre trigger) and the stop of measurement and stops. <u>Trigger</u> : The recording is carried out between N messages (defined by pre trigger) and N messages (defined by post trigger) around the trigger of release defined by programming (see chapter programming)
Format of identifier	Hex : Identifier in hexadecimal Decimal : identifier codified in decimal (EXCEL) Database : If the data base exists, the logical name of the message is registered.
Format of data	Hex : Data in hexadecimal Decimal : Data in decimal (EXCEL)
Timestamps	Absolute : Each event is dated in reference to the start of communication Relative : Each event is dated in reference to the previous event
Format of time	HH :MN :SS :MS (Hours, minutes, seconds, milliseconds) SS MS (EXCEL type usage):

3.8.3 - Replay a logging file

The messages recorded in a asc file, can be replayed in order to carry out an analysis.

Main Replay a record file	
File	
Frames Replay all received frames (Tx)	Network
	All All

- **Enable Replay** Enables to replay recorded messages
- **Replay Tx Frames** Replay all the messages marked like Tx in this file
- **Replay Rx Frames** Replay all the messages marked like Rx in this file

Replay Selected bus Select a bus to replay from the file.

3.8.4 Creating a CAN message

		Signals values	
lame	IS_Vers_BSI_112	Edition_calib	0
Frame us	sed in the interactive generator	Edition_soft	0
Transmit	on key E Transmit periodically (ms) 1000	Version_appli	0
	Delay after start (ms)	Version_soft	0
CAN frame o	opliquation	Version_systeme	0
dentifier	0x 112 996 Extended (29 bits)	Vers_date2_annee	0
ervice		Vers_date2_jour	0
) a ka Lawatka		Vers_date2_mois	0

Name	Logical name given to the message. Shown when this list is displayed.
Frame used in the interactive	This message will be able to be modified in the interactive generator.
Transmit on key	Transmits every time the selected key of the message is pressed.
Transmit	Periodical transmission of the message in milliseconds.
Frame configuration	Frame selection in a data base. Frame configuration is done automatically.
Identifier	Value of message identifier
Extended	Selection of type of identifier : Standard (11 bits) or Extended (29 bits)
Service	CAN service of the message : - Transmit message

- Remote frameData lengthLength of the data included in the messageDataValue of the data in hexadecimalSignals valuesEnter signals values to be codified in the present frame.

3.8.5 Creating a VAN message

Informations -		Signals values
Name	DEMANDES_CLIMATISATION	
Frame use	d in the interactive generator	
🗂 Transmit o	n key 🗧 🗂 Transmit periodically (ms) 1000	CMD_CLIM_DELESTAGE
	Delay after start (ms)	CMD_CLIM_DELESTAGE
VAN frame co	nfiguration	CMD_CLIM_DELESTAGE
dentifier	0x 464 🛛 🙀 🗖 Request acknowlegde	
Service	Transmit message	
)ata length	5 4	CMD_CLIM_LIBRE 0
) ata		CMD_CLIM_LUCH (SANS) C 0
/ala		
		LUM_JOUR (LUX) 500
	lao lao lao lao lao lao lao	PELEC_CLIM (WATT) 0
		Qk ↓ X Canc

Name	Logical name given to the message. Shown when this list is displayed.
Frame used in the interactive generator Transmit on key	This message will be able to be modified in the interactive generator. Transmission every time the selected key of the message is pressed.
Transmit periodically	Periodical transmission of the message in milliseconds.
Frame configuration	Frame selection in a data base. Frame configuration is done automatically.
Identifier	Value of he message identifier

Request acknowledge	Indicates if the message requires request acknowledgement.			
Service	 VAN service of the message Transmit message Request an in frame response In frame response Differed reply message 			
Data length	Length of data included in the message			
Data	Value of the data in hexadecimal			
Signals values	Enter signals values to be codified in the current frame.			

3.8.6 Creating a LIN message

Informations		Signals values
Frame user	d in the interactive generator	
Transmit or	n key 🗧 🗖 Transmit periodically (ms) 1000	
	Delay after start (ms)	
LIN frame conf dentifier Service Data length Data Error type	iguration 0x 00 LIN identifier 0x 00 20 0x 90 Transmit message	

Name	Logical name given to the message. Shown when this list is displayed.
Frame used in the interactive generator	This message will be able to be modified in the interactive generator.
Transmit on key	Transmission every time the selected key of the message is pressed.

Transmit	Periodical transmission of the message in milliseconds.				
Frame configuration	Frame selection in a data base. Frame configuration is done automatically.				
Identifier	Value of the message identifier				
LIN Identifier	Value of the message identifier which includes a field with the length of the message (6 bits)				
Service	 VAN service of the message Transmit message Request an in frame response In frame response 				
Data length	Length of data included in the message				
Data	Value of the data in hexadecimal				
Signals values	Enter signals values to be codified in the present frame.				
Error type	In order to send a protocol test, different frame types with protocol errors can be sent. - No errors - P0 Parity bit error - P1 Parity bit error - CRC error - Synchro byte error - Data length +1 error - Data length +2 error - Data length -1 error - Data length -2 error				

3.8.7 Creating an ISO message

Informations	-										Signals value	s	
¶ame													
Frame used in	the	inter	active	genera	ator								
🥅 Transmit on k	ey			E	ſ	Tra	nsmit pe	eriodica	ally (ms)	1000			
					D	elay af	ter start	(ms)		0			
ISO frame configu	ratio	m –		1							i.		
Address of source	0x [F1	1044		Ade	dress o	f target	0	33				
Service		ЗE	TP		Test	cerPr	esent			-			
Data length	-	1	÷							2- 46			
Data	0x [3E	00	00	00	00	00	00	00				
	1	00	00	00	00	00	00	00	00				
	1	00	00	00	00	00	00	00	00				
	1	00	00	00	00	00	00	00	00				
	1			-				-					
											1	7	1 2000000000000000000000000000000000000
												\checkmark	Ok 🗙 Car

Name	Logical name given to the message. Shown when this list is displayed.
Frame used in the interactive generator Transmit on key	This message will be able to be modified in the interactive generator. Transmission every time the selected key of the message is pressed.
Transmit periodically	Periodical transmission of the message in milliseconds.
Frame configuration	Frame selection in a data base. Frame configuration is done automatically.
Address of source	Address of source transmitted in the message. By default, the address of source corresponds to that programmed in the configuration parameters.
Address of target	Address of target transmitted in the message.
Service	KWP service of the message corresponding to the first byte of data. The proposed services are those ones described by the ISO14230 standard.
Data length	Length of data included in the message (excluding header byte,

address of source, address of target and CRC).

Value of data in hexadecimal. The values will be given Data according to the selected service.

Signals values Enter signals values to be codified in the present frame.

3.8.8 Creating an NWC message

Informations		Signals values
Name j		
Frame used in th	e interactive generator	
Transmit on key	E Transmit periodically (ms) 1000	
	Delay after start (ms)	
NWC frame configur	tion	
Service	Transmit message	
Source addr. C	x 00 Target addr. 0x 00	
Extended addr. 0	Communication ident 000	
ldent (000 1944 Extended (29 bits)	
Flow Control Ident 0	000 Extended (29 bits)	
Block size	0	
As Time out	10 Ar Time out 10	
Ra Time out	20 Pr Time out 0	
Ls I me out (S I Min)		
Number of FC		
Data length		
Data C		
	00 00 00 00 00 00 00	
		J
		✓ <u>D</u> k

Name	Logical name given to the message. Shown when this list is displayed.
Frame used in the interactive generator Transmit on key	This message will be able to be modified in the interactive generator. Transmission every time the selected key of the message is pressed.
Transmit periodically	Periodical transmission of the message in milliseconds.
Service	Type of service: Transmit message or Data reception
Source addr.	Address of source
Target addr.	Address of target
Extended addr.	Extended address

Communication ident	Communication identifier that will transit the network.
Extended	Selection of type of identifier. Standard (11 bits) or Extended (29 bits).
Ident	Frame identifier.
Flow Control Ident	Identifier of the flow control frame.
Frame configuration	Frame selection in a data base. Frame configuration is done automatically.
Block size	Number of consecutive blocks after reception of a flow control frame.
As Time out	Maximum transmission time of the transmitter.
Ar Time out	Maximum transmission time of the receiver.
Bs Time out	Time until reception of flow control.
Br Time out	Time until transmission of flow control.
Cs Time out (STMin)	Time between 2 blocks.
Cr Time out	Time until transmission of Consecutive Frame.
Number of FC	Number of maximum flow control expected.
Data length	Length of data included in the message.
Data	Value of data in hexadecimal. The values will be given according to the selected service.
Signals values	Enter signals values to be codified in the current frame.

3.8.9 Creating a J1587 message

Informations		[Signals values	
Frame used in the interactive	generator			
🥅 Transmit on key	E 📕 🧮 Transmit periodically (ms)	1000		
	Delay after start (ms)	0		
J1587 frame configuration				
MID 0x 00	1995 1995			
Priority 0				
Data length 0				
Data 0x 00 00	00 00 00 00			
00 00	00 00 00 00			
00 00	00 00 00			
·				

Name	Logical name given to the message. Shown when this list is displayed.
Frame used in the interactive	This message will be able to be modified in the interactive generator.
Transmit on key	Transmission every time the selected key of the message is pressed.
Transmit	Periodical transmission of the message in milliseconds.
Frame configuration	Frame selection in a data base. Frame configuration is done automatically.
MID	Value of the message identifier
Priority	Transmission priority of the message [0-7]
Data length	Length of data included in the message.
Data	Value of the data in hexadecimal

Signals values Enter signals values to be codified in the present frame.

3.9 Interactive generator

Interactive generator allows user to modify transmitted messages during the muxtrace run.

3.9.1 Posting of the interactive generators windows

The interactive generators can be used with the messages transmitted on networks CAN, NWC, FLAX, ISO and VAN.

You can activate the interactive generators from this window:

Project configuration		×
Informations Project name Board US	Cyclic update (ms) B_MPX_DIAG (2 CAN, 4 ISO9141) board B_MPX_DIAG (2 CAN, 4 ISO9141) board	100 1024
	Interactive generator List of interactive generators CAN - Display the interactif generator NWC - Display the interactif generator LIN - Display the interactif generator ISO - Display the interactif generator VAN - Display the interactif generator	
		₩ <u>0</u> k

3.9.2 Interactive generator window

			Message		Ident	Netv	lord	10 10	Period	05.0	Кеу	L	ength				Da	atas	1		
					(Hexa.)			12 11		12.62	(ms)			 0	1	2	3	4	5	6	7
	3	10 2	IS_Dyn_CMM_208	208	}	CAN 1	-		1000	Г	E	8	*	00	00	00	00	00	00	00	00
2 🗖	2	18 0	IS_Dat_CMM_488	488	}	CAN 1		Г	1000	Г	E	8	+	00	00	00	00	00	00	00	00
)	5	Cple_MT	_incertains	□ 0 _40			lográ	.r		1						+					
2.07] <u> </u>	romp_co						<u> </u>													

3.9.2.1 Messages

The list of the messages makes it possible to modify during acquisition:

- Identifiers value,
- period of the message,
- the key of emission,
- Message size and message data.

Selection of Frames	Allows user to choose messages having to appear in the interactive generator.
Message index	Indicate message index allowing user to make the correspondence with signals.
Delete	Allows user to delete a message and all associated signals
Sending a message	Sending this message.
Selection of signals	Allows user to choose signals having to appear in the interactive generator.

3.9.2.2 Messages

The list of the signals makes it possible to modify during acquisition:

- signal value,
- the step for increase and decrease the signal,
- To increase or to decrease the signal value with buttons or keys.



Indicate the index of the message to which the signal belongs.

Delete

Remove this signal of the interactive generator.

3.10 Replay asc files

MuxTrace allows user to replay messages contained in an ASC file.

3.10.1 Configuration of replay functionality

Once an asc file is associated with at least one bus, the replay functionality is activated. It is however necessary to configure the general.

Project name	Cyclic update (ms)	100
Board USB_M	PX_DIAG (2 CAN, 4 ISO9141) board By Memory size (frames)	1024
CAN CAN 1 CAN 2 NWC 1 NWC 2	 Record file replay ✓ Replay using a emulated PCI device ✓ Suspend the replay when finds trigger blocks in files ✓ Replay the record file in loop Start condition 	Delay (ms)
🔤 🤐 LIN 1	Immediate	• 0
- 🧩 LIN 2	Replay schedule	Period (ms)
LIN 3	Automatic (Use the timestamps found in the files)	▼ 0
4 J1587 4 NMEA0183 4 VAN		
- 45 J1587 - 45 NMEA0183 - 45 VAN - 22 Record file replay - 50 Ganals		
- 45 J1587 - 45 NMEA0183 - 45 VAN - 22 Record file replay - 5 Graph		
J1587 MEA0183 VAN Record file replay Signals Graph Digital input		

3.10.1.1 General options

The general options allow:

- To use the demonstration mode in order to not emit physically on a real bus, thus avoiding disturbing the network.
- To suspend the replay functionality when a trigger is found in the asc files.
- Repetitive output sending mode

3.10.1.2 Start conditions

- Immediately:

The first message is transmitted at the start of measurement.

With the first event:

The replay starts according to the dating of the first message present in the file. Thus, if the first message is dated at 15 seconds, the replay will begin 15 seconds after the startup of acquisition.

- After timeout: The first message will be transmitted after a time chosen by the user.

With an user event:

The first message will be transmitted when the user presses on the button of startup of replay.

3.10.1.3 To schedule replay

This functionality allows user to change periodicity of the messages found in the asc files with several manners:

- Automatically

The messages are transmitted according to the dating defined in the asc file.

- According a period of x ms between each frame

All the messages will be transmitted according to the same period chosen by user.

- Frame by frame

The messages are transmitted one by one. The user chose using a button when the next message will be transmitted.

- With an user program

He messages will be transmitted according to their respective dating, but the associated program will be able to choose the messages being able to be emitted and when the replay must be suspended.

3.10.2 Tool bar for the replay



3.10.2.1 State

Disabled	No asc file is associated to the replay functionality.
Enabled	<u>Fixed picture</u> : The replay is activated but acquisition is not started yet. <u>Animate picture</u> : The replay is started.
Stopped	The replay is finished.

3.10.2.2 Commandes

Start/Resume Starts the relay or resumes if this one were suspended.

Suspend

99

Suspends the replay.

3.11 Display of signals

When data bases are associated to the buses, it is possible to visualize the value of the codified signals in these bases.

3.11.1 Creating a list of signals



The signals are ordered by group. This first selection allows the user to organize the signals according to their environment. For example, create the group *Engine* where the signals *Engine Speed, Water Temperature, …* are classified. Create the group *Comfort* where the signals *Distance Covered, Vehicle Speed, …* are classified.

3.11.2 Visualizing the signals

When the program is executed, a window opens to display the signals.



Cyclic update

The logo on the left of the value of the signal changes every time a new value of the signal is displayed. If the logo does not change, it means that the frame containing the signal information is not being received.

3.12 Graph display of signals

When the data bases are associated to the buses, it is possible to visualize graphically the evolution of the value of the signals codified in these bases.

3.12.1 Creating a list of signals



It is possible to visualize up to 16 signals simultaneously. The user can also customize the scale used during visualization. By default, the scale is that stated in the data base.

3.12.2 Visualizing the signals

When the program is executed, a graph window opens to display the signals.



Cyclic update	The	update	frequency	of	the	graph	is	defined	in	the
	proje	et parar	neters.							
Auto V	Time		in alla							

- Axis X : Time in seconds
- Axis Y: Black : Minimum and maximum scale of all signals (double click on the legend) Green, yellow, red or blue : scale that adapts to the signal (one click on the legend)

3.13 Digital Inputs

MuxTrace allows the surveillance of digital inputs present on all network access boards of the range of EXXOTEST products.

Project name		Cyclic upda	ate (ms)	100
Board	Demo version	Memory siz	e (frames)	1024
NWC1	Digital input			
NWC 3	Digital input n°1	🔽 Rising edge	🔽 Falling edge	
🕂 🔆 NWC 4	Digital input n°2	🗖 Rising edge	🧧 Falling edge	
	Digital input n°3	🔲 Rising edge	🔲 Falling edge	
LIN 2	Digital input n°4	🥅 Rising edge	🥅 Falling edge	

Rising edge Only those inputs with a rising edge will be displayed in the visualization windows.

Falling edgeOnly those inputs with a falling edge will be displayed in
the visualization windows.

When a rising or falling edge has been detected, all visualization windows receive a list of digital inputs whose status has changed.

3.14 Digital Outputs

3.14.1 Output triggering

MuxTrace allows the user to activate a digital output present on all network access boards of the range of EXXOTEST products. The aim of this triggering (positive impulse of a few microseconds) is to synchronize an external tool with an event that takes place in the network.

Project name			Cyclic update (ms)	100
oard	Demo version	I)	Memory size (frames)	1024
NWC 1	Output trigger			
NWC 2	Digital output n*1	Active	Setup	
- 🎉 NWC 4	Digital output n*2	F Active	Setup	
LIN 5	Digital output n°3	📕 Active	Setup	
	Digital output n°4	F Active	Setup	
J1587 J158 J158 J158 J15 J158 J158 J158 J158 J158 J158 J15 J158 J15 J15				

Active

An impulse is generated in the output when the configured event is detected (see material's installation guide to connect the output)

3.14.2 Configuring the triggering condition

Name Bustune	[CAN	
Bus number	1	
CAN trigger se Trigger type	etup Error frame	
dentifier	0x 000	Extended (29 bits)

Frame mode Selection of network type

Bus Number of bus on which the event is detected

Type of event CAN identifier

Example

Identifier 1 is chosen as a triggering condition



3.15 Analog inputs (ANA)

MuxTrace allows the surveillance of analog inputs on all network access boards of the range of EXXOTEST products. The value of these inputs can be correlated with the numerical information transiting the networks.

oject name		Cyclic	c update (ms)		100
pard	Demo version	Mem	ory size (frames)		1024
	ANA input				
- X NWC 2 - X NWC 3	Analog input n°0 to n°3	Active	Setup	20 ms	
NWC 4	Analog input n°4 to n°7	C Active	Setup	20 ms	
∃-,7 LIN I,2 LIN 1	Analog input n°8 to n°11	C Active	Setup	20 ms	
🛁 🕹 LIN 2	Analog input n°12 to n°15	C Active	Setup	20 ms	
∃- 🞜 ISO 9141	Analog input n°16 to n°19	C Active	Setup	20 ms	
	Analog input n°20 to n°23	C Active	Setup	20 ms	
⊡ 🞜 J1587 	Analog input n°24 to n°27	Active	Setup	20 ms	
	Analog input n°28 to n°31	C Active	Setup	20 ms	
MMEAU183 MMEAU183 MMEAU183 MMEAU183 VAN 1 VAN 2 VAN 2 VAN 3 Graph Graph Digital input Trigger output					

Input ANA n °x to n °y

Select the list of analog inputs to go back to in the trace window.

The configuration parameters are an identifier and a periodicity. Thanks to their information, analog data are received in a way similar to a network message. It is necessary to use a data base according to the identifiers chosen (see example of ADC.DBC file)

3.16 Programming module

The module of programming allows the user to create his own program inside the MuxTrace environment. This program (DLL : dynamic library link), will permit to the user to personalize the MuxTrace with for example:

- create a sending scenario
- display user's text and values into the edit window
- start a recording on a particular condition
- to schedule asc file replay
- ...

nformations			
Project name		Cyclic update (ms)	100
Board USB_	MPX_DIAG (2 CAN, 4 ISO914	1) board BD Memory size (frames)	1024
CAN 2	Programming environmen	it	
- 🕼 LIN	File name	C:\Program Files\MUXTrace\SDK\Sample\DevCF	PP5\DevCPP5.dll 🛛 🚅
LIN 2 LIN 3 LIN 4 LIN 4 LIN 4	Version v1.00	SDK Display the v1.01 en Recommened SDK V	/ersion v1.01
150 T	Name		
ISO 3 ISO 4 ISO 7 ISO 7 ISO 7 ISO 7 ISO 7 ISO 7 ISO 7 ISO 7 ISO 7 ISO 4 ISO 4 ISO 4 ISO 4 ISO 4 ISO 4 ISO 4 ISO 7 ISO 7	GetInfos GetSDKVersion OnPreStart OnSend OnStart OnStop	Sample for the muxtrace software, showing e Compiled with SDK v1.01.	ach available
Record file replay	Variables		
Signals	Name	Value	1.
- Digital input	0 wNbOfFramesBefor	eSuspendR 10	
Trigger output	1		
ANA input	2		
rogramm			

3.16.1 Starter Development Kit

The MuxTrace repertory contains a sub repertory with the SDK for develop a MuxTrace DLL.

🛅 MUXTrace 🔹 🕨	(m)	Outils 🕨	~	Editeur d'analyse graphique
×	1	Désinstaller MUXTrace v4.38	Ê	Editeur de bases de données
	.	MUXTrace v4.38	٩	Programmation Muxtrace (dossier)

The SDK contains the following files and repertories:

Sample	Repertory containing samples of MuxTrace DLL.		
Prgmux.cpp	File containing the definition of the functions and the reserved entrance points. Do not modify.		
Prgmux.h	File containing the declarations of the functions and the entrance points reserved. Do not modify.		
PrgMux_Skeleton.cpp	File containing the skeleton and the list of the points entered user. To recopy before modifying.		

3.16.2 Library entry points

Muxtrace calls the DLL at various entrance points, these calls can be synchronized or not:

- If debug mode is checked off, the entrance points are synchronized to the display functions of MuxTrace. The performances are lesser.
- If debug mode is not checked off, the entrance points are not synchronized. The performances are optimized.

User's List of entrance points

3.16.2.1 GetInfos : DLL Information

<u>Prototype</u> : int GetInfos(char *szInfos, int *iVersion)

<u>Parameters</u>: szInfos = Character strings receiving information relating to the DLL. iVersion = DLL version : BCD coding (ex : 0x120 = v1.20).

<u>Description</u> Muxtrace calls this function when the DLL is loaded, then posts this information in the window of the programming module.

3.16.2.2 OnStart : Starting of acquisition

<u>Prototype</u> : int OnStart(void)

Description : Muxtrace calls this function when the user starts acquisition

3.16.2.3 OnStop : Stop of acquisition

<u>Prototype</u> : int OnStop(void)

Description : MuxTrace calls this function when the user stops acquisition

3.16.2.4 OnGetVariableName : Definitions of the user variables

<u>Prototype</u> : int OnGetVariableName(unsigned short wIndex, char *szVariable, char *szDefaultValue)

<u>Parameters</u>: wIndex = index of the variable to be defined. From 0 to 15. szVariable = Variable name szDefaultValue = Variable value.

<u>Codes retour</u> : Return value : 0 if succeed.

<u>Description</u> : Muxtrace calls this function when the DLL is loaded, then displays the list of variables in the window of programing module. This function is called as much as it returns a value different from 0.

3.16.2.5 OnSetVariableValue : Initialization of the user variables

<u>Prototype</u> : int OnSetVariableValue(const unsigned short wIndex, const char *szVariable, const char *szValue)

<u>Parameters</u>: wIndex = Variable index. From 0 to 15. szVariable = Variable name. szValue = Value to be assigned to this variable.

Description : MuxTrace calls this function at the time of the initialization of acquisition.

3.16.2.6 OnKey : When the user press a key

<u>Prototype</u> : int OnKey(int IKey)

parameter : IKey = Key code (symbol VK_xxx)

<u>Description</u> : MuxTrace calls this function when the user presses a key

3.16.2.7 OnTimer : every milliseconde

<u>Prototype</u> : OnTimer(DWORD dwTimer)

<u>parameter</u>: dwtimer = Time stamp in ms since the starting of acquisition

Description : MuxTrace calls periodically this function : every millisecond

3.16.2.8 OnCanEvent: On a reception of a CAN event

<u>Prototype</u> : OnCanEvent (tCanEvent *hCanEvent)

<u>parameter</u>: hCanEvent = Pointer on a structure containing the type of CAN event (see file REFMUX.H)

<u>Description</u> : MuxTrace calls this function when a CAN event happens. This event corresponds to :

- On receipt of a message
- On End of message transmission
- On Error detection

3.16.2.9 OnVanEvent: On a reception of a VAN event

Prototype : OnVanEvent (tVanEvent *hVanEvent)

<u>parameter</u>: hVanEvent = Pointer on a structure containing the type of VAN event (see file REFMUX.H)

<u>Description</u> : MuxTrace calls this function when a CAN event happens. This event corresponds to :

- On receipt of a message
- On End of message transmission
- On Error detection
- On End of transmission in error

3.16.2.10 OnLinEvent: On a reception of a LIN event

Prototype : OnLinEvent (tLinEvent *hLinEvent)

<u>parameter</u>: hLinEvent = Pointer on a structure containing the type of LIN event (see file REFMUX.H)

<u>Description</u> : MuxTrace calls this function when a LIN event happens. This event corresponds to :

- On receipt of a message
- On End of message transmission
- On Error detection
- On End of transmission in error

3.16.2.11 OnlsoEvent: On a reception of a ISO9141 event

Prototype : OnlsoEvent (tlsoEvent *hlsoEvent)

<u>parameter</u>: hlsoEvent = Pointer on a structure containing the type of ISO event (see file REFMUX.H)

<u>Description</u> : MuxTrace calls this function when an ISO event happens. This event corresponds to :

- On receipt of a message
- On End of message transmission
- On Error detection
- On End of transmission in error

3.16.2.12 OnReplayFrame : On a replay frame

<u>Prototype</u> : int OnReplayFrame(ST_PRGMUX_ONREPLAYFRAME *hOnReplay, unsigned short &wReplayThisFrame, unsigned short &wSuspendReplay)

<u>Parameters</u>: hOnReplay = Replay message (see file PRGMUX.H) wReplayThisFrame = if value = 0 this message will not be send. wSuspendReplay = if value = 0 replay will be suspended. <u>Description</u> : Muxtrace calls this function just before to send a replay message. It is thus possible to schedule the replay and to filter messages. You must activated "by program" in the replay configuration window.

3.16.2.13 OnTriggerReplay : Trigger present in a asc file

<u>Prototype</u> : int OnTriggerReplay(ST_PRGMUX_ONTRIGGERREPLAY *hOnTrigger, unsigned short &wSuspendReplay)

<u>Arguments</u> : hOnTrigger = Trigger definition (see file PRGMUX.H) wSuspendReplay = if value = 0 replay is suspended.

<u>Description</u> : Muxtrace calls this function when a trigger event is read from asc file. You must activated "by program" in the replay configuration window.

Reserved List of entrance points

3.16.2.14 GetSDKVersion : SDK Version

<u>Prototype</u> : int GetSDKVersion(void)

<u>Codes retour</u> : Return SDK version used for create the DLL. BCD coding (ex : 0x230 = v2.30).

<u>Description</u> : Calls this function when the DLL is loaded, thus allowing MuxTrace to compare its kit of development with that used by the DLL, in order to prevent possible conflicts being able to occur when that the kit used by the DLL is more recent than that of MuxTrace.

3.16.2.15 OnPreStart: Initialization before starting of acquisition

Prototype : OnPreStart (void)

<u>Description</u> : MuxTrace calls this function on start of acquisition but before starting of the communication with the networks. This event is useful to carry out initializations of shared information between MuxTrace and the DLL.

3.16.2.16 OnSend: management of the queue between the DLL and MuxTrace

Prototype : OnSend(tExxoFifoMsg *hCurExxoFifoMsg)

<u>Parameter</u>: hCurExxoFifoMsg = Pointer on a structure containing the actions to be transmitted towards MuxTrace

<u>Description</u> : MuxTrace calls this function periodically to detect the actions to be carried out. All the actions (sending, display, trigger...) are stored in a queue, which is unqueued at each call of this function.

3.16.3 Accessible functions since the library

3.16.3.1 DisplayMsg: Printing a text in the window of edition

<u>Prototype</u> : DisplayMsg (char *szText)

<u>Parameter</u> : szText = String to display (max : 1024 bytes)

Return: STATUS_OK if succeed Other value if failed

<u>Description</u> : This function allows to print the text in the window of edition.

3.16.3.2 CanSendMsg: Sending a CAN message

<u>Prototype</u> : tMuxStatus CanSendMsg(unsigned short wCard, unsigned short wBus, tCanMsg *hCanMsg)

Parameter: see file REFMUX.H (document DLL-MUX-CAN)

Return: STATUS_OK if succeed Other value If failed

<u>Description</u> : This function allows to send a data message or a request of distant transmission.

3.16.3.3 LinSendMsg: Sending LIN message

<u>Prototype</u> : tMuxStatus LinSendMsg(unsigned short wCard, unsigned short wBus, tLinMsg *hLinMsg)

Parameter: see file REFMUX.H (document DLL-MUX-LIN)

<u>Return</u>: STATUS_OK if succeed Other value if failed

<u>Description</u> : This function allows to transmit a message on the bus CAN.

3.16.3.4 IsoSendMsg: Sending a ISO9141 message

<u>Prototype</u> : tMuxStatus IsoSendMsg(unsigned short wCard, unsigned short wBus, tIsoMsg *hIsoMsg)

Parameter: see file REFMUX.H (document DLL-MUX-ISO)

<u>Return</u> : STATUS_OK if succeed Other value if failed

Description : This function allows to transmit a message on the bus ISO9141.

3.16.3.5 Iso14230SendMsg : Sending a ISO14230 message

<u>Prototype</u> : tMuxStatus lso14230SendMsg(unsigned short wCard, unsigned short wBus, tlso14230Msg *hlso14230Msg)

Parameters: see file REFMUX.H (document DLL-MUX-ISO)

<u>Codes retour</u> : STATUS_OK if succeed Other value if failed

Description : This function allows to transmit a message on the bus ISO14230.

3.16.3.6 VanSendMsg: Emission d'un message sur le bus VAN

<u>Prototype</u> : tMuxStatus VanSendMsg(unsigned short wCard, unsigned short wBus, tVanMsg *hVanMsg)

Parameter: See file REFMUX.H (document DLL-MUX-VAN)

<u>Return</u>: STATUS_OK if succeed Other value if failed <u>Description</u> : This function allows to transmit a message on the bus VAN.

3.16.3.7 IOSetOutput: Activation of logic outputs

<u>Prototype</u> : tMuxStatus IOSetOutput(unsigned short wCard, unsigned short wOutputValue, unsigned short wOutputMask)

Parameter: See file REFMUX.H (document DLL-MUX-CAN)

Return: STATUS_OK if succeed Other value if failed

Description : This function allows the user to activate the logic outputs on the card.

3.16.3.8 LinSetSleepMode : Send a master request frame for force all slaves into sleep mode

<u>Prototype</u> : tMuxStatus LinSetSleepMode(unsigned short wCard, unsigned short wBus)

Parameters: See file REFMUX.H (document DLL-MUX-CAN)

<u>Codes retour</u> : STATUS_OK if succeed Other value if failed

<u>Description</u> : Send a master request frame for force all slaves into sleep mode.

3.16.3.9 LinSetWakeUpMode : Request a wake up

<u>Prototype</u> : tMuxStatus LinSetWakeUpMode(unsigned short wCard, unsigned short wBus)

Parameters: See file REFMUX.H (document DLL-MUX-LIN)

<u>Codes retour</u> : STATUS_OK if succeed Other value if failed.

<u>Description</u> : This function allows a slave to request a wake-up.

3.16.3.10 Trigger : trigger for file recording

<u>Prototype</u> : tMuxStatus Trigger(void)

Parameter: None

<u>Return</u>: STATUS_OK if succeed Other value if failed

<u>Description</u> : This function allows to start a recording when the window "logging file setting" is set with the trigger condition "On trigger programm".

3.16.3.11 Stop : stop acquisition

Prototype : tMuxStatus Stop(void)

Parameter: None

Return: STATUS_OK if succeed Other value if failed

Description : This function stops the acquisition.

3.16.3.12 OpenProject : Open a MuxTrace project file

<u>Prototype</u> : tMuxStatus OpenProject(char *szProjectPath, unsigned short wAutoRun, unsigned short wNoSave)

<u>Parameter</u>: szProjectPath = Project path. wAutoRun = if value = true : start automatically muxtrace acquisition. wNoSave = if value = true : don't save the current project.

<u>Codes retour</u> : STATUS_OK if succeed Other value if failed

<u>Description</u> : Stop and close the current project to open another project and to start again acquisition automatically.

3.16.3.13 SuspendReplay : Suspend replay

<u>Prototype</u> : tMuxStatus SuspendReplay(void)

<u>Codes retour</u> : STATUS_OK if succeed Other value if failed

<u>Description</u> : Suspend replay. Call *ResumeReplay function to resume replay*.

3.16.3.14 ResumeReplay : Resume replay

<u>Prototype</u> : tMuxStatus ResumeReplay(void)

<u>Codes retour</u> : STATUS_OK if succeed Other value if failed

<u>Description</u> : Resume a suspended replay.

3.17 Access mode to the cards

MuxTrace makes it possible to work according to two access modes:

3.17.1 Exclusive mode

The exclusive access mode makes it possible to call directly the charts and to prevent other applications from using the chart in the course of use, thus making it possible MuxTrace to obtain optimal performances.

3.17.2 Shared mode

The shared mode allows several applications to work with a same card. In this mode, you can spy, with Muxtrace, others applications working with the same card.

MuxServer setup is available on the MuxTrace CDROM.

3.18 Expert mode

MuxTrace has an expert mode with which the following advanced functions can be used :

- managing data bases,
- display of signals present on the data bases,
- surveillance of digital inputs, triggering of outputs
- logging to a text file
- Managing the communication layer DIAG ON CAN (ISO 15765-2).

The expert mode can be activated in two different ways, Single license or Multiple license.

3.18.1 Expert mode Single license

Expert mode Single license activates all advanced functions for one computer only. It works with all network access boards of the range of EXXOTEST products.

Expert mode Single licence Multiple licence	3
Licence	
Register key	7EEAE39A
Unlook kou	

The expert mode Single license is protected by an authorization key. Contact your retailer to obtain it.

3.18.2 Expert mode Multiple license

The expert mode Multiple license activates all advanced functions for any computer providing the network access board being used has the Multiple license option.

Expert mode	×
Expert mode C Single licence I Multiple licence	
Licence	
Register key	7EEAE39A
Unlock key	
_	<u>O</u> k <u>X</u> Cancel

3.19 Execution

When the programme is executed, a window opens for each different network visualized. After starting the programme, the communication board is active on the network and it can acknowledge and receive the messages coming from the networks.

Echer Londguration Acquisition Fereitres Uptions Aide Signature Image: Control of the second seco
Signalize Image: Constraint of the second secon
Visualisation parameters Image: Content of the second
Temp_huile_mot (Degrt C) ~ Temp_ Information
Temp_ Information
Tomp_
Confort
KM_TOTAL (KM)
VITV (VITM (VITM (Status VITM (Nb Erreurs 0 Contrôleur 1 Nominal
IN VAN 2 IN X IN CAN 1 IN
Exporter
Heure Ident Lg Données Heure Ident Lg Données
Charge bus 0 % Contrôleur 1 Nominal Charge bus 11 % Contrôleur 1 Passif 0 Nb Erreurs 0 Contrôleur 2 Nominal Nb Erreurs 2608 Cpt internes TxErr : 128 RxErr : 0 0

3.19.1 Visualization parameters

Status Correct operation of the display. Correct operation of the display but PC finds it difficult to deal in time with all the messages coming from the network. Loss of messages coming from the network. The PC's capacity to display events is too lower. Increase refreshes frequency or decrease memory size. Display Display in fixed position :each identifier has a display line mode Sequential display : unroll messages on display Exporting Logs displayed messages on a text file for later use (printing, EXCEL type table ...) LIN only : Sending a wake-up call Sending a stand-by message -

3.19.2 Information window

The information window displays in real time the messages transiting the network.

Dir k	End of message transmission Reception of a message End of transmission with error Reception with error Transmission in degraded mode Reception in degraded mode		
Clock	Time when message has been received (absolute, after execution has started)		
Ident	Value of the network identifier in hexadecimal (CAN, VAN, LIN) or address of source -> address of target for ISO9141		
Lg	Length of message data		
Data	Content of data in hexadecimal		
Period	Display in fixed position: time difference in ms, in relation to the last identical identifier displayed. Display in sequential position : time difference in ms, in relation to the previous message (no matter the value of the identifier is).		
Svc	 Message service For the CAN bus: DA : Frame of data DR : Frame of remote transmission (RTR=1) For the NWC bus: DA : Frame of data For the LIN bus: DA : Frame of data or transmission of a writing frame RR : Transmission of a reading request IFR : Transmission of an in frame response WK : Reception of a wake-up call For the VAN bus: DA : Frame of data DAA : Frame of data DA : Frame of data DAA : Frame of data DAA : Frame of data with acknowledgement DRA : Differed response with acknowledgement or reception of a demand for a response in the frame with response. RRA : Transmission of a demand for a response in the frame IFR : Transmission of a response in the frame For the ISO9141 bus, interpretation of command code or response interpretation. For example : STCOM : Start communication STCOM : Start communication 		

• STCOMPR : Positive response to the start communication

request

• NAK ServiceNotSupported : Negative response

Sender Name of transmitting ECU in the frame

3.19.3 Single click of the mouse

In pause or stop mode, one click of the mouse on a message allows the user to set all windows of different networks at the same time as the event on which the click has been made. This allows the user to visualize the status of the data of other networks at that moment.

3.19.4 Double click of the mouse

If a data base is associated, a double click on the message allows the user to visualize all the data carried by this message in real time (in fixed display position only) or while the trace is seen in pause mode or when it stops.

3.19.5 Drag&Drop

Muxtrace allows user to drag signals from the trace window to the data window or to the graphical window.

3.19.5 Sorting out messages

With display in fixed position, the selection of the top part of the column, allows the user to sort out the identifying columns, the period and the senders of the message.

3.19.6 Status

Charge Occupation charge of the network. The charge is calculated every second Counter of number of errors observed after execution has started No errors **Controller 1** For the CAN bus: Controller status : ACTIVE, PASSIVE or BUS OFF _ For the LIN bus: Communication status : NOMINAL, DEGRADED or BUS IDLE -For the VAN network: Communication status of controller 1 : Nominal, communication in data, communication in datab, serious error. Controller 2 For VAN network only : Communication status of controller 2 : Nominal, communication in data, communication in datab, serious error. For CAN and LIN networks : Internal cntrs Value of internal counters of protocol controller, which manages the status of the controller (ACTIVE, PASSIVE and BUS OFF)

Comm. For bus CAN low speed – fault tolerant : - Bus status NOMINAL or DEGRADED

List of versions

Version	Date	Author	Modifications
01	22/06/2001	P. CHAZOT	Creating a document
02	05/10/2001	P. CHAZOT	Adding a LIN network
03	14/11/2001	P. CHAZOT	Adding a LIN identifier
04	05/09/2002	P. CHAZOT	Adding ISO9141 functions
05	30/01/2002	A.GAMBIER	Adding data bases, signals, digital inputs, Diag On Can communication layers (<i>ISO 15765-2</i>)
06	15/05/2003	P. CHAZOT	Adding logging on a text file, output triggering, re-starting after bus off, graph of signals
07	06/11/2003	P. CHAZOT	Adding analog inputs Adding decoding of a message in real time
08	09/12/2003	P.CHAZOT	Adding transmitting ECUs
09	08/06/2004	P.CHAZOT	MuxTrace V 4.30 - Adding filters (CAN, VAN) - Adding file recording functions - Adding programming functions
10	10/05/2005	A.GAMBIER	 MuxTrace v4.40 Adding bite rate detection Adding interactive generator Adding replay module Adding new functionalities to the programming module