

BUSMASTER – FlexRay Add-On

BUSMASTER © - FlexRay Overview

❖ Introduction

BUSMASTER Add-On for FlexRay can be used to monitor, analyse and simulate FlexRay bus

❖ Features Supported

- Channel Configuration
- Controller Configuration
- Transmission of Messages
- Monitoring Messages
- Network Statistics
- Filters
- Logging
- Signal watch
- Node Simulation

❖ Installation

FlexRay support in BusMaster is available as a Commercial Add-On

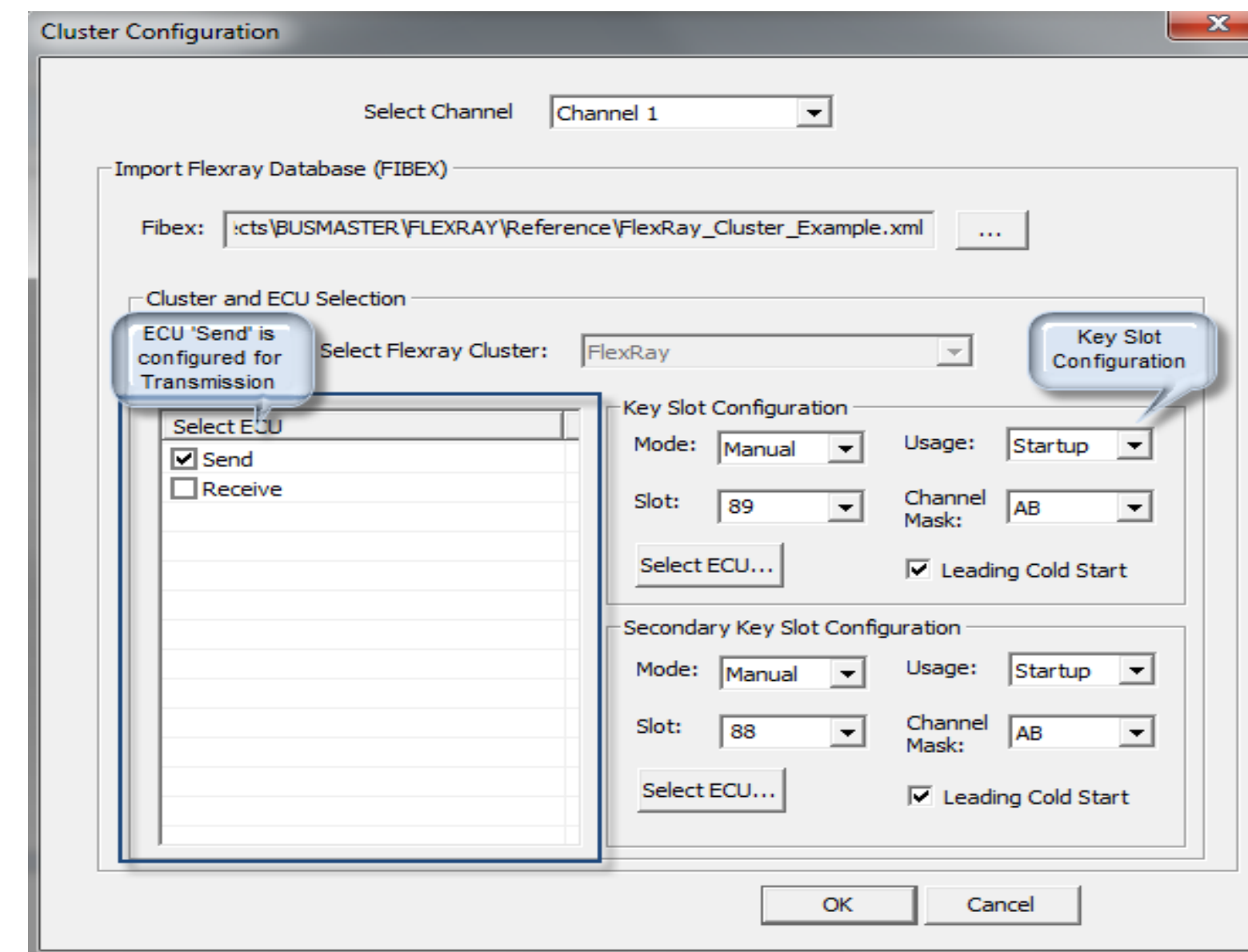
FlexRay Add-On v1.1.0 works on top of open source v3.2.1

For further details, please contact E-mail: BUSMASTER@in.bosch.com

BUSMASTER © - FlexRay Features

❖ Channel Configuration

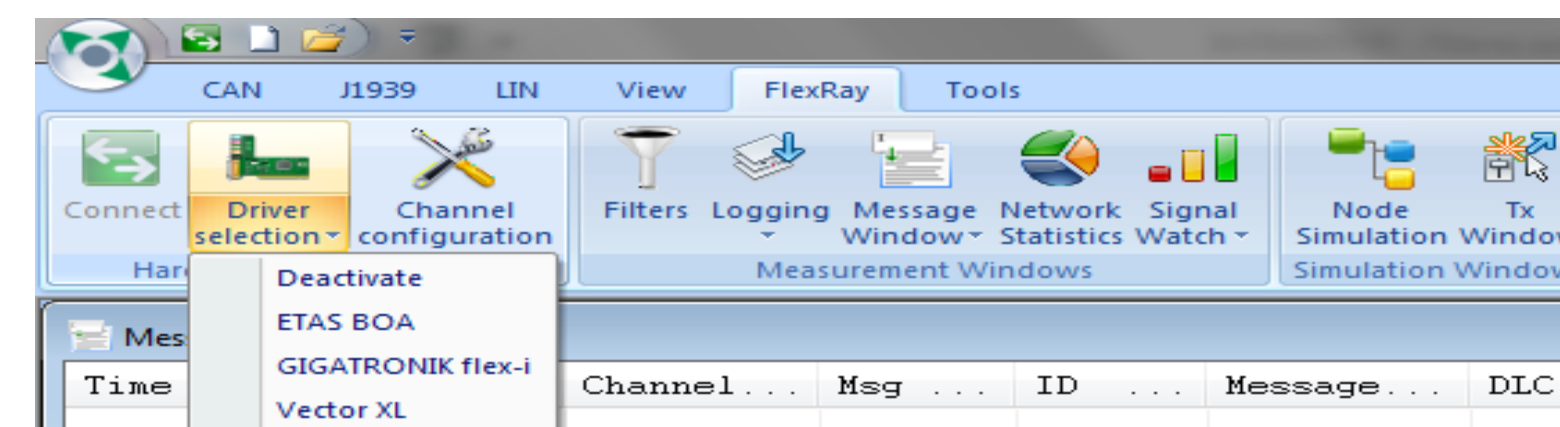
Cluster configuration is used to associate FIBEX file to a FlexRay channel



❖ Controller Configuration

BUSMASTER can be connected to FlexRay physical channel using FlexRay Controller

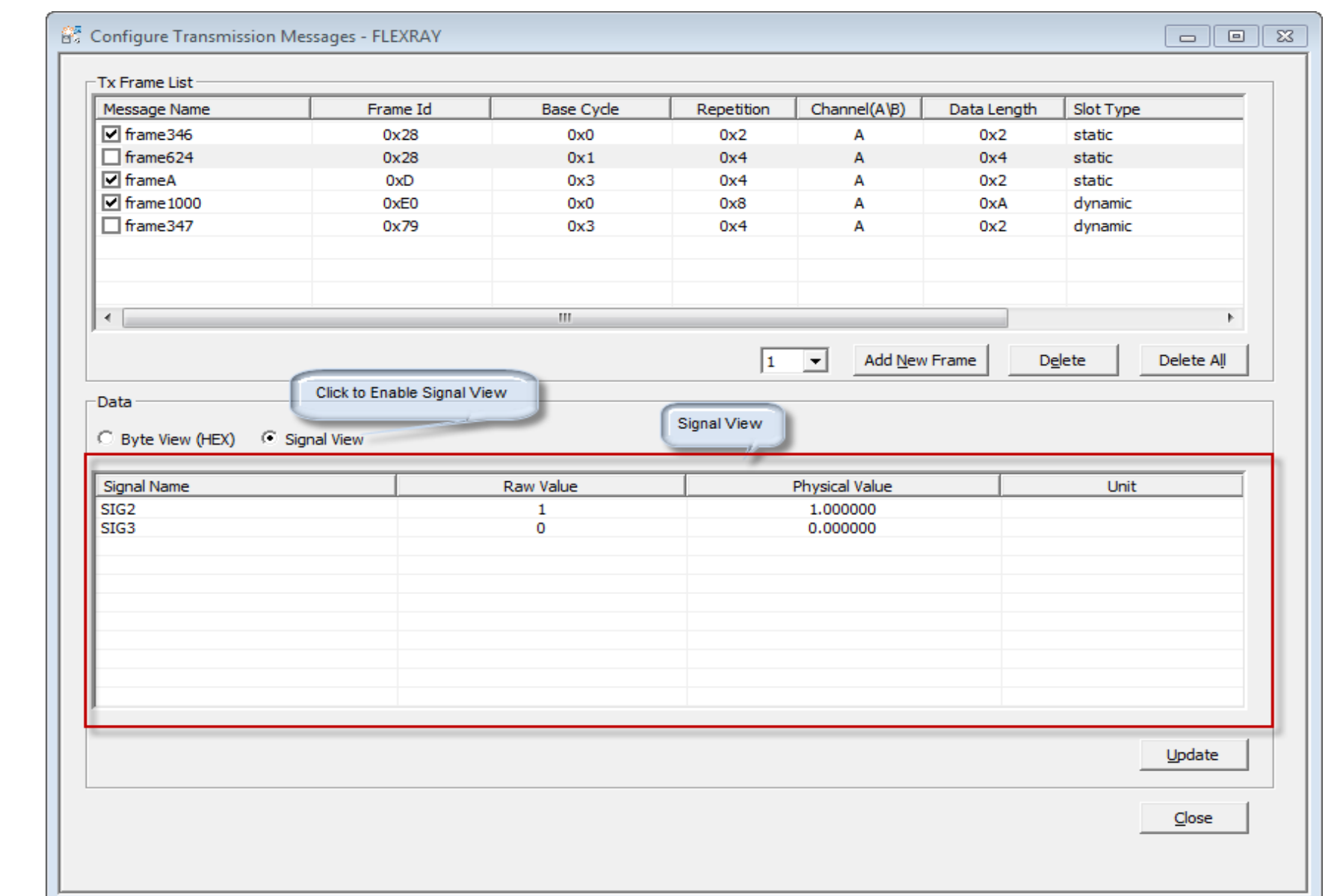
- Following FlexRay controllers are supported:
 1. ETAS BOA
 2. Vector XL



BUSMASTER © - FlexRay Features

❖ Transmission of Messages

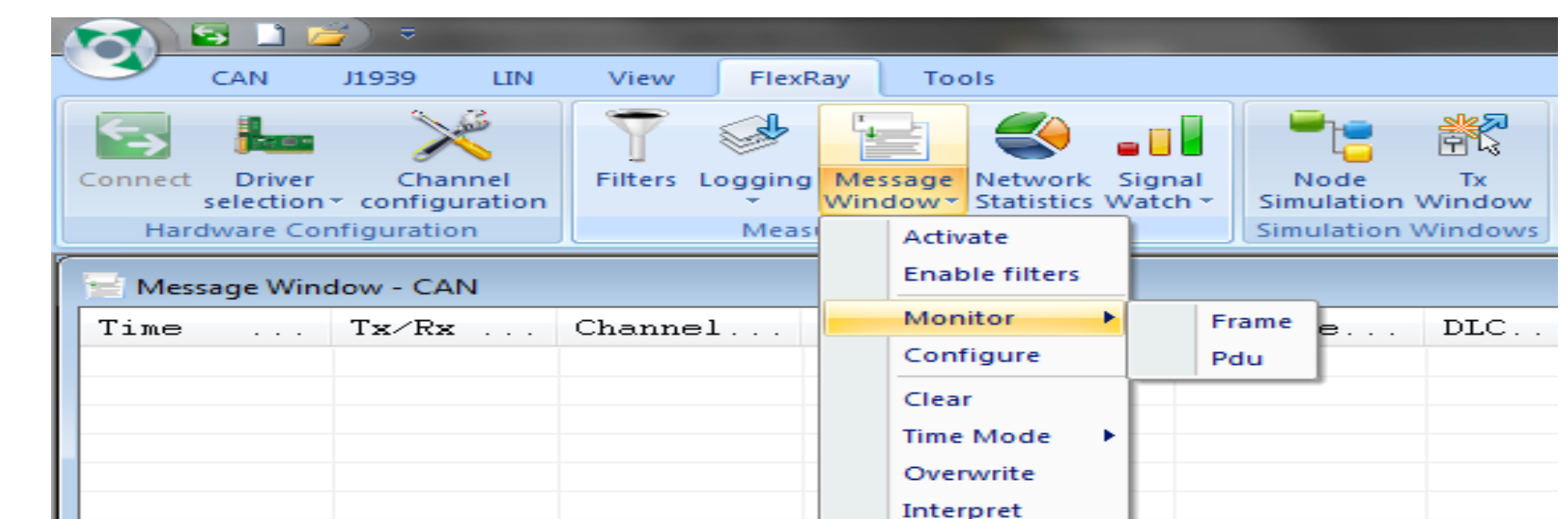
Configure FlexRay messages for transmission



❖ Monitoring Messages

On-line FlexRay Network Monitoring can be done using FlexRay Message window

- Frame Monitoring
- PDU Monitoring
- Frame and PDU Monitoring



BUSMASTER © - FlexRay Features

❖ Frame Monitoring

Message Window - FLEXRAY											
	Time	Tx...	Id	Type	Frame Ty...	Slot T...	Description	Cycle...	Cha...	D...	Data Byte(s)
+	15:50:28:2907	Tx	0001 [0, 1]	Frame	Data	static	New_Frame_1	51	A	16	000 000 000 0
+	15:50:28:2907	Tx	0002 [0, 1]	Frame	Data	static	New_Frame_2	51	A	16	000 000 000 0
	15:50:28:2907	Tx	0003 [0, 1]	Frame	Data	static	New_Frame_3	51	A	16	000 000 000 0
+	15:50:28:2907	Tx	0004 [0, 1]	Frame	Data	static	New_Frame_4	51	A	16	000 000 000 0
	15:50:28:2907	Tx	0005 [0, 1]	Frame	Data	static	New_Frame_5	51	A	16	000 000 000 0
	15:50:28:2907	Tx	0006 [0, 1]	Frame	Data	static	New_Frame_6	51	A	16	000 000 000 0
	15:50:28:2907	Tx	0007 [0, 1]	Frame	Data	static	New_Frame_7	51	A	16	000 000 000 0

❖ PDU Monitoring

Message Window - FLEXRAY											
	Time	Tx...	Id	Type	Frame Ty...	Slot T...	Description	Cycle...	Cha...	D...	Data B...
+	15:49:40:5771	Tx	0001	PDU			New_PDU_2	18	A	1	000
	15:49:40:5771	Tx	0001	PDU			New_PDU_1	18	A	1	000
+	15:49:40:5771	Tx	0002	PDU			New_PDU_3	18	A	1	000
	15:49:40:5771	Tx	0002	PDU			New_PDU_1	18	A	1	000
+	15:49:40:5771	Tx	0003	PDU			New_PDU_2	18	A	1	000
	15:49:40:5771	Tx	0004	PDU			New_PDU_2	18	A	1	000
	15:49:40:5771	Tx	0004	PDU			New_PDU_1	18	A	1	000
	15:49:40:5771	Tx	0004	PDU			New_PDU_5	18	A	1	000
	15:49:40:5771	Tx	0004	PDU			New_PDU_4	18	A	1	000
	15:49:40:5771	Tx	0007	PDU			New_PDU_6	18	A	1	000
	15:49:40:5771	Tx	0007	PDU			New_PDU_7	18	A	1	000
	15:49:40:5771	Tx	0007	PDU			New_PDU_8	18	A	1	000
	15:49:40:5771	Tx	0007	PDU			New_PDU_9	18	A	1	000

❖ Frame and PDU Monitoring

Message Window - FLEXRAY											
	Time	Tx...	Id	Type	Frame Ty...	Slot T...	Description	Cycle...	Cha...	D...	Data Byte...
+	15:43:54:3590	Tx	0001 [0, 1]	Frame	Data	static	New_Frame_1	29	A	16	000 000
	15:43:54:3590	Tx	0001	PDU			New_PDU_2	29	A	1	000
+	15:43:54:3590	Tx	0001	PDU			New_PDU_1	29	A	1	000
+	15:43:54:3590	Tx	0002 [0, 1]	Frame	Data	static	New_Frame_2	29	A	16	000 000
	15:43:54:3590	Tx	0002	PDU			New_PDU_3	29	A	1	000
+	15:43:54:3590	Tx	0002	PDU			New_PDU_1	29	A	1	000
	15:43:54:3590	Tx	0003 [0, 1]	Frame	Data	static	New_Frame_3	29	A	16	000 000
	15:43:54:3590	Tx	0003	PDU			New_PDU_2	29	A	1	000
+	15:43:54:3590	Tx	0004 [0, 1]	Frame	Data	static	New_Frame_4	29	A	16	000 000
	15:43:54:3590	Tx	0004	PDU			New_PDU_2	29	A	1	000
+	15:43:54:3590	Tx	0004	PDU			New_PDU_1	29	A	1	000
	15:43:54:3590	Tx	0004	PDU			New_PDU_5	29	A	1	000
	15:43:54:3590	Tx	0004	PDU			New_PDU_4	29	A	1	000
	15:43:54:3590	Tx	0005 [0, 1]	Frame	Data	static	New_Frame_5	29	A	16	000 000
	15:43:54:3590	Tx	0006 [0, 1]	Frame	Data	static	New_Frame_6	29	A	16	000 000
	15:43:54:3590	Tx	0007 [0, 1]	Frame	Data	static	New_Frame_7	29	A	16	000 000
	15:43:54:3590	Tx	0007	PDU			New_PDU_6	29	A	1	000
	15:43:54:3590	Tx	0007	PDU			New_PDU_7	29	A	1	000
	15:43:54:3590	Tx	0007	PDU			New_PDU_8	29	A	1	000
	15:43:54:3590	Tx	0007	PDU			New_PDU_9	29	A	1	000

BUSMASTER © - FlexRay Features

❖ Network Statistics

Displays FlexRay bus statistics like number of frames, PDU’s transmitted and received, error frames, static segment load etc.,

Network Statistics		
CAN FlexRay LIN		
Parameters	Channel A	Channel B
Frames Total	1113667	0
Frames [fr/s]	4245	0
Null Frames Total	434017	0
Null Frames [fr/s]	1662	0
PDU Total	1305261	0
PDU [pdus/s]	4970	0
Errors Total	0	0
Errors [fr/s]	0	0
Static Segment Load(Data+Null)	49.784314	0.000000
Transmitted		
Frames Total	0	0
Frames [fr/s]	0	0
Null Frames Total	0	0
Null Frames [fr/s]	0	0
PDU Total	0	0
PDU [pdus/s]	0	0
Errors Total		
Errors [fr/s]	0	0
Received		
Frames Total	1113667	0
Frames [fr/s]	4245	0
Null Frames Total	434017	0
Null Frames [fr/s]	1662	0
PDU Total	1305261	0
PDU [pdus/s]	4970	0
Errors Total		
Errors [fr/s]	0	0

❖ Filters

Configure Stop or Pass filters for FlexRay Frame’s/PDU’s

❖ Logging

Configure log files to log FlexRay Frame’s/PDU’s transmitted or received on the FlexRay bus

BUSMASTER © - FlexRay Features

❖ Signal Watch

Analyze selected signal’s Raw and Physical values

Message	Signal	Physical Value	Raw Value
frame_25	signal_25	2312.000000	2312
frame397	SIG1	34952.000000	34952
frame397	SIG10	34952.000000	34952
frameAB	signalA	0.000000	0

❖ Node Simulation

FlexRay Node functionality can be simulated with following handlers

- Bus Events
 - Message Handlers
 - PDU Handlers
 - POC Handlers
- Start Cycle Handlers
 - Timer Handlers
 - Key Handlers
 - DLL Handlers

```
ECU1.cpp
D:\Demo\ECU1.cpp
Include Headers
#include <Windows.h>
#include <FlexRayIncludes.h>
Bus Events
void OnBus_Connect()
Message Handlers
void OnMsgCh1_1_0_1(STFLX_MSG RxMsg)
PDU Handlers
void OnPDUName_New_PDU_1(New_PDU_1 RxPDU)
POC Handlers
void OnPOCState(STFLX_POCSTATUS stcPOCStatus)
Start Cycle Handlers
void OnStartCycle_0_1(STFLX_STARTCYCLE stcStartCycle)
Timer Handlers
void OnTimer_T1_1()
Key Handlers
void OnKey_a(unsigned char KeyValue)
DLL Handlers
void OnDLL_Load()
Utility Functions
Global Variables

37: {
38:  /* TODO */
39: } /* End FlexRay generated function - OnBus_Connect */
40: /* Start FlexRay generated function - OnMsgCh1_1_0_1 */
41: void OnMsgCh1_1_0_1(STFLX_MSG RxMsg)
42: {
43:  /* TODO */
44: } /* End FlexRay generated function - OnMsgCh1_1_0_1 */
45: /* Start FlexRay generated function - OnPDUName_New_PDU_1 */
46: void OnPDUName_New_PDU_1(New_PDU_1 RxPDU)
47: {
48:  /* TODO */
49: } /* End FlexRay generated function - OnPDUName_New_PDU_1 */
50: /* Start FlexRay generated function - OnPOCState */
51: void OnPOCState(STFLX_POCSTATUS stcPOCStatus)
52: {
53:  /* TODO */
54: } /* End FlexRay generated function - OnPOCState */
55: /* Start FlexRay generated function - OnStartCycle_0_1 */
56: void OnStartCycle_0_1(STFLX_STARTCYCLE stcStartCycle)
57: {
58:  /* TODO */
59: } /* End FlexRay generated function - OnStartCycle_0_1 */
60: /* Start FlexRay generated function - OnTimer_T1_1 */
61: void OnTimer_T1_1( )
62: {
63:  /* TODO */
64: } /* End FlexRay generated function - OnTimer_T1_1 */
65: /* Start FlexRay generated function - OnKey_a */
66: void OnKey_a(unsigned char KeyValue)
67: {
68:  /* TODO */
69: } /* End FlexRay generated function - OnKey_a */
70: /* Start FlexRay generated function - OnDLL_Load */
71: void OnDLL_Load()
72: {
73: }
```