PSI5 Sensor Programming

-Seskion GmbH-

Content:

- Preparation for the diagnostic mode
- Settings for the programming mode of the sensor
- Opening the diagnostic mode
- Bidirectional communication

Version:	(1.0) 05.11.2021 - Creation

PSI5 Sensor Programming

Sensors have a diagnostic mode. This mode is defined by the sensor manufacturers and is not standardized. If a sensor is set to diagnostic mode, it provides data from defined addressing ranges of the manufacturer. The diagnostic mode is activated by specific data sequence that the ECU sends to the sensor.

The diagnostic mode of the sensors is a very specific section and should only be used by users with appropriate knowledge.

The Simulyzer diagnostic mode allows:

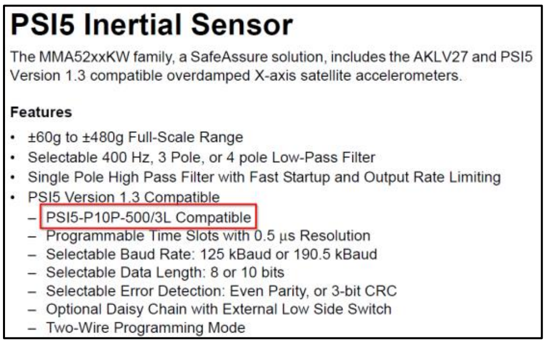
- put a sensor into dignosticmode,
- send commands to the sensor
- read out the memory cells of the sensor,
- overwrite the memory cells.

The following range of functions is available:

- Opening the diagnostic mode
- Bidirectional communication
- Memory data
- Configuration of the diagnostic memory.

Preparation for the diagnostic mode

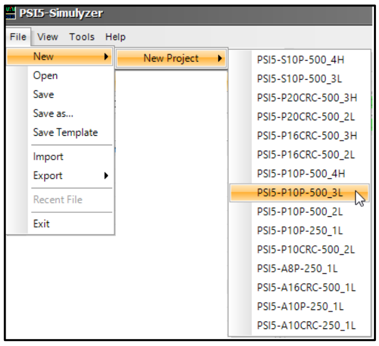
In order to be able to work with the Simulyzer software, a new project must be created at the beginning after connecting the Simulyzer and the sensor. To do this, use the menu group ***File*** and the command ***New*** and ***New Project***. Which version your PSI5 sensor has you can read in the features from the manual of the respective sensor. In this example a NXP sensor of the MMA52xxKW family with a PSI5-P10P-500-3L compatibility is used.


PSI5 Inertial Sensor

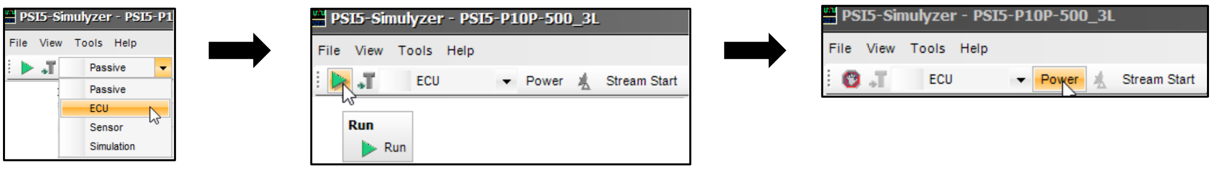
The MMA52xxKW family, a SafeAssure solution, includes the AKLV27 and PSI5 Version 1.3 compatible overdamped X-axis satellite accelerometers.

Features

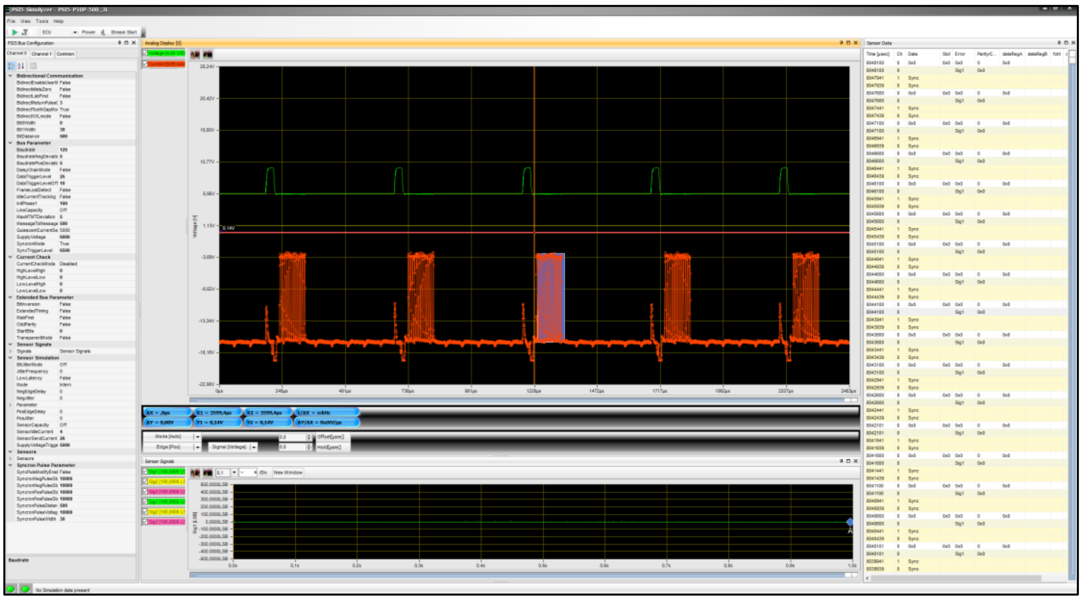
- ±60g to ±480g Full-Scale Range
- Selectable 400 Hz, 3 Pole, or 4 pole Low-Pass Filter
- Single Pole High Pass Filter with Fast Startup and Output Rate Limiting
- PSI5 Version 1.3 Compatible
  - PSI5-P10P-500/3L Compatible
  - Programmable Time Slots with 0.5 µs Resolution
  - Selectable Baud Rate: 125 kBaud or 190.5 kBaud
  - Selectable Data Length: 8 or 10 bits
  - Selectable Error Detection: Even Parity, or 3-bit CRC
  - Optional Daisy Chain with External Low Side Switch
  - Two-Wire Programming Mode



Next, start a measurement by first selecting the ***ECU mode*** and then selecting the ***green arrow*** and ***Power***.

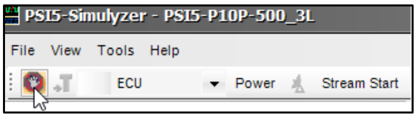


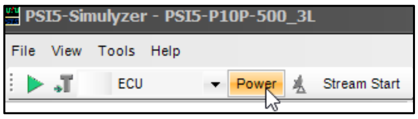
You will now see that on the right side the individual data with time stamp are listed in tabular form. In the middle this is graphically represented.



PSI5 Sensor Programming

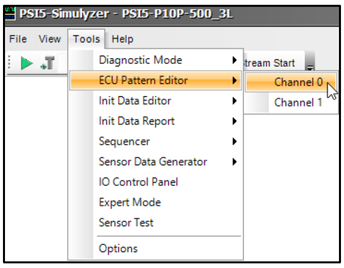
To stop the measurement, press the ***red symbol*** and then the ***power*** button to end the measurement completely.

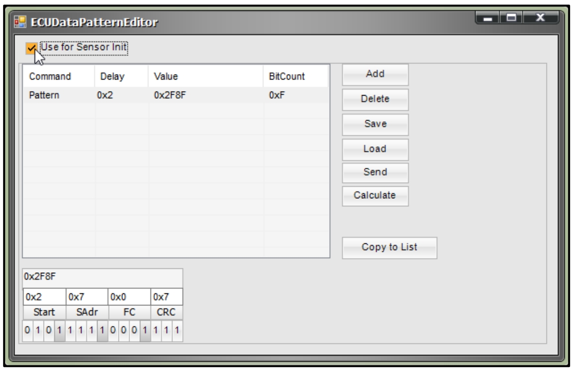




Settings for the programming mode of the sensor

To enter the programming mode of the sensor it is necessary to select the settings correctly. The correct settings for your sensor can be found in the respective data sheet.
To set the correct settings in the program, select in the menu group ***Tools*** the command ***ECU Pattern Editor***. Then select ***Channel 0*** or ***Channel 1*** depending on which channel your sensor is connected to.





In the ECU Pattern Editor you will find a predefined example, which you have to update with your data. First check the ***Use for Sensor Init*** checkbox at the top.

To get into the programming mode of the sensor, you first have to set the Delay and Value correctly. You can find the Delay as shown here for example in a graphic in the sensor datasheet. This indicates that the delay must be at least 31 sync-pulses long. Therefore "0x1f" is entered at Delay, because 31 = 1f in hexadecimal.

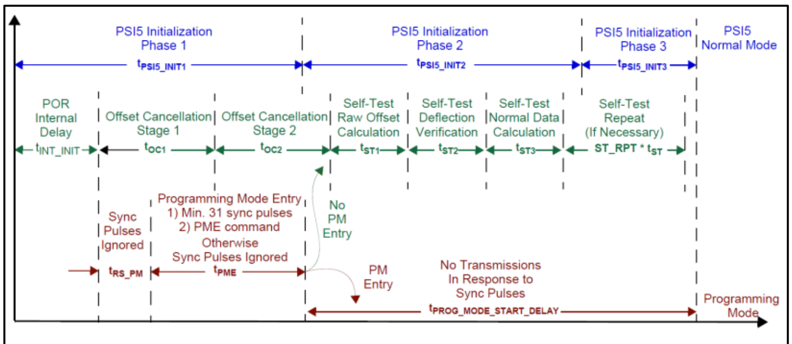


Figure 32. Initialization Timing

You also set the correct value with data from the data sheet. The values are listed as hexadecimal and must be converted to decimal values. Here the SAdr has the decimal value = 1 and FC has the decimal value = 7. You can set these values in the ECU Pattern Editor at the bottom left and then copy the resulting value into the correct cell.

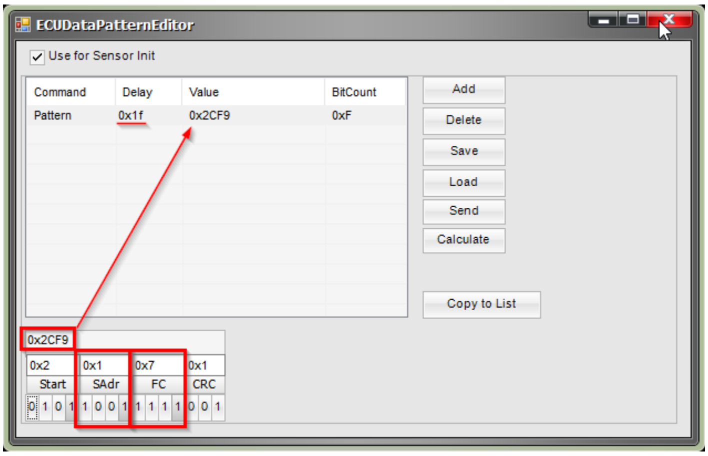
Table 17. Programming Mode Via PSI5 Commands and Responses

#	CMD Type	SAdr	FC	Command	Register Address	Data Field	Response (OK)			Response (Error)		
							RC	RD1	RD0	RC	RD1	RD0
S0	Short	001	100	Execute Programming of NVM	N/A	N/A	OK	0x2AA	N/A	Error	ErrN	N/A
S1	Short		101	Invalid Command	N/A	N/A	No Response			No Response		
S2	Short		110	Invalid Command	N/A	N/A	No Response			No Response		
S3	Short		111	Enter Programming Mode	N/A	N/A	OK	0x0CA	N/A	No Response		
LR	Long		010	Read nibble located at address RA5:RA0	Varies	Varies	OK	RData	RData+1	Error	ErrN	0x000
LW	Long		011	Write nibble to register RA5:RA0	Varies	Varies	OK	WData	RA5:RA0	Error	ErrN	0x000
XLR	XLong		000	Invalid Command	Any	Any	No Response			No Response		
XLW	XLong		001	Invalid Command	Any	Any	No Response			No Response		

Note: When reading the last address in the data array, RData+1 will always return 0x00.

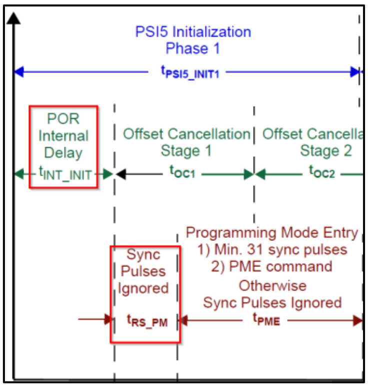
PSI5 Sensor Programming

Make sure that the number above the respective description (SAdr / FC) must have the same decimal value as previously determined. The hexadecimal number cannot be entered 1:1 below it. For this example, the ECU Pattern Editor looks like this:



To close the editor, click Close in the upper right corner.

To enter the programming mode, the values "POR Internal Delay" and "Sync Pulses Ignored" in the data sheet must be read out and added together to determine the advance.



The synchronization pulse is specified with at least 58 milliseconds for this sensor. The Internal Oscillator Frequency is 4 MHz = 0.00025 ms and is calculated into the Internal Delay, so the Internal Delay = $\frac{16.000ms}{1/_{0,00025\ ms}} = 4\ ms$. Together, this is now 62 milliseconds.

2.6 Dynamic Electrical Characteristics - PSI5

$V_L \le (V_{CC} - V_{SS}) \le V_H$, $T_L \le T_A \le T_H$, $\Delta T \le 25$ K/min, unless otherwise specified

#	Characteristic	Symbol	Min	Typ	Max	Units
104	Synchronization Pulse (Figure 5, Figure 28 and Figure 32) Reset to first sync pulse (Program Mode Entry)	t_{RS_PM}	58	—	—	ms

2.7 Dynamic Electrical Characteristics - Signal Chain

$V_L \le (V_{CC} - V_{SS}) \le V_H$, $T_L \le T_A \le T_H$, $\Delta T \le 25$ K/min, unless otherwise specified

#	Characteristic		Symbol	Min	Typ	Max	Units
138	Internal Oscillator Frequency	*	f_{OSC}	3.80	4	4.20	MHz

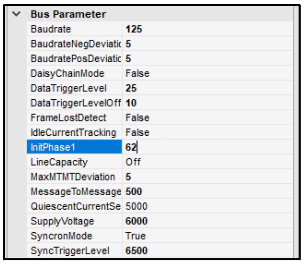
2.8 Dynamic Electrical Characteristics - Supply and SPI

$V_L \le (V_{CC} - V_{SS}) \le V_H$, $T_L \le T_A \le T_H$, $\Delta T \le 25$ K/min, unless otherwise specified

#	Characteristic	Symbol	Min	Typ	Max	Units
177	Reset Recovery Internal Delay (After internal POR)	t_{INT_INIT}	—	16000 / f_{OSC}	—	s

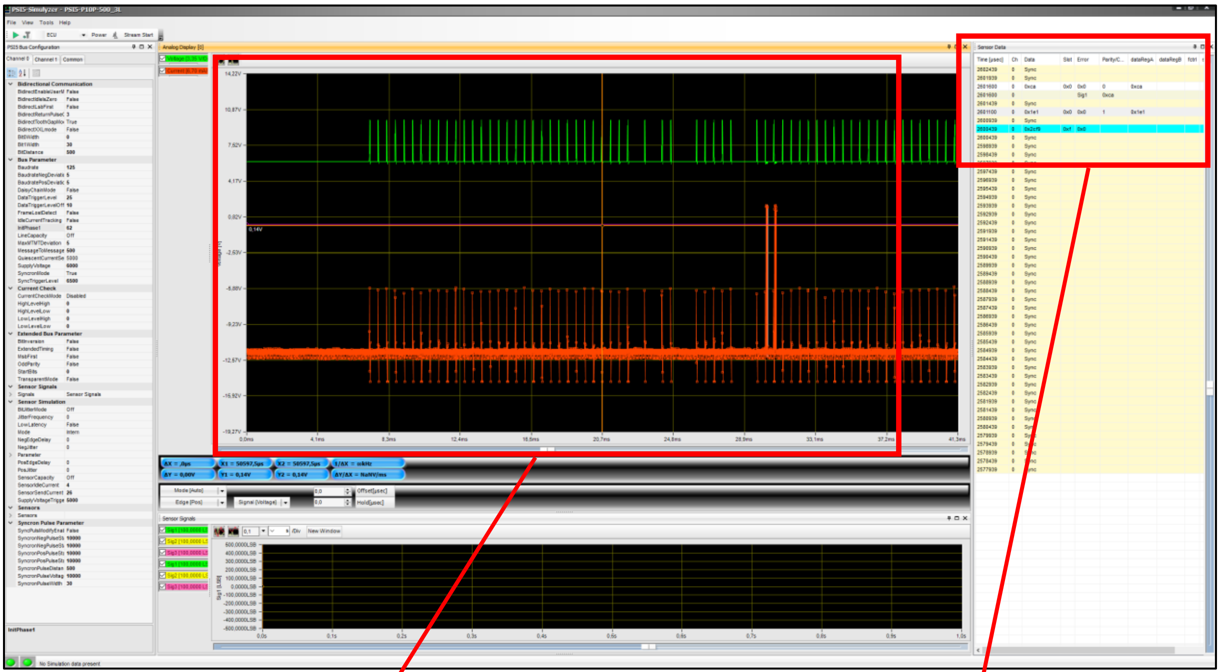
PSI5 Sensor Programming

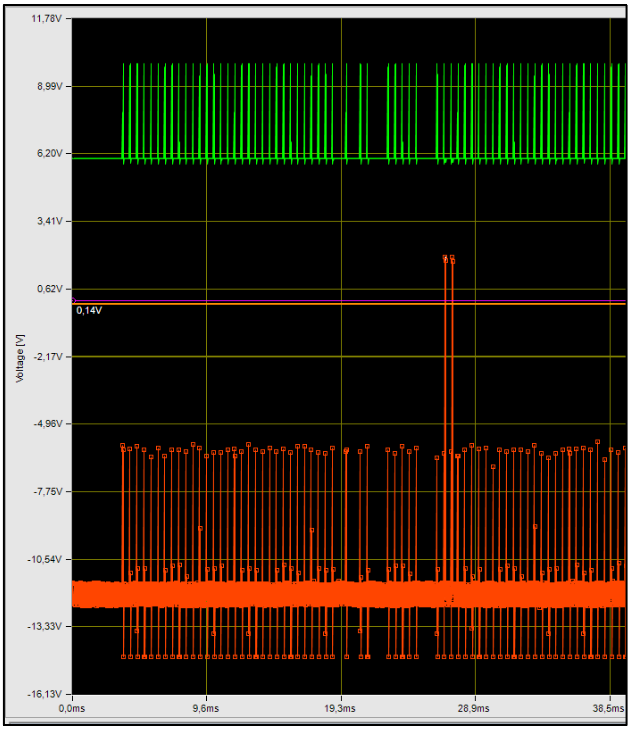
The 62 milliseconds are entered on the left in the software under Bus Parameter at "InitPhase1".

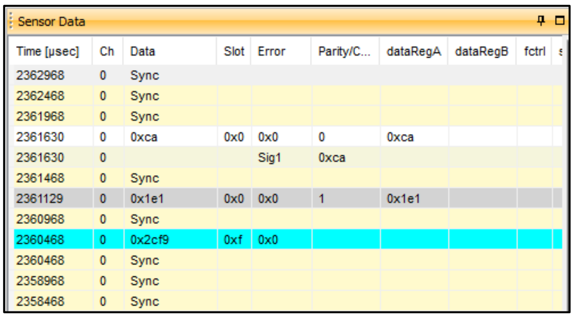


Bus Parameter	
Baudrate	125
BaudrateNegDeviatic	5
BaudratePosDeviatic	5
DaisyChainMode	False
DataTriggerLevel	25
DataTriggerLevelOff	10
FrameLostDetect	False
IdleCurrentTracking	False
InitPhase1	62
LineCapacity	Off
MaxMTMTDeviation	5
MessageToMessage	500
QuiescentCurrentSe	5000
SupplyVoltage	6000
SyncronMode	True
SyncTriggerLevel	6500

Now start the measurement again with the ***green arrow*** and the ***power*** button.
The sensor now goes into programming mode and returns the following image:





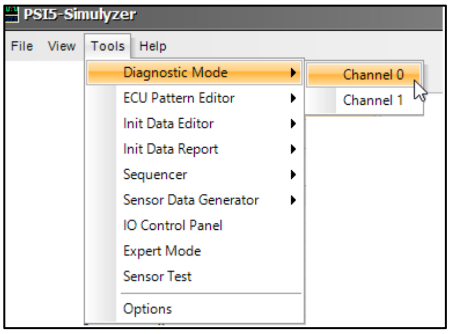


Sensor Data

Time [µsec]	Ch	Data	Slot	Error	Parity/C...	dataRegA	dataRegB	fctrl
2362968	0	Sync						
2362468	0	Sync						
2361968	0	Sync						
2361630	0	0xca	0x0	0x0	0	0xca		
2361630	0			Sig1	0xca			
2361468	0	Sync						
2361129	0	0x1e1	0x0	0x0	1	0x1e1		
2360968	0	Sync						
2360468	0	0x2cf9	0xf	0x0				
2360468	0	Sync						
2358968	0	Sync						
2358468	0	Sync						

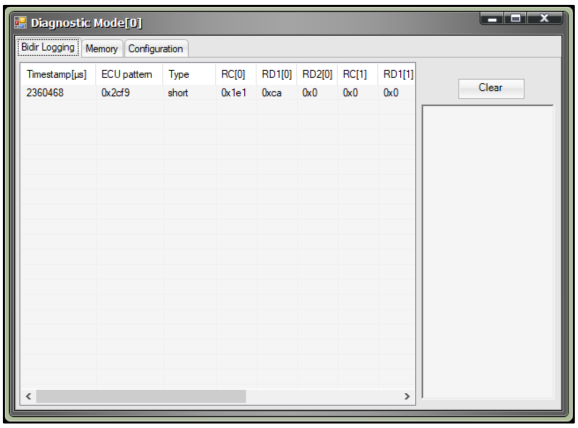
Opening the diagnostic mode

Select the menu group ***Tools*** and the command ***Diagnostic Mode*** and click in the subgroup on the desired ***channel***, which you want to program.



Bidirectional communication

The communication between ECU and sensor is displayed as logging. Here the menu item ***Bidir Logging*** shows the same time stamp and ECU pattern as in the right table.



To find out the programmability of your sensor, you must first find out the CMD type, whether this is a short or long type. It is also important with which FC can be read or written.

5.3.6 Programming Mode Via PSI5 Command and Response Summary

Table 17. Programming Mode Via PSI5 Commands and Responses

#	CMD Type	SAdr	FC	Command	Register Address	Data Field	Response (OK) RC	Response (OK) RD1	Response (OK) RD0	Response (Error) RC	Response (Error) RD1	Response (Error) RD0
S0	Short	001	100	Execute Programming of NVM	N/A	N/A	OK	0x2AA	N/A	Error	ErrN	N/A
S1	Short		101	Invalid Command	N/A	N/A	No Response			No Response		
S2	Short		110	Invalid Command	N/A	N/A	No Response			No Response		
S3	Short		111	Enter Programming Mode	N/A	N/A	OK	0x0CA	N/A	No Response		
LR	Long		010	Read nibble located at address RA5:RA0	Varies	Varies	OK	RData	RData+1	Error	ErrN	0x000
LW	Long		011	Write nibble to register RA5:RA0	Varies	Varies	OK	WData	RA5:RA0	Error	ErrN	0x000
XLR	XLong		000	Invalid Command	Any	Any	No Response			No Response		
XLW	XLong		001	Invalid Command	Any	Any	No Response			No Response		

Note: When reading the last address in the data array, RData+1 will always return 0x00.

PSI5 Sensor Programming

Here is a small explanation how the short or long frame commands look like at least for this sensor:

5.3.2.1 Short Frame Command and Response Format

Short frames are the simplest type of command message. No data is transmitted in a short frame command. Only specific instructions are performed in response to short frame commands. The Short Frame format is shown in Figure 43. Short Frame commands and responses are defined in Section 5.3.6, Table 18.

Start Bits				Sensor Address				Function Code				CRC		
S2	S1	S0	Sy	SA0	SA1	SA2	Sy	FC0	FC1	FC2	Sy	C2	C1	C0
0	1	0	1	1	0	0	1	0	0	1	1	0	0	0

Response	
RC	RD1
$1E2	$3FF

Figure 43. Programming Mode Via PSI5 Short Command and Response Format

5.3.2.2 Long Frame Command and Response Format

Long frames allow for the transmission of data nibbles for register writes. The device can provide register data in response to a read or write request. The Long Frame format is shown in Figure 44. Long Frame commands and responses are defined in Section 5.3.6.

Start Bits				Sensor Address				Function Code				Register Address							Data					CRC			
S2	S1	S0	Sy	SA0	SA1	SA2	Sy	FC0	FC1	FC2	Sy	RA0	RA1	RA2	Sy	RA3	RA4	RA5	Sy	D0	D1	D2	Sy	D3	C2	C1	Sy	C0
0	1	0	1	1	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	1	1	1	1	1	0	0	1	0

Response		
RC	RD1	RD0
$1E2	$3FF	$3FF

Figure 44. Programming Mode Via PSI5 Long Command and Response Format

If you now look again in the manual, then you can find a table with data, which shows you bit functions read and write. With the ***Nibble Addr*** you can get the respective functions in the Configuration Mode. The ***Type*** indicates whether you have ***read or write rights*** in the function.

3 Functional Description

3.1 User Accessible Data Array

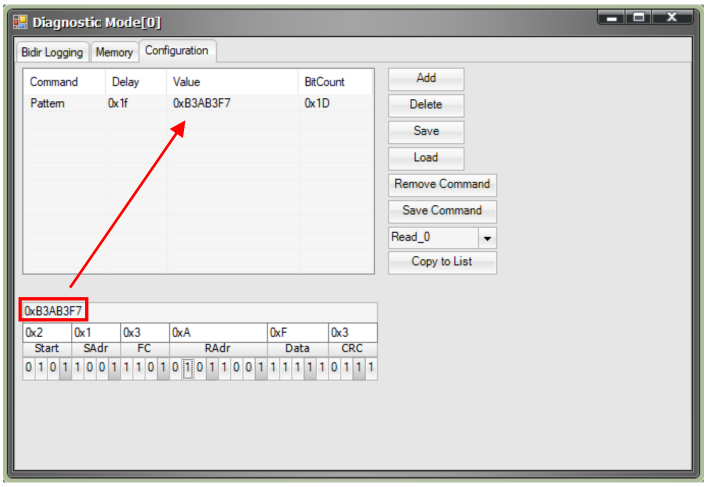
A user accessible data array allows for each device to be customized. The array consists of an OTP factory programmable block, an OTP user programmable block, and read only registers for device status. The OTP blocks incorporate independent CRC circuitry for fault detection (reference Section 3.2). Portions of the factory programmable array are reserved for factory-programmed trim values. The user accessible data is shown in Table 2.

Table 2. User Accessible Data

Byte Addr (XLong Msg)	Register	Nibble Addr (Long Msg)	Bit Function 7	6	5	4	Nibble Addr (Long Msg)	Bit Function 3	2	1	0	Type
$00	SN0	$01	SN[7]	SN[6]	SN[5]	SN[4]	$00	SN[3]	SN[2]	SN[1]	SN[0]	
$01	SN1	$03	SN[15]	SN[14]	SN[13]	SN[12]	$02	SN[11]	SN[10]	SN[9]	SN[8]	
$02	SN2	$05	SN[23]	SN[22]	SN[21]	SN[20]	$04	SN[19]	SN[18]	SN[17]	SN[16]	F, R
$03	SN3	$07	SN[31]	SN[30]	SN[29]	SN[28]	$06	SN[27]	SN[26]	SN[25]	SN[24]	
$04	DEVCFG1	$09	0	0	1	0	$08	0	RNG[2]	RNG[1]	RNG[0]	
$05	DEVCFG2	$0B	LOCK_U	PCM	SYNC_PD	LATENCY	$0A	DATASIZE	BLANKTIME	P_CRC	BAUD	
$06	DEVCFG3	$0D	TRANS_MD[1]	TRANS_MD[0]	LPF[1]	LPF[0]	$0C	TIMESLOTB[9]	TIMESLOTB[8]	TIMESLOTA[9]	TIMESLOTA[8]	
$07	DEVCFG4	$0F	TIMESLOTA[7]	TIMESLOTA[6]	TIMESLOTA[5]	TIMESLOTA[4]	$0E	TIMESLOTA[3]	TIMESLOTA[2]	TIMESLOTA[1]	TIMESLOTA[0]	
$08	DEVCFG5	$11	TIMESLOTB[7]	TIMESLOTB[6]	TIMESLOTB[5]	TIMESLOTB[4]	$10	TIMESLOTB[3]	TIMESLOTB[2]	TIMESLOTB[1]	TIMESLOTB[0]	U, R
$09	DEVCFG6	$13	INIT2_EXT	ASYNC	U_DIR[1]	U_DIR[0]	$12	U_REV[3]	U_REV[2]	U_REV[1]	U_REV[0]	
$0A	DEVCFG7	$15	MONTH[3]	MONTH[2]	MONTH[1]	MONTH[0]	$14	YEAR[3]	YEAR[2]	YEAR[1]	YEAR[0]	
$0B	DEVCFG8	$17	CRC_U[2]	CRC_U[1]	CRC_U[0]	DAY[4]	$16	DAY[3]	DAY[2]	DAY[1]	DAY[0]	
$0C	SC	$19	0	TM_B	RESERVED	IDEN_B	$18	OC_INIT_B	IDEF_B	OFF_B	TEMPF_B	R

Type codes
- F: Freescale programmed OTP location
- U: User programmable OTP location via PSI5
- R: Readable register via PSI5

In ***Diagnostic Mode*** the Long Frame Command can be set in the ***Configuration***. It works exactly the same as in the ECU Pattern Editor with Copy/Paste. With ***Add*** you can add a new command or you can delete it with ***Delete***.
If you want to save a command, you can name it where "Read_0" is now and save it with ***Save Command***.



Now you can find the individual commands in the ***Bidir Logging*** in the right column. By clicking on them, these commands will be run. With ***Clear*** you can delete the left logging.