

Test Report

Integration Test Report CanFblGM

General Information

		Version
Test Specification	Test Specification – GM FBL Diagnostics Tests	1.0
Requirement reference	GM SLP5 RequirementsSpec_SLP5.doc	
Test Person	Jason Learst	
Test Proceeding Date	2015-02-16 to 2015-02-23	
Test Report Recipients	Vector Informatik GmbH	

Test environment

Component		Version / Derivative
Target System	TMS570LS0714PGEQQ1	
Generation Tool	GENy	01.04.0043.0000
Compiler	TI CCS v4.9.5	
Test Database	demo_sw.dbc	-

Component under test

Component		Configuration	Version	Date
GM FBL		Standard	05.03.02	

Overall Test result	OK
Details / hints	-

Revision history of template

Version	Editor	Description
1.0.0 dated 2014-01-27	A. Wenckebach	Create template (derived from GM SLP4)
1.1.0 dated 2014-02-27	A. Wenckebach	Minor clarifications and improvements on configuration description. Added Multi Processor configuration test case
1.2.0 dated 2014-03-20	A.Wenckebach	Further clarifications
1.3.0 dated 2014-04-16	A.Wenckebach	Improvements
1.4.0 dated 2015-02-09	A.Wenckebach	Add Compression test cases, Improvements

Test Report distribution

After performing the test, this test report and procedure test results must be filed in project folder _Doc.

An electronic copy must be put in the ALMplus delivery.

Abbreviations / Glossary

Abbreviation	Description
OK	The obtained result was in accordance to the expected.
N. a.	Not applicable. The selected test configuration excludes this test.
N. t.	Not tested. The test was not performed.
Failed	The obtained result was not in accordance to the specification.

Table of contents

1 Detailed description of test environment	1
2 Setup and General Test Proceeding	2
2.1 DUT Preparation Phase	2
2.2 Check Test Entry Criteria.....	2
2.3 Specify Tests to be performed during the Integration / Module Test.....	2
2.4 Test End Phase	2
2.5 Post Test Phase.....	2
3 Test Category	3
3.1 Automatic test Procedures.....	3
3.1.1 General Info	3
3.1.2 File Modifications	3
3.1.3 Download Container configuration.....	3
3.1.4 DUT configuration.....	5
3.1.5 Link to Automatic Test-Suite Report.....	6
3.2 Download module Tests	7
3.3 Static code test.....	7
3.4 Requirement tests	8
3.4.1 Set-1 Tests	8
3.4.2 Set-2 Tests	10
3.4.2.1 All Set-2 Test Cases.....	10
3.4.2.2 For deliveries that support multiple device-drivers	10
3.4.2.3 For deliveries that support multiple Application Files	10
3.4.3 Set-3 tests.....	12
3.4.4 Set-4 tests.....	13
3.4.5 Set-5 tests.....	15
3.4.6 Set-6 tests.....	17
3.4.7 Set-7 tests.....	19
3.4.8 Set-8 tests.....	20
3.4.8.1 All Set-8 Tests	20
3.4.9 Watchdog Timing Measurement.....	21
4 Compression module.....	22
5 Test metrics	23

1 Detailed description of test environment

Test hardware

Test hardware	Nexteer Hardware w/ TMS570LS0714PGEQQ1
Controller/Derivative	TMS570LS0714PGEQQ1
Frequency / Oscillator	20MHz, w/ PLL 160MHz
Memory	768Kb Flash, 128Kb RAM
Communication channels	DCAN_0

For details on compiler options, module version please check

[DeliveryDescription_CBD1400501.html](#).

2 Setup and General Test Proceeding

2.1 DUT Preparation Phase

Assemble test system. Build application. Prepare data files for individual test-cases.

2.2 Check Test Entry Criteria

Are necessary resources available?

No issues on delivery?

Compiler options as required in place?

2.3 Specify Tests to be performed during the Integration / Module Test

Evaluate criteria to select/deselect test.

A) No compression module included

Results: (excluded tests with reasons for exclusion)

Exclude 20.02.00, 20.03.00, 20.04.00 through A)

2.4 Test End Phase

For each test: retrieve and store test results/configuration.

After testing: Update Test metrics.

Store Automatic Test-Suite Report (TestReport.html) in ZIP file.

2.5 Post Test Phase

Place the Test Report into the ALMplus delivery folder ..\Dxx_doc\30_TestReports

Distribute Test Report(s) to customer.

3 Test Category

3.1 Automatic test Procedures

3.1.1 General Info

Please also check the _Doc folder for description how to configure CANoe.

Insert all modules with header Information in binary format.

Be sure to configure CANoe menu Diagnostic/Iso Tp Can identifiers as required.

E.g. Type B IDs: request 0x249/ response 0x649 (usual configuration)

Type C IDs: request 0x45/ response 0x345

Please check the traces for error frames, especially for single wire CAN ECUs this may indicate Baudrate switch problems and/or you require changing XML.can, see below.

Note: Neither application nor calibration files shall be present on the ECU when tests are executed.

The Test environment is known to run with Canoe version smaller V8.1. .

3.1.2 File Modifications

The ap-files fbl_ap.c, fbl_apdi.c and headers have to be modified in order to allow all tests to be correctly executed. Please take the required modifications from any previous test setup.

This macro has to be set:

```
#define ENABLE_TEST_PROCEDURES
```

If you have an ECU that takes more time to reset than 100ms, please change the variable ECU_INIT_TIME_DELAY in the XML.can CAPL node and recompile before starting tests.

```
// Configure delay in addition to expected ~100ms reset  
int ECU_INIT_TIME_DELAY = 500;
```

```
// Configure delay if memdriver deinit takes long time.  
int ECU_DEINIT_TIME_DELAY = 100;
```

The timings will change the time tests wait to start sending messages after reset and will affect the Security delay related tests (timer starts after reset).

3.1.3 Download Container configuration

Several download containers need to be created for the Automated Test environment. These can be either manually created (changes described below), or created using Generate_Automated_Test_Files.bat. The following download containers need to be configured:

1. Demo Application with invalid starting address:

Modify first PMA entry starting address to invalid address not defined in Flashblock table (e.g. 0xFFFFFFFF for most controllers will do). No need to update checksum.

2. Demo Application with invalid data Blocksize:

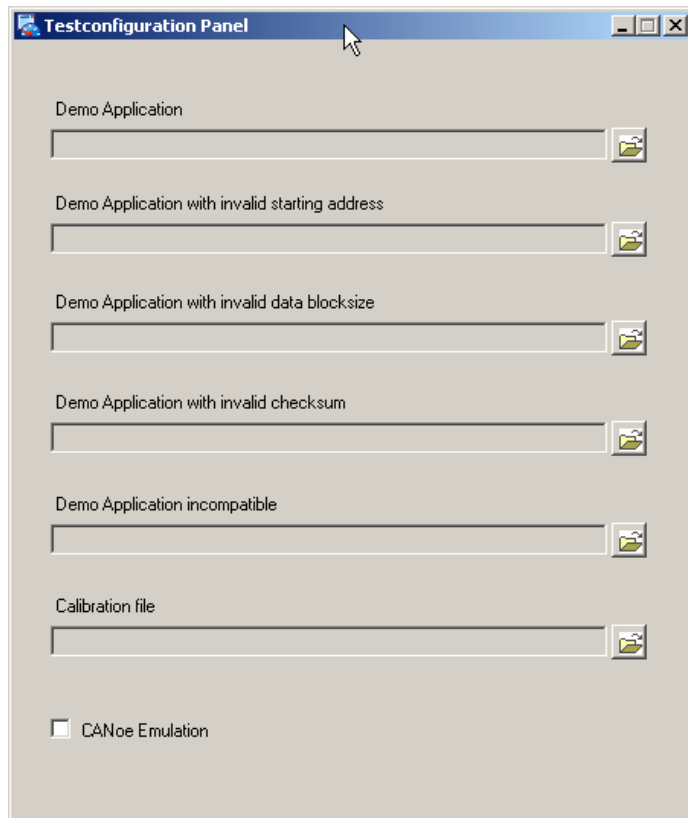
Modify first PMA entry length to invalid address that leads to value not defined in Flashblock table (e.g. 0xFFFFFFFF for most controllers will do). No need to update checksum.

3. Demo Application with invalid Checksum:

Modify data to different value, which makes digest not matching (e.g. set FF last 4 data bytes). Or use Container with destroyed Digest

4. Demo Application incompatible:

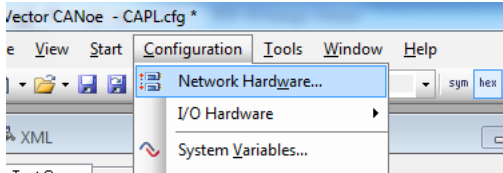
Change DCID value of application to invalid value (0x8000->e.g. 0x8001).



3.1.4 DUT configuration

Prepare the DUT configuration as described in the table below and verify your test configuration on the DUT as followed:

1. Run CANoe (CAPL.cfg), switch your view to desktop “ManualTests”
2. Remember to set the correct baudrate in CANoe.



3. Set configuration switches named below in GENy, generate and rebuild, load to target
4. Press ‘a’ to start the validation of the DUT configuration, verify FBL configuration
5. Switch your view to desktop “AutomaticTests”
6. Select TypeB_DualWire or TypeB_SingleWire based on the customer request.
7. Execute the automatic tests (XML.can) and enter the test result in the table above.

Test No	Test Name	Test Entry	Expected result	Result
Overall Result	Automatic test procedures	<p>Use GENy to create initial configuration per Test Specification. The generated file, fbl_cfg.h, must contain:</p> <p><u>GENy Menue configuration:</u> FBL_DISABLE_MULTIPLE_NODES (deselect multiple ECUs in “Menu Configuration->Edit Channel”)</p> <p><u>FblDrvCan_XXX component configuration:</u> FBL_ENABLE_APPL_TASK FBL_WATCHDOG_ON Project state Production (reduced) FBL_DISABLE_COMPRESSION</p> <p><u>FblCan_14230_Gm component configuration:</u> FBL_ENABLE_CALIBRATION_MODULES FBL_ENABLE_CAN_CONFIGURATION/ (ECU SPS Type: TYPE_B)</p> <p>FBL_DISABLE_RAM_INTEGRITY_CHECK FBL_DISABLE_ROM_INTEGRITY_CHECK FBL_ENABLE_INITIATE_DIAG_OP FBL_ENABLE_READ_MEMORY_BY_ADDRESS FBL_ENABLE_WRITE_DATA_BY_IDENTIFIER FBL_ENABLE_SECURITY_ACCESS</p>	OK	OK

3.1.5 Link to Automatic Test-Suite Report

If the Automatic Test-Suite passes all test-cases, enter “OK” in the Result field above. Otherwise, enter “Failed” for the overall result, and list the test-cases that failed. Add the Report file to the Test results in the project repository. Automatic test procedure results are also provided to the customer.

3.2 Download module Tests

Please verify that FblHdr download module testing has been run on latest version of fbl_hdr.c.
FblHdr download module tests verify correct

- SBA-ticket handling
- Key-NBID/App-NBID handling
- Gap Fill
- Partition handling
- Signature handling

Test No	Test Name	Test Entry	Expected result	Result
N/A	FblHdr download module tests	Tests have been executed on delivered FblHdr module release. Check if version has been tested: ComponentTestReport.log Tests need to get executed on version if not yet done.	Tests have been executed on the version delivered.	OK

3.3 Static code test

Test No	Test Name	Test Entry	Expected result	Result
01.02.00	Bootloader Code Inspection	For project specific Software modules / changes: Perform inspection on all loops ("for", "do" and "while") to verify watchdog and response-pending handling for usually not delivered modules (Also verify in watchdog measurement, that these modules are executed)	All loops contain unconditional watchdog support; All loops in service-handler functions contain response-pending callbacks	OK

3.4 Requirement tests

3.4.1 Set-1 Tests

Check Before performing the following tests, change the configuration used in the previous test set (via GENy) as follows. :

FblDrvCan_<HW>:

- Verify “Stay in Boot” setting (check 16.02.00 once with disabled option and once with enabled)
- Add a second channel “Init Structure” to enable FBL_ENABLE_HIGHSPEED_MODE if applicable (Single Wire Can)
- Disable FblStart Function (Comment out Placeholder Function *ApplFblReadCanInitTable()* in fbl_apnv.c)
- Set Project State to “Integration”

FblCan_14230_Gm->GM FBL configuration Settings:

- ECU Sps Type “TYPE C”
- Enable “RAM Integrity Check”
- Enable “ROM Integrity Check”
- Set Start-/End-Address to appropriate value
- Disable “Gap Fill”

FblCan_14230_Gm->GM Diagnostic Service Options

- Verify enabled “Security-Access”

Several tests for this configuration are verified semi-automatically. To run these tests, prepare the DUT as described in the Test-Specification. Run CANoe, load the CAPL.cfg configuration, and press ‘1’. You will be prompted as needed to enter the input files.

Enter the overall result of the semi-automatic tests in the table below. Also identify any test cases that fail (See the report file ManualReport.html for details).

Test No	Test Name	Test Entry	Expected result	Result
Overall Result			OK	
04.01.00	SPS-TYPE-C Single-Node	Implement DID \$7E in ApplFblReadDataByIdentifier() to delay before sending response (wait 10.1 seconds when not in programming mode, 60.1 seconds when in programming mode).		OK
08.05.00	Response Timing			

The following test-cases are performed manually. Record the result of each test in the table below.

Test No	Test Name	Test Entry	Expected result	Result
06.01.00	RAM & ROM check valid	Enter address-region data and calculated checksum into FBL GM-Header inside GENy (PMA Address, PMA Length and Boot Info Block: Checksum)	Report-Programmed-State does not indicate RAM or ROM integrity-check errors. Verify Bits state shows Ram/Rom Integrity OK (in Canoe send Sequence F2 from tab "Sequences") Check Response manually: TX: A2 RX: 02 E2 00	OK
16.01.00	Sleep-Mode	i.E. watch ApplFblEnter-StopMode() with a debugger	FBL stays awake as long as messages are received. FBL attempts to sleep 60 seconds after last message is received.	OK
16.02.00	Stay-in-Boot Mode	GENy: set STAY_IN_BOOT Download an application. Verify it is running. vFlash: set "Force Boot Mode". Download again a application. Wait. Cause a reset to jump into boot-loader. The bootloader responds to the ping messages and the download starts. GENy: clear STAY_IN_BOOT Download an application. Verify it is running. vFlash: set "Force Boot Mode". Download again a application. Wait. Cause a reset to jump into boot-loader. The bootloader does not respond to the ping messages and the download never starts.	FBL responds to "ping" message and does not start Operating Software upon reset if fbl_cfg.h contains FBL_ENABLE_STAY_IN_BOOT. FBL does not respond to "ping" message, and starts Operating Software upon reset if fbl_cfg.h does not contain FBL_ENABLE_STAY_IN_BOOT.	OK

Save the configuration and Test results (use script /_Tsi_Set1/GetResults_AND_cleanup.cmd).

3.4.2 Set-2 Tests

Before performing the following tests, change the configuration used in Set-1 (via GENy) as follows:

3.4.2.1 All Set-2 Test Cases

General:

- On the Configuration Menu in GENy, select “Edit Channel (channel 0)”. Select both “DUT” and “DUT1” nodes. Enter Node Addresses 0x45 and 0x46 in GENy.

FblDrvCan_<HW>:

- Enable FblStart Function
- Verify disabled Stay in boot

FblCan_14230_Gm->GM FBL configuration Settings:

- Disable “RAM Integrity Check”
- Disable “ROM Integrity Check”
- Verify disabled “Gap Fill”

FblCan_14230_Gm->GM Diagnostic Service Options

- Disable “Security-Access”

3.4.2.2 For deliveries that support multiple device-drivers

If your delivery supports multiple devices (such as flash and EEPROM), change the configuration as follows:

- Add Secondary Device Type to configuration, use it in FlashBlocktable.

3.4.2.3 For deliveries that support multiple Application Files

FblCan_14230_Gm->GM Modules configuration

- Enable “Multiple application support” (Add Data Element to WrapperNv “App_NBID_2ndApp” to use Demo prepared solution)

Generate the new configuration files, compile the FBL, and load the new DUT onto the target hardware.

Several tests for this configuration are verified semi-automatically. To run these tests, prepare the DUT as described in the Test-Specification. CANoe should already be running; press ‘2’ to start the tests. 12.02.00 needs to be run with vFlash (configure e.g. 1037 byte)

Enter the overall result of the semi-automatic tests in the table below. Also identify any test cases that fail (See the report file ManualReport.html for details).

Test No	Test Name	Test Entry	Expected result	Result
Overall Result 04.03.00	SPS-TYPE-C Multi- ple-Node	ApplFblCanParamInit() implemented to select diagnostic-address based on I/O (switch) settings.	OK	OK

The following test-cases are performed manually. Record the result of each test in the table below.

Test No	Test Name	Test Entry	Expected result	Result
12.02.00	Data spans multiple Address-Regions	Create Operating S/W GBF file with two address-regions and one or more calibration-file entries	vFlash downloads operating S/W successfully. (Remember to deactivate in vFlash: Configure/Misc./Security Access	OK
14.02.00	Interrupt Calibration D/L	Create dummy Calibration module	FBL restarts successfully after an interrupted (via power-loss, connection-failure, or reset) download. FBL completes allows download after an interrupted download.	OK

After performing the above tests, save the generated configuration files (all found in the GenData folder), the GENy configuration file (found in the CANdb folder), modified source-files, and generated data files to a uniquely-named ZIP file (for example, Set_2_cfg.zip).

3.4.3 Set-3 tests

Before performing the following tests, change the configuration used in Set-2 (via GENy) as follows:

FblCan_14230_Gm->GM FBL configuration Settings:

- ECU SPS Type to “TYPE_B”
- Enable “RAM Integrity Check”.
- Enable “Gap Fill”

Generate the new configuration files, compile the FBL, and load the new DUT onto the target hardware.

The tests for this configuration are verified semi-automatically. To run these tests, prepare the DUT as described in the Test-Specification. CANoe should already be running; press ‘3’ to start the tests. You will be prompted as needed to enter the input files.

Enter the overall result of the semi-automatic tests in the table below. Also identify any test cases that fail (See the report file ManualReport.html for details).

Test No	Test Name	Test Entry	Expected result	Result
Overall Result 04.04.00	SPS-TYPE-B Multi- ple-Node	ApplFblCanParamInit() implemented to select diagnostic-address based on I/O (switch) settings.	OK	OK

After performing the tests, save the generated configuration files (all found in the GenData folder), the GENy configuration file (found in the CANDb folder), modified source-files, and all generated data files to a uniquely-named ZIP file (for example, Set_3_cfg.zip).

3.4.4 Set-4 tests

Before performing the following tests, change the configuration used in Set-3 (via GENy) as follows:

General:

- On the Configuration Menu, select “Edit Channel (channel 0)”. Select only “DUT” node (de-select “DUT1” node). Enter Node Address 0x45 in GENy

FblDrvCan_<Hw>->Memory Configuration:

- Remove Secondary Device Types
- Change Project State to “Production (Reduced Checks)”
 - Keep NVM Dummy init (if applicable) in ApplFblInit, e.g. by asking for VGEN_ENABLE_BRS
- Verify disabled stay in boot

FblCan_14230_Gm->GM FBL configuration Settings:

- Disable “RAM Integrity Check”
- Enable “ROM Integrity Check”, configure wrong checksum

FblCan_14230_Gm->GM Modules configuration

- Disable “Calibration Module support”
- Disable “Multiple application Module support” if applicable

FblCan_14230_Gm->GM Diagnostic Service Options:

- Enable “Security-Access”
- Disable “Write Data By Identifier”
- Disable “Initiate Diagnostic Operation”
- Disable “Read Memory By Address”
- Seed and key size: 2

User Preconfig:

- Add #define FBL_ENABLE_VERIFY_INTEGRITY_WORD

Generate the new configuration files, compile the FBL, and load the new DUT onto the target hardware.

Several tests for this configuration are verified semi-automatically. To run these tests, prepare the DUT as described in the Test-Specification. Press ‘4’ to start the tests. You will be prompted as needed to enter the input files.

Enter the overall result of the semi-automatic tests in the table below. Also identify any test cases that fail (See the report file ManualReport.html for details).

Test No	Test Name	Test Entry	Expected result	Result
Overall Result 04.02.00	SPS-TYPE-B Single Node	None	OK	OK

The following test-cases are performed manually. Record the result of each test in the table below.

Test No	Test Name	Test Entry	Expected result	Result
12.13.00	High-Speed Download	None	FBL is able to download Operating Software at 83.333KBPS(single wire) / standard speed (non single wire). Operating Software is able to start FBL to download at this speed. Verify Bits state shows Rom Integrity error while in Boot-loader (in Canoe send F2_BIS_ONLY_TYPE_B from tab "Sequences"). First byte in response is 0x01: <ul style="list-style-type: none">- 1A F2- 5A F2 x1 xx (A2 service will also show Rom Error Response "E2 53")	OK

After performing the above tests, save the generated configuration files (all found in the GenData folder), and the GENy configuration file (found in the CANdb folder) to a uniquely-named ZIP file (for example, Set_4_cfg.zip).

3.4.5 Set-5 tests

Before performing the following tests, change the configuration used in Set-4 (via GENy) as follows:

FblDrvCan_<Hw>->Memory Configuration:

- Add Device “Dummy” to FlashBlock Table (Segment Size equal to “Flash” device)
- Add additional real devices, if applicable
- Configure Flashblock that contains Calibration file to “Dummy” device.
- Add fbl_dio.c and fbl_dio.h to the project (this adds a second device which implements the His “Dummy” Device functions; these simply map to the internal flash device)
- Change Project State to “Integration”
- Disable “Sleep Mode”

FblCan_14230_Gm->GM Module configuration:

- Enable “Calibration module support”
- Enable “Multiple Application support” if applicable

User Preconfig:

- Remove `#define FBL_ENABLE_VERIFY_INTEGRITY_WORD`

Use a configuration with several Cal partitions, e.g.:

- First partition including cal mid2, mid3
- 2nd partition including cal mid4

Generate the new configuration files, compile the FBL, and load the new DUT onto the target hardware.

Test No	Test Name	Test Entry	Expected result	Result
11.01.00 08.06.00	Erase during Operating Software D/L	<p>Create dummy Operating S/W to fill all programmable memory (for each application, if several available).</p> <p>Check in GENy LBT “Header Address” and “Presence Pattern Address” to allow this (header at start, pattern at end of memory).</p>	<p>FBL is able to download Demo Application after downloading dummy application that fills all memory.</p> <p>Memory unused by Demo Application contains fill char.</p> <p>Check this with all available drivers (use within Fill All file(s) to all available mem): response pendings are seen regularly during erase (check correct format and timing in trace).</p>	OK

11.03.00 14.01.00	Erase Calibration Partitions And Download of multiple Calibrations in one partition	If Appl/Large file Presence Pattern/header location is in conflict with Calibration Region now: Update in GENy LBT "Header Address" and "Presence Pattern Address" to allow for calibration files, rebuild.	Download of all calibration modules is successful. Re-download of "first" calibration module erases second module in same partition from memory. Download of 2 nd cal from partition alone will Cause Err_PartitionId (0x0001) for Pec error	OK
------------------------------------	---	--	---	----

After performing the above tests, save the generated configuration files (all found in the GenData folder), the GENy configuration file (found in the CANdb folder), and all generated data files to a uniquely-named ZIP file (for example, Set_5_cfg.zip).

3.4.6 Set-6 tests

Before performing the following tests, change the configuration used in Set-5 (via GENy) as follows:

FblDrvCan_<Hw> Memory Configuration:

- Remove Device “Dummy” from Memory Configuration,

FblCan_14230_Gm->GM FBL configuration Settings:

- Disable “Gap Fill”

FblCan_14230_Gm->GM Module configuration:

- Disable “Calibration Modules support”

Preconfig

- Configure loading SBA-ticket in ApplFblInit (#define FBL_TEST_SBA_TICKET) for 07.01.00 and 20.01.00 a), remove or modify loaded SBA ticket to leave it invalid for 20.01.00 b)

If you do not have a compiled in device driver, set 07.01.00 to n/a.

Generate the new configuration files, compile the FBL, and load the new DUT onto the target hardware.

Test No	Test Name	Test Entry	Expected result	Result
07.01.00	Multiple Devices / Compiled-in Device-Driver	Create two dummy data files with GM-Header First download the larger, that the smaller module.	Verify Erasure of compiled in driver, e.g. by checking: Data files are successfully downloaded. Memory not used by second and smaller data file is erased.	OK

20.01.00	Wrong signature/digest with/without SBA ticket	<p>Destroy digest value in header by manually modifying it.</p> <p>Download the file once with loaded, once with unloaded SBA ticket.</p> <p>The Fbl will report on 1A F2 the BIS status (2 byte information; e.g. send by vFlash)</p> <p>Request: 1A F2 Response: 5A F2 01 8X</p>	<p>With correct SBA ticket loaded:</p> <p>Verify MSB in 2nd Bis byte is set (2nd byte = 8X)</p> <p>Download OK</p> <p>Without correct SBA ticket loaded:</p> <p>Verify Bis byte set to value different to 8X (2nd byte != 8X)</p> <p>Download fails with Pec error Err_Signature (0x0015), if no SBA ticket loaded; download is successful else.</p> <p>For the case no ticket is loaded:</p> <p>Add the time between \$36 request and NRC response to the result field.</p> <p>This is close to the time required to calculate the signature on this Ecu. Note: This time is influenced by</p> <ul style="list-style-type: none">• CPU power (PII/architecture)• Execution speed (wait states, 16/32bit access, execution from Ram)• Watchdog triggering• etc.. <p>Check if the time is acceptable for the given ECU, consider informing the customer in Readme.</p>	OK / 0.22708s
----------	--	--	---	---------------

After performing the above tests, save the generated configuration files (all found in the GenData folder), the GENy configuration file (found in the CANdb folder), and all generated data files to a uniquely-named ZIP file (for example, Set_6_cfg.zip).

3.4.7 Set-7 tests

Before performing the following tests, change the configuration used in Set-6 as follows:

FblDrvCan_<Hw>->Memory Configuration:

- Change Project State to “Production”

Preconfig

- Remove loading SBA-ticket in ApplFblInit (remove #define FBL_TEST_SBA_TICKET)

Generate the new configuration files, compile the FBL, and load the new DUT onto the target hardware.

Test No	Test Name	Test Entry	Expected result	Result
12.10.00	Start from Operating Software	None	FBL is able to download Demo Application at 33.333KBPS(single wire) / standard speed (non single wire). Demo Application is able to start FBL using that speed. Demo Application is running after download.	OK

After performing the above tests, save the generated configuration files (all found in the GenData folder), and the GENy configuration file (found in the CANdb folder) to a uniquely-named ZIP file (for example, Set_7_cfg.zip).

3.4.8 Set-8 tests

Before performing the following tests, change the configuration used in Set-7 (via GENy) as follows:

3.4.8.1 All Set-8 Tests

FblDrvCan_<Hw>->Memory Configuration:

- Change Project State to “Integration”

FblCan_14230_Gm->GM Module configuration:

- Enable “Calibration module support”
- Enable “Gap Fill”

Generate the new configuration files, compile the FBL, and load the new DUT onto the target hardware.

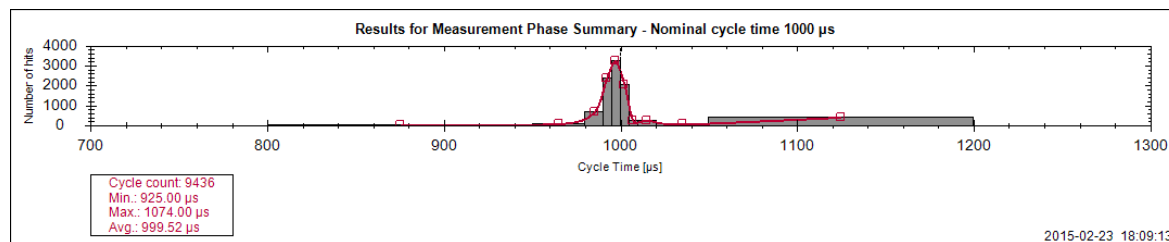
Test No	Test Name	Test Entry	Expected result	Result
11.02.00	Erase during Calibration D/L	Create dummy Operating S/W and three dummy Calibration modules in three different partitions	Download of all calibration modules is successful. Re-download of calibration module does not affect memory containing other calibration modules.	OK
13.01.00	Gaps-Filled	None	All unused memory contains fill character.	OK

After performing the above tests, save the generated configuration files (all found in the GenData folder), the GENy configuration file (found in the CANdb folder), and all generated data files to a uniquely-named ZIP file (for example, Set_8_cfg.zip).

3.4.9 Watchdog Timing Measurement

Configure the watchdog to be triggered every 1ms. This is the minimum accuracy supported by the bootloader. Toggle I/O D[6] during AplFblWDTrigger().

Implement I/O inside the WD-trigger routine. Measure the accuracy of the watchdog timing with an oscilloscope and document the result here (e.g. by providing a screenshot of the performed measurement). Measurement is done during download of the DemoAppl project inclusive cal modules (DemoAppl\Appl\ApplHdr_3Part_each_1Cal).



Toggle Count	Min cycle time [µs]	Max cycle time [µs]	Average [µs]
9436	925.00	1074.00	999.52

4 Compression module

If your delivery contains a compression module perform the below test cases.

To prepare this

- Enable FblDrvCan_<HW> -> Data Processing -> Compression Mode
- Generate the new configuration files, compile the FBL, and load the new DUT onto the target hardware.

Generate the following containers (e.g. use
TsiComon\ManualTestFiles\Generate_Compression_Test_Files.bat):

- Uncompressed application
- Compressed application
- Compressed application, corrupted (modify first data Byte->byte 10)

Hint: The length of the compressed data envelope is stored in address 0x02...0x07 of the container.

Test No	Test Name	Test Entry	Expected result	Result
20.02.00	Compression 1	Download un-compressed container	Download of container is successful	N.a.
20.03.00	Compression 2	Download compressed container	Download of container is successful	N.a.
20.04.00	Compression 3	Download corrupted, compressed container	Download fails i.E.: PEC_LengthExceeded	N.a.

5 Test metrics

No	Metric		
1	Percentage of Tests Executed	Expected result: 19/21	Achieved result: 19/21
2	Percentage of Tests Successfully Executed	Expected result: 19/19	Achieved result: 19/19