**Background pattern

Description automatically generated**

OSSAT OBC Development Board

Verification Procedure

[CC BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/): Open Source Satellite

Reference: KS-DOC-01526-02

Date: 06 December 2022

Author: Kevin Manser

VERSION 02

RELEASED

Contents

[1 Document History 3](#_Toc126834309)

[2 Applicable Documents 3](#_Toc126834310)

[3 References 3](#_Toc126834311)

[4 Glossary 4](#_Toc126834312)

[5 Introduction 5](#_Toc126834313)

[5.1 Objective 5](#_Toc126834314)

[5.2 Scope 5](#_Toc126834315)

[5.3 Applicability 5](#_Toc126834316)

[5.4 Test Rationale 5](#_Toc126834317)

[5.5 Firmware Requirements 5](#_Toc126834318)

[5.6 Handling 5](#_Toc126834319)

[5.7 Personnel 5](#_Toc126834320)

[5.8 Equipment Required 5](#_Toc126834321)

[5.9 Description of EUT 6](#_Toc126834322)

[5.10 Test Selection 6](#_Toc126834323)

[5.11 Recording Results 6](#_Toc126834324)

[6 Inspection 7](#_Toc126834325)

[6.1 Overview 7](#_Toc126834326)

[6.2 Visual Inspection 7](#_Toc126834327)

[6.3 Continuity and Insulation Tests 8](#_Toc126834328)

[7 Initial Powered Tests 9](#_Toc126834329)

[7.1 Overview 9](#_Toc126834330)

[7.2 Test Setup 9](#_Toc126834331)

[7.3 Procedure 9](#_Toc126834332)

[8 Load and Check the Test Firmware 12](#_Toc126834333)

[8.1 Overview 12](#_Toc126834334)

[8.2 Test Setup 12](#_Toc126834335)

[8.3 Procedure 12](#_Toc126834336)

[9 Debug Functions Test 14](#_Toc126834337)

[9.1 Overview 14](#_Toc126834338)

[9.2 Test Setup 14](#_Toc126834339)

[9.3 Procedure 14](#_Toc126834340)

[10 Oscillators and PPS 15](#_Toc126834341)

[10.1 Overview 15](#_Toc126834342)

[10.2 Test Setup 15](#_Toc126834343)

[10.3 Procedure 15](#_Toc126834344)

[11 SRAM Tests 17](#_Toc126834345)

[11.1 Overview 17](#_Toc126834346)

[11.2 Test Setup 17](#_Toc126834347)

[11.3 Procedure 17](#_Toc126834348)

[12 CAN BUS Tests 19](#_Toc126834349)

[12.1 Overview 19](#_Toc126834350)

[12.2 Test Setup 19](#_Toc126834351)

[12.3 Procedure 19](#_Toc126834352)

[13 UART Tests 21](#_Toc126834353)

[13.1 Overview 21](#_Toc126834354)

[13.2 Test Setup 21](#_Toc126834355)

[13.3 Procedure 21](#_Toc126834356)

[14 GPIO Tests 23](#_Toc126834357)

[14.1 Overview 23](#_Toc126834358)

[14.2 Test Setup 23](#_Toc126834359)

[14.3 Procedure 23](#_Toc126834360)

[15 Thermistor Channel Tests 30](#_Toc126834361)

[15.1 Overview 30](#_Toc126834362)

[15.2 Test Setup 30](#_Toc126834363)

[15.3 Procedure 30](#_Toc126834364)

[16 Test Report Sheet 34](#_Toc126834365)

[16.1 EUT Details 34](#_Toc126834366)

[16.2 Equipment used: 34](#_Toc126834367)

[17 Software Apps Used 34](#_Toc126834368)

[18 Acceptance 34](#_Toc126834369)

# Document History

Please see the following record of revisions:

|  |  |  |  |
| --- | --- | --- | --- |
| Document Revision | Document  Status | Change  Description | <SharePoint  Version> - |
| 01 | Draft | Fist issue |  |
| 02 | Released | Update to formal Issue |  |

# Applicable Documents

The following references are applicable to this document.

|  |  |  |  |
| --- | --- | --- | --- |
| Document Reference | Document Title | Date | Reference in this Document |
| KS-SCH-01466 | OSSAT OBC Development Board Schematic |  |  |
| KS-BRD-01467 | OSSAT OBC Development Board PCB |  |  |
| KS- KS-SOF-01541 | OSSAT OBC Development Board Test Firmware Definition |  |  |

# References

The following references are applicable to this document.

|  |  |  |  |
| --- | --- | --- | --- |
| Document Reference | Document Title | Date | Reference in this Document |
|  |  |  |  |

Glossary

|  |  |
| --- | --- |
| AtoD (ADC) | Analogue to Digital Convertor |
| CAN | Controller Area Network (serial bus) |
| DC | Direct Current |
| DGND | Digital ground (electrical ground) |
| DMM | Digital Multi-Meter |
| EUT | Equipment Under Test |
| GND | Ground (electrical ground) |
| GPIO | General Purpose Input / Output |
| LED | Light Emitting Diode |
| LDO | Low Drop Out (referring to a voltage regulator) |
| MCU | Micro Controller Unit |
| OSSAT | Open Source Satellite |
| PC | Personal Computer |
| PCB | Printed Circuit Board |
| PPS | Pulse Per Second |
| PSU | Power Supply Unit |
| SRAM | Static Random Access Memory |
| ST-Link | A debugger / emulator proprietary to STM |
| STM-Cube | A software development environment proprietary to STM |
| TP | Test Point |
| UART | Universal Asynchronous Receiver Transmitter |
| USB | Universal Serial Bus |
| wrt | with respect to |

# Introduction

## Objective

The objective of this document is to define the functional checks and test required to verify the electrical functionality of the OSSAT OBC Development Board.

## Scope

The scope of tests is to confirm electrical/electronic performance of the board.

## Applicability

This procedure is applicable to:

* OSSAT OBC Development Board – KS-BRD-01467

## Test Rationale

While it would be desirable to apply thorough performance tests of each interface, it has been decided to adopt a simpler test strategy which is sufficient to confirm basic electrical functionality and capture connectivity faults. To that end, most of the serial interfaces and the thermistor channels are operated with the related MCU lines configured as GPIO, to provide simple drive or monitoring functions.

## Firmware Requirements

The board is based upon an MCU and consequently this procedure relies upon suitable firmware being loaded, to drive the MCU in accordance with the Test Rationale in section 5.4. The firmware is listed in the ‘References’ section and this procedure includes instructions about loading, as required.

The firmware will provide an on-screen menu function that will select specific test modes. Some functions will start automatically, after reset.

## Handling

The EUT shall be handled with care at all times, while observing full ESD procedures.

## Personnel

Testing shall only be carried out by suitably trained personnel.

## Equipment Required

The following table lists the equipment required for testing the EUT with this procedure.

| Ser | Description | Notes |
| --- | --- | --- |
| 1 | Bench PSU | Capable of delivering 3.3V @ 0.5A |
| 2 | Arbitrary waveform generator (sig gen) | Capable of delivering logic level square waves at audio frequencies. |
| 3 | Oscilloscope | Bandwidth >100MHz |
| 4 | Logic analyser may be required | May be built into the scope |
| 5 | Frequency Counter | Can use arbitrary waveform generator (or oscilloscope) |
| 6 | Lab Test Cables | As required |
| 7 | ST-Link adaptor & Leads | Used to load the Test Firmware. Works with STM32Cube. |
| 8 | CAN Bus analyser | e.g. Microchip CAN Bus Analyser that connects to PC via USB |
| 9 | Test PC (Laptop) running Windows | To be used for driving the ST-Link adaptor for loading the firmware and operating the test Firmware.  Installed apps must include: STM32Cube, a Terminal emulator, e.g Terraterm and CAN bus analyser app, as required. |
| 10 | Various axial resistors | As described in tests |
| 11 | Modified USB cable, with micro-USB connector | Leave Micro-USB connector in place and cut connector off other end. Strip, to permit Red (+5V) and Black (ground) wires to connect to a bench PSU. |

## Description of EUT

The OSSAT OBC Development Board is a small PCB that provides a development environment for software targeted for a central computer in a satellite. This document is intended to verify correct hardware functionality of the OSSAT OBC Development Board.

The board includes a number of peripherals:

* 2 x SRAM (total 4M x 16)
* CAN bus interfaces
* 2 x UART Interfaces
* 8 x Thermistor Temperature monitoring channels
* 31 x GPIO Lines

It also includes an ST-Link interface for boot loading and debug. It also has a test interface that provides 3 x LED drives, 2 x user push buttons and a reset button.

## Test Selection

This document covers multiple tests that are intended to prove the design and verify board build. If multiple boards are built, then there is the option of choosing to run just a subset a subset of the full set of tests. Any reduced scope of testing should be noted.

## Recording Results

The results of each test shall be noted in the spaces provided. Many tests require just a ‘pass’ or ‘fail’ indication, however some tests require a value to be recorded.

# Inspection

## Overview

These tests are carried out with the EUT unconnected and unpowered.

## Visual Inspection

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Verify the board is clean and de-fluxed. | Board clean and de-fluxed |  |
| 2 | Visually inspect the board, as follows: | - | n/a |
| a) | Verify all components are fitted. | All components fitted |  |
| b) | Confirm that capacitor positions do not have resistors fitted. | No resistors fitted to capacitor positions |  |
| c) | Verify values are correct on resistors and IC’s. | Values correct on resistors and IC’s |  |
| d) | Verify polarised components are the correct orientation. | Orientation of polarised components correct |  |
| e) | Verify that all joints are soldered. | All joints soldered |  |
| f) | Verify no visible bridges between adjacent pins. | No visible bridges between adjacent pins |  |
| Notes and observations: | | | |

## Continuity and Insulation Tests

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Use a DMM to check continuity from TP1 to: | - | n/a |
| a) | TP2, TP12, TP15 | <1R |  |
| b) | All mounting holes | <1R |  |
| c) | J1-5, J1-7 | <1R |  |
| d) | J3 (H1)-2,4,6,8,10,15,18,21,23,24,35,36,48,50,52 | <1R |  |
| e) | J4 (H2)-2,4,6,8,10,12,14,16,26,27,28,36,38,40,42,46,48,50,52 | <1R |  |
| 2 | Use a DMM to check continuity <1 Ohm from TP14 to: | - | n/a |
| a) | J4-29, J4-30, J4-17, J4-19 | <1R |  |
| b) | J1-3 | <1R |  |
| 3 | Use a DMM to check DC resistance from J1-5 to J1-11 | 90 to 110R |  |
| Notes and observations: | | | |

# Initial Powered Tests

## Overview

These tests take the EUT through initial power up and carry out some basic checks, before the Test Firmware is loaded. The user buttons are not tested here, because there are no pull up resistors until the firmware is loaded.

## Test Setup

Setup a bench PSU, set to 5V, with **current limit set to approx. 0.3A**, **turned** **off**. Connect the PSU to the EUT via a (modified) USB cable.

The tests in this section require that there is no firmware loaded into the MCU. If firmware has previously been loaded then it should be removed.

## Procedure

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Confirm that no Firmware is loaded in the MCU. | No Firmware is loaded in the MCU |  |
| 2 | Turn on and check the following DC voltages wrt TP2 (DGND). If either is out of spec then investigate the reason before proceeding further. | n/a | n/a |
| a) | Verify the voltage at TP11 (note USB spec is 4.75 to 5.25V under no load). | 4.5 to 5.5V  [Record the value] |  |
| b) | Verify the voltage at TP14. | 3.2 to 3.4V  [Record the value] |  |
| 3 | Measure the supply current into the EUT. If it is out of spec then investigate the reason before proceeding further.  NB the current consumption will be different if the Test Firmware is already loaded. | 0.03 to 0.1A  [Record the value] |  |
| 4 | Check that no components are becoming excessively hot. If they are then investigate the reason before proceeding further. | No components are becoming excessively hot |  |
| 5 | Verify the Power LEDs are on and the other 3 LEDs are off. | Power LEDs are on  Remaining 3 LEDs are off |  |
| 6 | Monitor the voltage on R15 (NRST) wrt DGND. Verify the voltage. | >3V  [Record the value] |  |
| 7 | Verify the voltage on C3 wrt DGND (LDO) | 0.95 to 1.26V  [Record the value] |  |
| 8 | Verify the voltage on C4 wrt DGND (LDO) | 0.95 to 1.26V  [Record the value] |  |
| 9 | The following tests require that the 3V3 voltage is within spec. Use a DMM to verify the following voltages, with respect to DGND (TP1): | - | n/a |
| a) | R3 (thermistors), all pins | 3.2 to 3.4V  [Record the value] |  |
| b) | R10 (thermistors), all pins | 3.2 to 3.4V  [Record the value] |  |
| c) | J1, pins 4, 8, 10, 12 (ST-Link pull ups) | 3.2 to 3.4V  [Record the value] |  |
| d) | U10-1 (CAN pull up) | 3.2 to 3.4V  [Record the value] |  |
| e) | U10-8 (CAN pull down) | 0 to 0.1V  [Record the value] |  |
| f) | U11-1 (CAN pull up) | 3.2 to 3.4V  [Record the value] |  |
| g) | U11-8 (CAN pull down) | 0 to 0.1V  [Record the value] |  |
| h) | R78 both pins (AOCS UART TX) | 3.2 to 3.4V  [Record the value] |  |
| i) | R81 both pins (POWER UART TX) | 3.2 to 3.4V  [Record the value] |  |
| j) | FB1 Both pins | 3.2 to 3.4V  [Record the value] |  |
| k) | R4 both pins | 3.2 to 3.4V  [Record the value] |  |
| l) | FB2 both pins | 3.2 to 3.4V  [Record the value] |  |
| m) | R88 MCU side (JTRST pull down) | 3.2 to 3.4V  [Record the value] |  |
| n) | R15 both pins (NRST) | 3.2 to 3.4V  [Record the value] |  |
| o) | R6 both pins (BOOT0) | 0 to 0.1V  [Record the value] |  |
| p) | R5 both pins (PDR\_ON) | 3.2 to 3.4V  [Record the value] |  |
| q) | R13 (/WE pull up) | 3.2 to 3.4V  [Record the value] |  |
| r) | R17 (/OE pull up) | 3.2 to 3.4V  [Record the value] |  |
| s) | TP3 (/E1 PULL UP) | 3.2 to 3.4V  [Record the value] |  |
| 10 | Monitor the voltage at R15 (either side) wrt DGND. Check the following: | - | n/a |
| a) | Verify the voltage | 3.2 to 3.4V  [Record the value] |  |
| b) | Hold down the reset button. Verify R15 voltage. Then release the button | 0.2 to 0.5V  [Record the value] |  |
| c) | Temporarily short J1-12 (ST-Link reset) to DGND. Verify R15 voltage. Then remove the short. | 0.2 to 0.5V  [Record the value] |  |
| Notes and observations: | | | |

# Load and Check the Test Firmware

## Overview

This section loads the test firmware into the EUT. It implicitly confirms that the ST-Link interface is working. It confirms the test program runs.

## Test Setup

Acquire the test firmware from the repository (typically Github). Store in a convenient location, on a Test PC, e.g. C:\Temp.

Setup a bench PSU, set to 5V, with **current limit set to approx. 0.3A**, **turned** **off**. Connect the PSU to the EUT via a (modified) USB cable.

Plug the ST-Link adaptor into the interface on the EUT and into the Test PC. Start the ST-Link app on the PC and open the Test Firmware.

Start a Terminal Emulator (typically Terraterm) app on the PC that will communicate via the UART on the ST-Link adaptor. Select ST-Link channel and set the serial parameters to: 8-N-1-115200.

## Procedure

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Turn on. | - |  |
| 2 | Use STM32Cube to load the Test Firmware onto the EUT. Verify it reports that the load has been successful. | Load successful |  |
| 3 | Reset the EUT by momentarily pressing the Reset button. Verify the test program logo/menu is re-written to the terminal. This confirms that reset is functioning correctly. | Test program logo / menu re-written to terminal |  |
| 4 | Cycle the power. Verify the Test program logo/menu is re-written to the terminal. | Test program logo / menu re-written to terminal |  |
| 5 | Select a menu function (excluding SRAM) from the menu. Verify the terminal screen reports the relevant test actions and then re-writes the test program logo/menu. | Relevant test action reported to terminal  Test program logo / menu re-written to terminal |  |
| 6 | Measure the supply current into the EUT. If it is out of spec then investigate the reason before proceeding further.  NB the current consumption will be different if the Test Firmware is already loaded | 0.04 to 0.1A  [Record the value] |  |
| Notes and observations: | | | |

# Debug Functions Test

## Overview

This section uses the Test Firmware to verify the Debug I/O functions.

## Test Setup

Setup a bench PSU, set to 5V, with **current limit set to approx. 0.3A**, **turned off**. Connect the PSU to the EUT via a (modified) USB cable. Turn on the power supply.

Start a Terminal Emulator app on the PC that will communicate via the UART on the ST-Link adaptor. Select ST-Link channel and set the serial parameters to: 8-N-1-115200.

## Procedure

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Check the DC voltage on User button1, wrt DGND. | 3.0 to 3.4V  [Record the value] |  |
| 2 | Hold down User Button 1 and check the button voltage. Then release the button. | 0 to 0.1V  [Record the value] |  |
| 3 | Check the DC voltage on User button2, wrt DGND. | 3.0 to 3.4V  [Record the value] |  |
| 4 | Hold down User Button 2 and check the button voltage. Then release the button. | 0 to 0.1V  [Record the value] |  |
| 5 | Verify the LED and Button functions as follows: | - | n/a |
| a) | LEDs 1 & 2 toggle alternately at approx. 2Hz. | LED1 & 2 toggle at approximately 2Hz |  |
| b) | Momentarily press button 2. Verify LED3 latches on. | LED 3 latches on |  |
| c) | Momentarily press button 1. Verify LED3 latches off. | LED 3 latches off |  |
| Notes and observations: | | | |

# Oscillators and PPS

## Overview

This section checks the operation of the crystal oscillators and PPS signal. The MCU oscillator circuits have fairly weak drive, which is affected by scope probe loading. Hence no tests are made at the crystal pins and instead the oscillator outputs at the PPS output are used to check frequencies.

Note that the MCU uses internal RC oscillators, prior to the firmware, configuring the clocks. The PPS accuracy is not specifically tested, because it is derived from one of the oscillators.

## Test Setup

Setup a bench PSU, set to 5V, with **current limit set to approx. 0.3A**, **turned off**. Connect the PSU to the EUT via a (modified) USB cable. Turn on the power supply.

NB: For the counter, ensure noise rejection turned off and be cautious with the probe (Perhaps use a x10 scope probe)

## Procedure

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Monitor GPIO6 with a scope/counter and check the 32.768kHz clock as follows: | - | n/a |
| a) | Verify is has a logic level square wave. | 32.768kHz clock has a logic level square wave |  |
| b) | Verify the duty cycle. | 30 to 70% |  |
| c) | Verify the frequency using a counter.  (Test to +/-10Hz) | 32758 to 32778Hz  [Record the value] |  |
| d) | Cycle the Power at least 20 times, with differing supply voltage/ ramp up speeds (and temperatures). Verify the 32.768kHz clock always starts at the same nominal frequency as above. | 32.768kHz clock always starts with the nominal frequency |  |
| 2 | Monitor the PPS signal and verify it is a logic level square wave of approx. 1Hz | PPS has a logic level square wave at approx. 1Hz |  |
| 3 | Monitor GPIO7 with a scope/counter and check the 25MHz clock as follows: | - | n/a |
| a) | Verify is has a logic level square wave. | 25MHz clock has a logic level square wave |  |
| b) | Verify the duty cycle. | 40 to 60% |  |
| c) | Verify the frequency using a counter  (Test to +/-50ppm) | 24998750 to  25001250Hz  [Record the value] |  |
| d) | Cycle the Power at least 20 times, with differing supply voltage/ ramp up speeds (and temperatures). Verify the clock always starts at the same nominal frequency as above. | 25MHz clock always starts with the nominal frequency |  |
| Notes and observations: | | | |

# SRAM Tests

## Overview

This section uses the Test Firmware to verify the operation of the SRAM. Note that current measurements are made with different operating conditions

## Test Setup

Setup a bench PSU, set to 5V, with **current limit set to approx. 0.3A**, **turned off**. Connect the PSU to the EUT via a (modified) USB cable. Turn on the power supply.

Start a Terminal Emulator app on the PC that will communicate via the UART on the ST-Link adaptor. Select ST-Link channel and set the serial parameters to: 8-N-1-115200.

**OPTIONAL**: Set-up a logic analyser, (using the oscilloscope) and configure as follows:

| Signal | Information |
| --- | --- |
| /E | Memory chip select form MCU. Trigger scope on falling edge |
| /CE0 | Chip select for first SRAM device 0 |
| /CE1 | Chip select for first SRAM device 1 |
| /RD | Read enable |
| /WR | Write enable |
| /UB | Upper byte select |
| /LB | Lower byte select |
| A0 | Address line (LSB) |
| A1 | Address line |

## Procedure

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Run the SRAM Write test and check the following: | - | n/a |
| a) | Verify the test completes. | Test completes |  |
| b) | Monitor the EUT current consumption during the test (while fitted devices are being accessed). | 0.04 to 0.1A  [Record the value] |  |
| c) | **OPTIONAL**: Acquire the write timing waveforms from the logic analyser and store with this document.  Verify timing is compatible with the SRAM datasheet. | [Record the writing timing waveforms]  Recorded waveforms are compatible with SRAM datasheet |  |
| 2 | Run the SRAM Read Test and check the following: | - | n/a |
| a) | Verify the test completes and reports “MEMORY ALL OK OK”, at the end of the status messages. Note if the test fails it will report “ERROR READING MEMORY” | Memory test reports “MEMORY ALL OK OK” |  |
| b) | Monitor the EUT current consumption during the read test (while fitted devices are being accessed). | 0.04 to 0.15A  [Record the value] |  |
| c) | **OPTIONAL**: Acquire the Write timing waveforms, from the logic analyser and store with this document  Verify timing is compatible with the SRAM datasheet | [Record the writing timing waveforms]  Recorded waveforms are compatible with SRAM datasheet |  |
| d) | **OPTIONAL**: Use the scope to check the data lines and verify that all lines are active at some stage, during the test | All lines are active |  |
| 3 | Run the SRAM Chip select test and verify the terminal reports CE0 and CE1 pass and CE2 to CE7 all fail (don’t care about CE8 and CE9 but they probably pass | Terminal reports:  CE0 & CE1 pass  CE2 to CE7 fail |  |
| Notes and observations (including STM32 SRAM tuning): | | | |

# CAN BUS Tests

## Overview

This section uses the Test Firmware to verify the operation of the CAN Busses. The firmware instigates a CAN transaction after reset. If an Ack is received then transmission stops, otherwise the interface will continue to transmit. The test requires a CAN analyser to be attached, in order to service the transaction. This, in turn proves that the EUT can receive as well as transmit.

## Test Setup

Setup a bench PSU, set to 5V, with **current limit set to approx. 0.3A**, **turned off**. Connect the PSU to the EUT via a (modified) USB cable. Turn on the power supply.

Start a Terminal Emulator app on the PC that will communicate via the UART on the ST-Link adaptor. Select ST-Link channel and set the serial parameters to: 8-N-1-115200.

Configure a CAN Bus analyser, with PC app, as required. Set the bitrate to 1MHz. Open a (rolling) monitor window, to observe bus traffic. Note you may have to temporarily disconnect the analyser to clear any errors.

## Procedure

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Plug into CAN channel 1. Set up a scope to monitor the CAN Bus lines. Reset the EUT, to instigate the CAN transaction and check the following: | - | n/a |
| a) | Verify the analyser shows a successful transaction, without an error | Successful transaction with no errors |  |
| b) | Verify the scope shows a single transaction (<150us long), after which the CAN traffic stops. | Single transaction (<150us long) and then no more CAN traffic |  |
| c) | Verify the bus HI signal level when dominant. | 2.7 to 3.3V  [Record the value] |  |
| d) | Verify the bus HI signal level when recessive. | 1.8 to 2.2V  [Record the value] |  |
| e) | Verify the bus LO signal level when dominant | 0.7 to 1.3V  [Record the value] |  |
| f) | Check the bus LO signal level when recessive. Verify its voltage compared with the result for Hi recessive. | -10/-100mV  [Record the value] |  |
| g) | Disconnect the analyser and reset the EUT. Monitor the CAN lines on a scope and verify there is continuous activity | Continuous activity on CAN lines |  |
| 2 | Plug into CAN channel 2. Set up a scope to monitor the CAM Bus lines. Reset the EUT: | - | n/a |
| a) | Verify the analyser shows a successful transaction, without an error | Successful transaction with no errors |  |
| b) | Verify the scope shows a single transaction (<150us long), after which the CAN traffic stops. | Single transaction (<150us long) and then no more CAN traffic |  |
| c) | Verify the bus HI signal level when dominant. | 2.7 to 3.3V  [Record the value] |  |
| d) | Verify the bus HI signal level when recessive. | 1.8 to 2.2V  [Record the value] |  |
| e) | Verify the bus LO signal level when dominant | 0.7 to 1.3V  [Record the value] |  |
| f) | Check the bus LO signal level when recessive. Verify its voltage compared with the result for Hi recessive. | -10/-100mV  [Record the value] |  |
| g) | Disconnect the analyser and reset the EUT. Monitor the CAN lines on a scope and verify there is continuous activity | Continuous activity on CAN lines |  |
| Notes and observations: | | | |

# UART Tests

## Overview

This section uses the Test Firmware to verify the operation of the UARTs.

## Test Setup

Setup a bench PSU, set to 5V, with **current limit set to approx. 0.3A**, **turned off**. Connect the PSU to the EUT via a (modified) USB cable. Turn on the power supply.

Start a Terminal Emulator app on the PC that will communicate via the UART on the ST-Link adaptor. Select ST-Link channel and set the serial parameters to: 8-N-1-115200.

Setup a separate serial adaptor to allow interface with the UARTs on the EUR (3V3 signal levels). Parameters: 8-N-1-115200.

**NB: Current version of the firmware does not work if additional characters are sent (e.g. between tests, or before the test starts). If this happens then reset the EUT.**

## Procedure

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Run the test for the UART 4 | - | n/a |
| 2 | Send a character to the EUT. Verify that the character is reflected back within 5ms. | Character is reflected back within 5ms |  |
| 3 | Use a scope to check the following: | - | n/a |
| a) | Verify TX line high level. | 3.0 to 3.3V  [Record the value] |  |
| b) | Verify TX line low level. | 0 to 0.6V  [Record the value] |  |
| c) | Verify TX line rise time (20-80%). | <100ns  [Record the value] |  |
| d) | Verify TX line fall time (20-80%). | <100ns  [Record the value] |  |
| e) | Use a scope to monitor the TX and RX lines on the UART 2 and verify there is no cross-talk. | No cross talk |  |
| 4 | Run the test for the UART 2 | - | n/a |
| 5 | Send a character to the EUT. Verify that the character is reflected back within 5ms. | Character is reflected back within 5ms |  |
| 6 | Use a scope to check the following: | - | n/a |
| a) | Verify TX line high level | 3.0 to 3.3V  [Record the value] |  |
| b) | Verify TX line low level | 0 to 0.6V  [Record the value] |  |
| c) | Verify TX line rise time (20-80%) | <100ns  [Record the value] |  |
| d) | Verify TX line fall time (20-80%) | <100ns  [Record the value] |  |
| e) | Use a scope to monitor the TX and RX lines on the UART 4 and verify there is no cross-talk. | No cross talk |  |
| Notes and observations: | | | |

# GPIO Tests

## Overview

This section uses the Test Firmware to verify the GPIO Lines. NB: GPIO outputs DO6 and DO7 are not tested here, because they are used for the clock frequency tests.

## Test Setup

Setup a bench PSU, set to 5V, with **current limit set to approx. 0.3A**, **turned off**. Connect the PSU to the EUT via a (modified) USB cable. Turn on the power supply.

Start a Terminal Emulator app on the PC that will communicate via the UART on the ST-Link adaptor. Select ST-Link channel and set the serial parameters to: 8-N-1-115200.

## Procedure

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | Run the test for the GPIO. | - |  |
| 2 | Use a scope to monitor the signal on GPIO0. Verify it normally sits at logic 0, but has a logic 1 pulse of approx. 27ms, when the GPIO test is run. | Nominally logic 0  Goes to logic 1 for ~27ms when GPIO test is run |  |
| 3 | Use GPIO0 as a trigger / reference.  Connect a second scope channel to monitor each of the following pins, in turn. For each one, run the test and verify a pulse appears of the same width, but later in time to the previous GPIO line, so no 2 pulses occupy the same time. Note: the pulses will be approx. 50ms apart. | - | n/a |
| a) | GPIO1 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| b) | GPIO2 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| c) | GPIO3 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| d) | GPIO4 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| e) | GPIO5 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| f) | NB: GPIO6&7 are tested elsewhere | - |  |
| g) | GPIO8 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| h) | GPIO9 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| i) | GPIO10 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| j) | GPIO11 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| k) | GPIO12 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| l) | GPIO13 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| m) | GPIO14 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| n) | GPIO15 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| o) | GPIO16 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| p) | GPIO17 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| q) | GPIO18 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| r) | GPIO19 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| s) | GPIO20 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| t) | GPIO21 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| u) | GPIO22 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| v) | GPIO23 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| w) | GPIO24 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| x) | GPIO25 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| y) | GPIO26 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| z) | GPIO26 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| aa) | GPIO27 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| bb) | GPIO28 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| cc) | GPIO29 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| dd) | GPIO30 | Nominally logic 0  Goes to logic 1 for ~27ms when the GPIO test is run  Doesn’t occur at the same time as any other GPIO logic 1 |  |
| Notes and observations: | | | |

# Thermistor Channel Tests

## Overview

This section uses the Test Firmware to verify the operation of the thermistor channels. The MCU lines usually operate as AtoD convertors, but for this test those lines are configured as GPIO outputs and pulses are sent out.

## Test Setup

Setup a bench PSU, set to 5V, with **current limit set to approx. 0.3A**, **turned off**. Connect the PSU to the EUT via a (modified) USB cable. Turn on the power supply.

Start a Terminal Emulator app on the PC that will communicate via the UART on the ST-Link adaptor. Select ST-Link channel and set the serial parameters to: 8-N-1-115200.

## Procedure

|  |  |
| --- | --- |
| Time & Date Undertaken: |  |
| Operator: |  |

| # | Test | Pass Criteria | Result |
| --- | --- | --- | --- |
| 1 | For each step below, temporarily connect a 10K resistor between the channel and its adjacent ground. Hold the reset button down and check the voltage across the resistor: | - | n/a |
| a) | Verify voltage at THM0. | 1.55 to 1.75V  [Record the value] |  |
| b) | Verify voltage at THM1. | 1.55 to 1.75V  [Record the value] |  |
| c) | Verify voltage at THM2. | 1.55 to 1.75V  [Record the value] |  |
| d) | Verify voltage at THM3. | 1.55 to 1.75V  [Record the value] |  |
| e) | Verify voltage at THM4. | 1.55 to 1.75V  [Record the value] |  |
| f) | Verify voltage at THM5. | 1.55 to 1.75V  [Record the value] |  |
| g) | Verify voltage at THM6. | 1.55 to 1.75V  [Record the value] |  |
| h) | Verify voltage at THM7. | 1.55 to 1.75V  [Record the value] |  |
| 2 | With the EUT powered up normally (no test running), check the voltage across the following channels, wrt DGND: | - | n/a |
| a) | Verify voltage at THM0. | 1.6 to 1.8V  [Record the value] |  |
| b) | Verify voltage at THM1. | 1.6 to 1.8V  [Record the value] |  |
| c) | Verify voltage at THM2. | 1.6 to 1.8V  [Record the value] |  |
| d) | Verify voltage at THM3. | 1.6 to 1.8V  [Record the value] |  |
| e) | Verify voltage at THM4. | 1.6 to 1.8V  [Record the value] |  |
| f) | Verify voltage at THM5. | 1.6 to 1.8V  [Record the value] |  |
| g) | Verify voltage at THM6. | 1.6 to 1.8V  [Record the value] |  |
| h) | Verify voltage at THM7. | 1.6 to 1.8V  [Record the value] |  |
| 3 | Use a scope to monitor the signal on THM0. Run the ‘Analogue’ test. Verify it has a high pulse >3V for approx. 0.25sec, when the test is run. | THM0 pulse >3V for approx. 0.25s |  |
| 4 | Use THM0 as a trigger / reference.  Connect a second scope channel to monitor each of the following J3 pins, in turn. For each one, run the test and verify a >3V pulse appears of the same width but in a different position, so no 2 pulses occupy the same time. | - | n/a |
| a) | Verify correct pulse on THM1. | THM1 pulse >3V for approx. 0.25s  No pulse on another line at the same time |  |
| b) | Verify correct pulse on THM2. | THM2 pulse >3V for approx. 0.25s  No pulse on another line at the same time |  |
| c) | Verify correct pulse on THM3. | THM3 pulse >3V for approx. 0.25s  No pulse on another line at the same time |  |
| d) | Verify correct pulse on THM4. | THM4 pulse >3V for approx. 0.25s  No pulse on another line at the same time |  |
| e) | Verify correct pulse on THM5. | THM5 pulse >3V for approx. 0.25s  No pulse on another line at the same time |  |
| f) | Verify correct pulse on THM6. | THM6 pulse >3V for approx. 0.25s  No pulse on another line at the same time |  |
| h) | Verify correct pulse on THM7. | THM7 pulse >3V for approx. 0.25s  No pulse on another line at the same time |  |
| 5 | Repeat the previous test, but check the rise and fall times for each channel: | - | n/a |
| a) | Verify rise and fall time (20/80%) at THM0. | 1 to 2ms  [Record the value] |  |
| b) | Verify rise and fall time (20/80%) at THM1. | 1 to 2ms  [Record the value] |  |
| c) | Verify rise and fall time (20/80%) at THM2. | 1 to 2ms  [Record the value] |  |
| d) | Verify rise and fall time (20/80%) at THM3. | 1 to 2ms  [Record the value] |  |
| e) | Verify rise and fall time (20/80%) at THM4. | 1 to 2ms  [Record the value] |  |
| f) | Verify rise and fall time (20/80%) at THM5. | 1 to 2ms  [Record the value] |  |
| h) | Verify rise and fall time (20/80%) at THM6. | 1 to 2ms  [Record the value] |  |
| i) | Verify rise and fall time (20/80%) at THM7. | 1 to 2ms  [Record the value] |  |
| Notes and observations: | | | |

# Test Report Sheet

## EUT Details

|  |  |
| --- | --- |
| Schematic |  |
| PCB Number |  |
| Serial Number |  |
| Firmware part number |  |
| Firmware Version |  |
| Board Status & Modifications |  |

## Equipment used:

Equipment used for testing shall be recorded below. Any uncalibrated equipment should be recorded as ‘FIO’ in the Cal Due Date column.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Mfr & Part no | S/N | Calibration Date | Notes |
| Laptop |  |  |  |  |
| Can bus analyser |  |  |  |  |
| USB to serial interface |  |  |  |  |
| DVM |  |  |  |  |
| Oscilloscope |  |  |  |  |
| Bench PSU |  |  |  |  |
| Function Generator |  |  |  |  |

# Software Apps Used

Software apps used for testing shall be recorded. (No need to include Microsoft items).

|  |  |  |  |
| --- | --- | --- | --- |
| Description | Mfr / Identification | Version | Notes |
| Terminal emulator |  |  |  |
| Development tool |  |  |  |
| Can bus analyzer app |  |  |  |
| Driver for USB-Serial |  |  |  |

# Acceptance

|  |  |
| --- | --- |
| Results Approved By |  |
| Date |  |

A screenshot of a cell phone

Description automatically generated