|  |
| --- |
| Quantum-PCI Quantum PCIe  Программа тестирования и испытаний |

Содержание

Сводка тестовых элементов и результатов

Введение 10

Проверка короткого замыкания источника питания

Проверка выходного напряжения источника питания 12

Проверка последовательности включения 13

5.1 Требования к последовательности включения. 13

5.2 Результат тестирования последовательности включения. 13

Измерение уровня напряжения и пульсаций на шинах питания устройств 16

6.1 Измерение напряжения на шинах питания устройств. 16

6.1.1 Измерение напряжения на шинах питания FPGA. 16

6.1.2 Измерение напряжения на шинах питания модуля MAC. 16

6.1.3 Измерение напряжения на шинах питания модуля GPS. 17

6.2 Измерение пульсаций напряжения на шинах питания устройств 17

6.2.1 Измерение пульсаций напряжения на шинах питания FPGA 17

6.2.2 Измерение пульсаций напряжения на шинах питания модуля MAC 19

6.2.3 Измерение пульсаций напряжения на шинах питания модуля GPS 20

Тестирование выходных сигналов генератора тактовых импульсов 22

7.1 Тестирование выходных сигналов генератора тактовых импульсов 125МГц LVDS 22

7.2 Тестирование выходных сигналов генератора тактовых импульсов 200МГц LVDS 23

Предварительное тестирование 25

8.1 Модуль GPS 25

8.1.1 Интерфейс UART модуля GPS 25

8.1.2 Выходные сигналы 1PPS модуля GPS 25

8.2 Модуль MAC 26

8.2.1 Интерфейс UART модуля MAC 26

8.2.2 Выходной сигнал 1PPS модуля MAC 27

8.2.3 Выходной сигнал RF (10МГц) модуля MAC. 28

8.2.4 USB интерфейсы модуля MAC. 29

8.3 Датчики 30

8.3.1 I2C интерфейс для всех датчиков. 30

8.4 Устройства EEPROM 33

8.4.1 I2C интерфейс для устройства EEPROM. 33

8.4.2 I2C интерфейс для второго устройства EEPROM . 34

8.5 Драйвер RGB светодиодов и RGB светодиод 35

8.5.1 I2C интерфейс для драйвера RGB светодиодов . 35

8.5.2 Тестирование RGB светодиодов . 35

8.6 FPGA 36

8.6.1 Загрузка образа FPGA в SPI флеш-память через JTAG 36

8.6.2 Тестирование PCIe интерфейса FPGA 37

8.7 Результат измерения Tr, Tf для тактовых сигналов и данных I2C шины. 38

8.8 Просадка напряжения на выводе VCC буферного микрочипа для сигналов FPGA<>SMA I/O. 41

8.9 FT4232 USB to UART/MPSSE . 44

8.9.1 Тестирование интерфейса USB2.0 для FT4232 . 44

8.9.2 FT4232: Тестирование двигателя MPSSE канала A для JTAG интерфейса . 45

8.9.3 FT4232: Тестирование двигателя MPSSE канала A для SPI интерфейса . 45

8.9.4 FT4232: Тестирование двигателя MPSSE канала B для I2C интерфейса . 46

8.9.5 FT4232: Тестирование UART интерфейса канала C . 47

8.9.6 FT4232: Тестирование UART интерфейса канала D . 48

8.10 Функция программируемой задержки времени U34 49

8.11 RC-завершение для уменьшения отражения сигнала. 50

8.11.1 R-завершение для 10МГц от модуля MAC к FPGA . 50

8.11.2 RC-завершение для 1PPS от FPGA к буферному микрочипу 52

Функциональное тестирование 57

9.1 Тестирование PCIe интерфейсов FPGA 57

9.2 Тестирование интерфейса I2C-мастер FPGA <> I2C-устройства 58

9.2.1 Тестирование интерфейса I2C-мастер FPGA <> I2C-датчики 58

9.2.2 Тестирование интерфейса I2C-мастер FPGA <> I2C-устройства EEPROM 60

9.3 Тестирование драйвера светодиодов FPGA для RGB\_LED 61

9.4 Интерфейс ядра UART FPGA с модулями GPS и MAC 63

9.4.1 Тестирование интерфейсов UART FPGA<> GPS 63

9.4.2 Тестирование интерфейсов UART FPGA<> MAC 64

9.5 Тестирование PTP с одним клиентом с помощью команды phc2sys 65

9.6 Тестирование PTP с одним клиентом с помощью команды phc2sys 66

9.7 Тестирование PTP с одним клиентом с помощью команды ts2phc 68

9.8 Тестирование PTP с одним клиентом с помощью команды ts2phc 69

9.9 Тестирование 24-часового удержания 1PPS 71

9.10 Тестирование 24-часового удержания 1PPS 72

9.11 Тестирование загрузки нескольких клиентов PTP. 73

9.12 Тестирование задержки одного PCIE. 73

Список таблиц:

Таблица 1: История версий 7

Таблица 2: Справочная документация 7

Таблица 3: Сводка тестовых элементов и результатов 8

Таблица 4: Результаты проверки короткого замыкания источника питания. 11

Таблица 5: Результаты проверки выходного напряжения источника питания. 12

Таблица 6: Результаты проверки последовательности включения. 13

Таблица 7: Измеренное напряжение на шинах питания FPGA. 16

Таблица 8: Измеренное напряжение на шинах питания модуля MAC. 17

Таблица 9: Измеренное напряжение на шинах питания модуля GPS. 17

Таблица 10: Измеренные пульсации напряжения на шинах питания FPGA. 19

Таблица 11: Измеренные пульсации напряжения на шинах питания модуля MAC. 20

Таблица 12: Измеренные пульсации напряжения на шинах питания модуля GPS. 21

Таблица 13: Тестирование выходных сигналов генератора тактовых импульсов 125МГц LVDS. 23

Таблица 14: Тестирование выходных сигналов генератора тактовых импульсов 200МГц LVDS. 24

Список рисунков:

Рисунок 1: Последовательность включения. 13

Рисунок 2: Спецификация напряжения на шинах питания FPGA. 16

Рисунок 3: Характеристики выхода LVDS. 22

Рисунок 4: Схема подключения I2C для датчиков. 30

Рисунок 5: Схема подключения I2C для устройства EEPROM. 34  
Рисунок 6: Спецификация шины I2C. 39  
Рисунок 7: Интерфейс I2C-мастер FPGA <> I2C-датчики. 58  
Рисунок 8: Интерфейс I2C-мастер FPGA <> I2C-устройства EEPROM. 60

**История версий платы**

**Таблица 1: История версий**

| **Версия** | **Дата** | **Автор** | **Проверил** |
| --- | --- | --- | --- |
|  |  |  |  |

**Ссылки на документы**

|  |  |
| --- | --- |
| **№ п/п** | **Документ** |
| 1 | FPGA Datasheet: “Artix‐7 FPGAs Data Sheet: DC and AC Switching Characteristics”, DS181 (v1.26) March 23, 2021 |
| 2 | PCIE Spec: “PCI Express® Base Specification”, Revision 2.1, March 4, 2009 |
| 3 | “BNO080/85/86 Data Sheet”, Revision 1.11 |
| 4 | “Atmel-8807E-SEEPROM-AT24MAC402-602-Datasheet\_012015” |
| 5 | “PCA9546A 4-channel I2C-bus switch with reset Product data sheet”, Rev. 6 — 30 April 2014 |
| 6 | “LM75B Digital temperature sensor and thermal watchdog Product data sheet”, Rev. 6.1 — 6 February 2015 |
| 7 | “SENSIRION SHT3xA-DIS Datasheet”, December 2019 - Version 7 |
| 8 | “PCA9617A Level translating Fm+ I2C-bus repeater Product data sheet”, Rev. 1 — 20 March 2013 |
| 9 | “TDK ICP-10100, ICP-10101, ICP-10110, ICP-10111 Datasheet”, Document Number: DS-000186, Revision: 1.2, Release Date: 05/06/2019 |
| 10 | “Micron Serial NOR Flash Memory 3V, Multiple I/O, 4KB, 32KB, 64KB, Sector Erase  MT25QL128ABA Datasheet”, Rev. C 4/15 EN |
| 11 | “PCI Express™® Card Electromechanical Specification”, Revision 2.0, April 11, 2007 |
| 12 | “TEXAS INSTRUMENTS TS3A5018 SCDS189H-JANUARY 2005-REVISED MAY 2018” |
| 13 | “PCA9617A Level translating Fm+I2C-bus repeater Rev.1-20 March 2013” |
| 14 | “FTDI Chip FT4232H QUAD HIGH SPEED USB TO MULTIPURPOSE USRT/MPSSE IC Datasheet Version 2.4 Document No.:FT\_000060 Clearance No.:FTDI#78” |
| 15 | “MICROCHIP 24AA64/24LC64/24FC64 64K I2C Serial EEPROM DS21189Q” |
| 16 | “IS32FL3207 18-CHANNEL LED DRIVER WITH 16-BIT PWM Rev. A,05/12/2020” |
| 17 | AT24MAC402 and AT24MAC602 I2C-Compatible (2-wire) 2-Kbit Serial EEPROM with a Factory-Programmed EUI-48™ or EUI-64™Address Plus an Embedded Unique 128-bit Serial Number 2-Kbit (256 x 8) DATASHEET |
|  |  |

1. Перечень тестов и испытаний

**Таблица 3**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Тестовый элемент** | | | **Описание теста** | | | **Результат**  **(Успешно/**  **Неудача)** |
| 3 | Проверка короткого замыкания источника питания | | | | |  |
| 4 | Проверка выходного напряжения источника питания | | | | |  |
| 5 | Проверка последовательности включения | | | | |  |
|  | 5.1 | Требования к последовательности включения. | | | | - |
|  | 5.2 | Результаты теста последовательности включения. | | | |  |
| 6 | Измерение уровня напряжения и пульсаций на шинах питания устройств | | | | | |
|  | 6.1 | Измерение напряжения на шинах питания устройств. | | | | - |
|  |  | 6.1.1 | Измерение напряжения на шинах питания FPGA. | | |  |
|  |  | 6.1.2 | Измерение напряжения на шинах питания модуля MAC. | | |  |
|  |  | 6.1.3 | Измерение напряжения на шинах питания модуля GPS. | | |  |
|  | 6.2 | Измерение пульсаций напряжения на шинах питания устройств. | | | |  |
|  |  | 6.2.1 | Измерение пульсаций напряжения на шинах питания FPGA. | | |  |
|  |  | 6.2.2 | Измерение пульсаций напряжения на шинах питания модуля MAC. | | |  |
|  |  | 6.2.3 | Измерение пульсаций напряжения на шинах питания модуля GPS. | | |  |
| 7 | Тестирование выходных сигналов генератора тактовых импульсов | | | | |  |
|  | 7.1 | Тестирование выходных сигналов генератора тактовых импульсов 125МГц LVDS. | | | |  |
|  | 7.2 | Тестирование выходных сигналов генератора тактовых импульсов 200МГц LVDS. | | | |  |
| 8 | **Предварительное тестирование** | | | | |  |
|  | 8.1 | GPS модуль | | | |  |
|  |  | 8.1.1 | Интерфейс UART модуля GPS. | | |  |
|  |  | 8.1.2 | Выходные сигналы 1PPS модуля GPS. | | |  |
|  | 8.2 | MAC модуль | | | |  |
|  |  | 8.2.1 | Интерфейс UART модуля MAC. | | |  |
|  |  | 8.2.2 | Выходной сигнал 1PPS модуля MAC. | | |  |
|  |  | 8.2.3 | Выходной сигнал RF модуля MAC. | | |  |
|  |  | 8.2.4 | USB интерфейсы модуля MAC. | | |  |
|  | 8.3 | Датчики | | | |  |
|  |  | 8.3.1 | I2C интерфейс для всех датчиков. | | |  |
|  | 8.4 | EEPROM | | | |  |
|  |  | 8.4.1 | I2C интерфейс для устройств EEPROM. | | |  |
|  | 8.5 | Драйвер RGB светодиодов и RGB светодиод | | | |  |
|  |  | 8.5.1 | | I2C интерфейс для драйвера RGB светодиодов. | |  |
|  |  | 8.5.2 | | Тестирование RGB светодиодов. | |  |
|  | 8.6 | FPGA | | | |  |
|  |  | 8.5.1 | Загрузка образа FPGA в SPI флеш-память через JTAG. | | |  |
|  |  | 8.5.2 | Тестирование PCIe интерфейса FPGA (Требуется образ FPGA для поддержки теста PCIe). | | |  |
|  | 8.7 | Результаты измерения Tr/Tf для тактовых сигналов и данных I2C шины. | | | |  |
|  | 8.8 | Просадка напряжения на выводе VCC буферного микрочипа для сигналов FPGA<>SMA I/O. | | | |  |
|  | 8.9 | FT4232 USB to UART/MPSSE | | | |  |
|  |  | 8.9.1 | Тестирование интерфейса USB2.0 для FT4232 | | |  |
|  |  | 8.9.2 | FT4232: Тестирование двигателя MPSSE канала A для JTAG интерфейса | | |  |
|  |  | 8.9.3 | FT4232: Тестирование двигателя MPSSE канала A для SPI интерфейса | | |  |
|  |  | 8.9.4 | FT4232: Тестирование двигателя MPSSE канала B для I2C интерфейса | | |  |
|  |  | 8.9.5 | FT4232: Тестирование UART интерфейса канала C | | |  |
|  |  | 8.9.6 | FT4232: Тестирование UART интерфейса канала D | | |  |
|  | 8.10 | Функция программируемой задержки времени U34 | | | |  |
|  | 8.11 | RC-завершение для уменьшения отражения сигнала. | | | |  |
|  |  | 8.11.1 | | | R-завершение для 10МГц от модуля MAC к FPGA |  |
|  |  | 8.11.2 | | | RC-завершение для 1PPS от FPGA к буферному микрочипу |  |
| 9 | Функциональное тестирование | | | | |  |
|  | 9.1 | Тестирование PCIe интерфейсов FPGA. | | | |  |
|  | 9.2 | Тестирование интерфейса I2C-мастер FPGA <> I2C-устройства. | | | |  |
|  |  | 9.2.1 | Тестирование интерфейса I2C-мастер FPGA <> I2C-датчики. | | |  |
|  |  | 9.2.2 | Тестирование интерфейса I2C-мастер FPGA <> I2C-устройства EEPROM. | | |  |
|  | 9.3 | Тестирование драйвера светодиодов FPGA для USR\_LED и RGB\_LED. | | | |  |
|  | 9.4 | Интерфейс ядра UART FPGA с модулями GPS и MAC. | | | |  |
|  |  | 9.4.1 | Тестирование интерфейсов UART FPGA<> GPS. | | |  |
|  |  | 9.4.2 | Тестирование интерфейсов UART FPGA<> MAC. | | |  |
|  | 9.5 | Тестирование PTP с одним клиентом в течение 24 часов с помощью команды phc2sys. | | | |  |
|  | 9.6 | Тестирование PTP с одним клиентом с помощью команды phc2sys | | | |  |
|  | 9.7 | Тестирование PTP с одним клиентом в течение 24 часов с помощью команды ts2phc. | | | |  |
|  | 9.8 | Тестирование PTP с одним клиентом с помощью команды ts2phc | | | |  |
|  | 9.9 | Тестирование 24-часового удержания 1PPS | | | |  |
|  | 9.10 | Тестирование 24-часового удержания 1PPS | | | |  |
|  | 9.11 | Тестирование загрузки нескольких клиентов PTP | | | |  |
|  | 9.12 | Тестирование задержки одного PCIE. | | | |  |

**Введение**

Этот документ описывает этапы запуска, аппаратного тестирования и функционального тестирования изделия Quantum-PCI.

Документ включает в себя этапы запуска, предварительного тестирования и функционального тестирования.

Предварительное тестирование включает проверку короткого замыкания источника питания, уровня напряжения, пульсаций и последовательности напряжения на шинах питания. Также проводится предварительное тестирование выходных сигналов генератора тактовых импульсов.

Предварительные сигнальные тесты для интерфейсов PCIe, SPI, I2C и GPIO для каждого устройства на плате проводятся, чтобы убедиться и подтвердить, что все узлы и функции работают нормально.

**2 Проверка короткого замыкания источника питания**

Используйте цифровой измерительный прибор для измерения сопротивления между контрольной точкой и SG. Все источники питания не должны иметь короткого замыкания на SG.

**Таблица 4: Результат проверки на короткое замыкание источника питания.**

| **Тестовый элемент** | **Контрольная точка +** | **Контрольная точка -** | **Результат измерения** | **Результат** |
| --- | --- | --- | --- | --- |
| XP12R0V | TP52 | GND | >1M | Pass |
| XP3R3V | TP53 | GND | 1.86k | Pass |
| XP1R0V\_FPGA | TP54 | GND | 1.18k | Pass |
| XP1R2V\_FPGA | TP55 | GND | 3.36k | Pass |
| XP1R8V\_FPGA | TP56 | GND | 1.28k | Pass |
| XP2R5V\_FPGA | TP57 | GND | 3.08k | Pass |
| XP3R3V\_FPGA | TP58 | GND | 5.06k | Pass |
| XP5R0V | TP60 | GND | 25.32k | Pass |

**3 Проверка выходного напряжения источника питания**

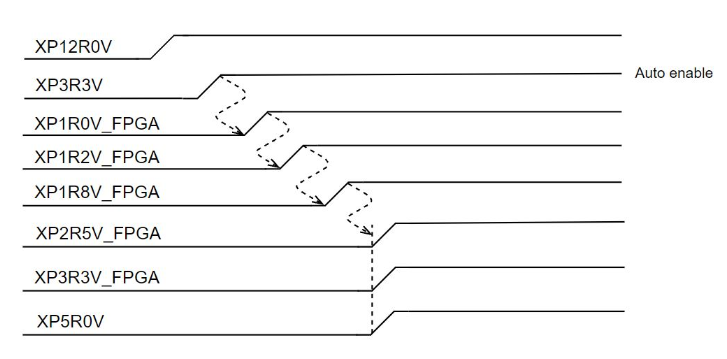
Используйте цифровой измерительный прибор для измерения напряжения на каждой шине питания платы.

Таблица 5: Результаты проверки выходного напряжения источника питания.

| **Тестовый элемент** | **Контрольная точка +** | **Контрольная точка -** | **Результат измерения (V)** | **Критерий** | **Результат** |
| --- | --- | --- | --- | --- | --- |
| XP12R0V | TP52 | GND |  | 11.4V ~ 12.6V | Pass |
| XP3R3V | TP53 | GND |  | 3.135V ~ 3.465V | Pass |
| XP1R0V\_FPGA | TP54 | GND |  | 0.97V ~ 1.03V | Pass |
| XP1R2V\_FPGA | TP55 | GND |  | 1.14 ~ 1.26V | Pass |
| XP1R8V\_FPGA | TP56 | GND |  | 1.71V ~ 1.89V | Pass |
| XP2R5V\_FPGA | TP57 | GND |  | 2.375V ~ 2.625V | Pass |
| XP3R3V\_FPGA | TP58 | GND |  | 3.135V ~ 3.465V | Pass |
| XP5R0V | TP60 | GND |  | 4.75V ~ 5.25V | Pass |

**4 Проверка последовательности включения питания**

**4.1 Требования к последовательности включения питания.**



**Рисунок 1: Последовательность включения питания.**

**4.2 Результат проверки последовательности включения питания.**

Таблица 6: Результаты проверки последовательности включения питания.

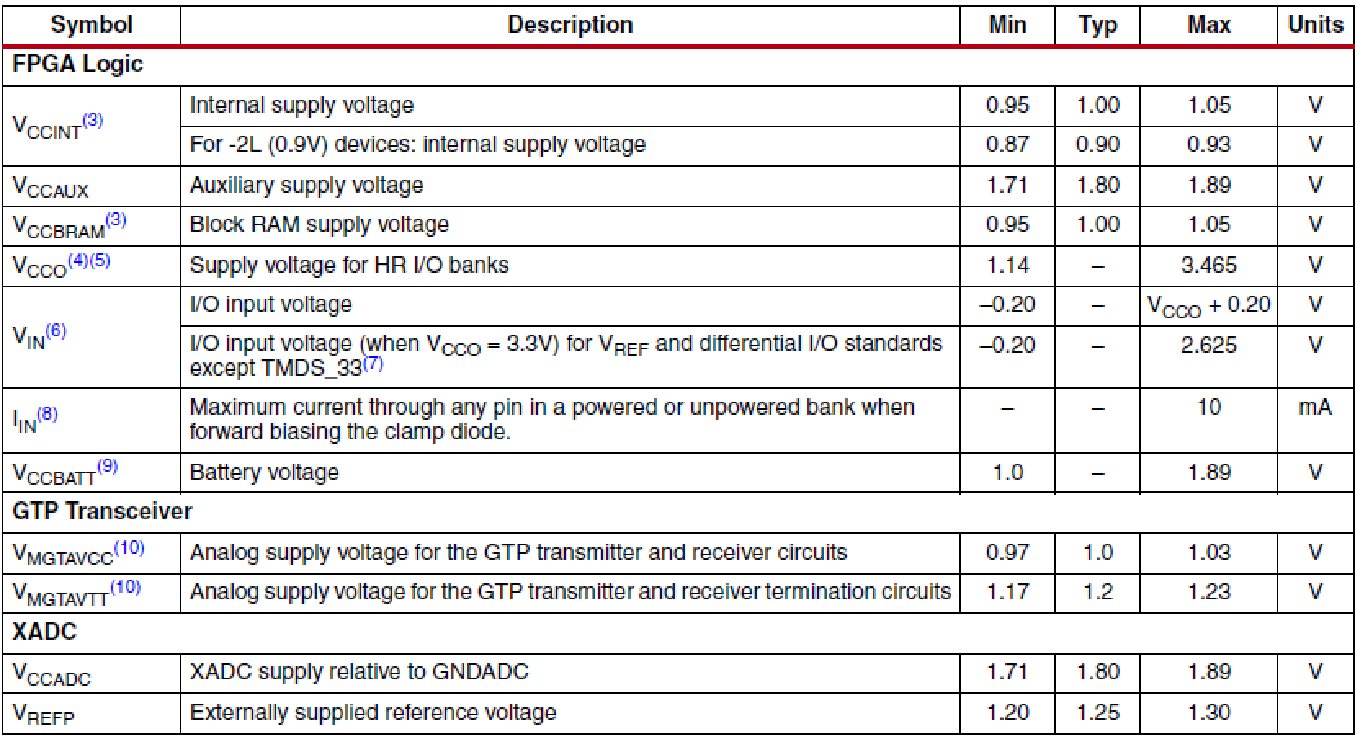
|  |  |  |
| --- | --- | --- |
| **Силовая шина** | **Форма волны** | **Положительно/отрицательно** |
| XP12R0V > XP3R3V  CH1= XP12R0V  CH2= XP3R3V |  | Pass. |
| XP3R3V > XP1R0V \_FPGA  CH1= XP3R3V  CH2= XP1R0V \_FPGA |  | Pass. |
| XP1R0V \_FPGA > XP1R2V\_FPGA  CH1= XP1R0V \_FPGA  CH2= XP1R2V\_FPGA |  | Pass. |
| XP1R2V \_FPGA > XP1R8V\_FPGA  CH1= XP1R2V \_FPGA  CH2= XP1R8V\_FPGA |  | Pass. |
| XP1R8V\_FPGA, > XP2R5V\_FPGA, XP3R3V\_FPGA, XP5R0V  CH1= XP1R8V\_FPGA  CH2= XP2R5V\_FPGA  CH3= XP3R3V\_FPGA  CH4= XP5R0V |  | Pass. |

**5 Измерение уровня напряжения на шинах питания и пульсаций напряжения на узлах**

Напряжения должны соответствовать требованиям к питанию каждого чипа или модуля.

**4.3 Измерение напряжения на шинах питания устройств.**

**4.3.1 Измерение напряжения на шинах питания ПЛИС.**



**Рисунок 2: Спецификация напряжения на силовой шине ПЛИС.**

Таблица 7: Напряжение на шине питания, измеренное на ПЛИС.

| **Шина** | **Тестовая точка** | **Критерий** | **Результат** | **Да/Нет** |
| --- | --- | --- | --- | --- |
| XP1R0V\_FPGA\_VCCINT | Under ASIC | 0.97V ~ 1.03V | 1.009 |  |
| XP1R0V\_FPGA\_MGTAVCC | Under ASIC | 0.97V ~ 1.03V | 1.009 |  |
| FPGA\_MGTAVTT | Under ASIC | 1.17 ~ 1.23V | 1.206 |  |
| XP1R8V\_FPGA\_VCCAUX | Under ASIC | 1.71V ~ 1.89V | 1.796 |  |
| XP2R5V\_FPGA | Under ASIC | 2.375V ~ 2.625V | 2.514 |  |
| XP3R3V\_FPGA | Under ASIC | 3.135V ~ 3.465V | 3.324 |  |

**4.3.2 Напряжение на силовой шине, измеренное на модуле MAC.**

Таблица 8: Напряжение на шине питания, измеренное на модуле MAC.

| **Шина** | **Тестовая точка** | **Критерий** | **Результат** | **Да/Нет** |
| --- | --- | --- | --- | --- |
| XP5R0V\_MAC\_USB\_POWER | J31: pin6 | 4.75V ~ 5.25V | 5.038 |  |
| XP5R0V\_MAC\_VCC | MAC\_PIN5 | 4.5V ~ 32V | 5.03V |  |

**Напряжение на силовой шине, измеренное на модуле GPS**

Таблица 9: Напряжение на силовой шине, измеренное на модуле GPS.

| **Шина** | **Тестовая точка** | **Критерий** | **Результат** | **Да/Нет** |
| --- | --- | --- | --- | --- |
| XP5R0V\_VCC\_ANT | P1: J6 и P1:J13 | 4.5V ~ 5.5V | 5.05V |  |
| XP3R3V | P2: J6 и P2:J13 | 2.7V ~ 3.6V | 3.321V |  |

**4.3 Измерение пульсаций напряжения на силовой шине на устройствах**

Таблица 10: Пульсации напряжения на силовой шине, измеренные на ПЛИС.

|  |  |  |
| --- | --- | --- |
| **Узел** | **Форма сигнала** | **Да/Нет** |
| XP1R8V\_FPGA\_VCCAUX  Read from DMM = 1.796V  **Criteria:** V(ripple): < 100mVp-p |  | Pass. |
| XP1R0V\_FPGA\_VCCINT  Read from DMM = 1.009V  **Criteria:** V(ripple) < 100mVp-p |  |  |
| XP3R3V\_FPGA  Read from DMM = 3.324  **Criteria:** V(ripple) < 100mVp-p |  |  |

|  |  |  |
| --- | --- | --- |
| **Узел** | **Форма сигнала** | **Да/Нет** |
| XP1R0V\_FPGA\_MGTAVCC  Read from DMM = 1.009  **Criteria:** V(ripple) < 100mVp-p |  |  |
| FPGA\_MGTAVTT  Read from DMM = 1.206V  **Criteria:** V(ripple) < 60mVp-p |  |  |

**Пульсации напряжения на силовой шине, измеренные на модуле MAC**

Таблица 11: Пульсации напряжения на силовой шине, измеренные на модуле MAC

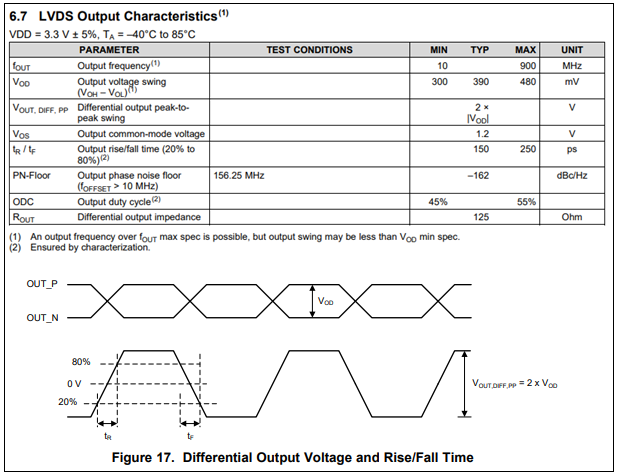
|  |  |  |
| --- | --- | --- |
| **Узел** | **Форма волны** | **Да/нет** |
| XP5R0V\_MAC\_USB\_POWER  Read from DMM = 5.038V  **Criteria:** V(ripple): < 100mVp-p |  |  |
| XP5R0V\_MAC  Read from DMM = 5.03V  **Criteria:** V(ripple) < 100mVp-p |  |  |

**Пульсации напряжения на силовой шине, измеренные на модуле GPS**

Таблица 12: Пульсации напряжения на силовой шине, измеренные на модуле GPS.

|  |  |  |
| --- | --- | --- |
| **Узел** | **Форма волны** | **Да/Нет** |
| XP5R0V\_VCC\_ANT  (P1: J6)  Read from DMM = 5.05V  **Критерий:** V(ripple): < 100mVp-p |  |  |
| XP3R3V (P2: J6)  Read from DMM = 3.321V  **Критерий:** V(ripple) < 100mVp-p |  |  |

**5 Проверка выходной мощности тактового генератора**



**Рисунок 3 - Характеристики выхода LVDS.**

* 1. Проверка выходного сигнала тактового генератора LVDS на частоте 125 МГц

**\* Это измерение выполняется с помощью несимметричного датчика. Для этого измерения мы просто проверяем выход тактовой частоты.**

Таблица 13: Проверка выходного сигнала LVDS тактового генератора на частоте 125 МГц.

|  |  |  |
| --- | --- | --- |
| **Сигнал** | **Форма волны** | **Да/Нет** |
| MHZ125CLKP\_CLKIN  MHZ125CLKN\_CLKIN | 125CLK\_P    //===================================  125CLK\_N    //=================================== |  |

4.2 Проверка выходного сигнала генератора тактовых импульсов 200 МГц LVDS

Для этого измерения мы просто проверяем выходной сигнал генератора тактовых импульсов.

**\* Это измерение выполняется с помощью несимметричного датчика. Для этого измерения мы просто проверяем выход тактовой частоты.**

Таблица 14: Проверка выходного сигнала LVDS тактового генератора на частоте 200 МГц.

|  |  |  |
| --- | --- | --- |
| **Сигнал** | **Форма сигнала** | **Да/Нет** |
| MHZ200CLKP\_CLKIN  MHZ200CLKN\_CLKIN | 200CLK\_P    //===================================  200CLK\_N |  |

8. **Предварительные тесты**

Предварительный тест, нам нужно запустить тест, чтобы убедиться, что все устройства, такие как GPS-модуль, MAC-модуль, ПЛИС, EEPROM и все сенсорные устройства, работают в обычном режиме.

4.5 Модуль GPS

4.5.1 Интерфейс UART модуля GPS

Требование: Протестируйте интерфейсы UART для модуля GPS, чтобы убедиться, что модуль GPS работает в обычном режиме.

Метод тестирования: (Тестирование проводится без загрузки встроенное ПО в FPGA для этого тестового примера)

1. Используйте кабель micro USB для подключения между ПК <> J43 и платой Quantum-PCIe.

2. Установите на SW1 перемычку для подключения контактов 1 и 8 (UART\_SEL = 0 для FTDI <-> GPS2, SW2 для FTDI <-> FPGA и FPGA <-> GPS2), а также 1,4 и 2,3 для подключения сигналов FT232 UART к GPS UART.

3. На главном КОМПЬЮТЕРЕ установите последовательный порт: скорость передачи данных = 115200 бод, бит данных = 8, бит контроля четности = нет, стоп-бит = 1, управление потоком данных = нет.

4. Откройте графический интерфейс U-blox (в Windows) для получения информации из модуля GPS.

Тестовый результат

|  |  |  |
| --- | --- | --- |
| **Тестовый узел** | **Результат** | **Да/Нет** |
| GNSS UART interface. |  | Pass |

4.4.1 Вывод сигналов GPS-модуля 1PPS

**Требование:** Проверьте сигналы 1PPS для модуля GPS, чтобы убедиться, что модуль GPS работает в обычном режиме.

**Метод проверки:** Используйте осциллограф для регистрации сигналов 1PPS.

Выполните измерение на выводе TP1 модуля GNSS.

Результат проверки.

|  |  |  |
| --- | --- | --- |
| **Тестовый элемент** | **Результат** | **Да/Нет** |
| Выходной сигнал 1PPS поступает с контакта TP1 модуля GNSS. |  |  |

* 1. MAC модуль
     1. Интерфейс UART модуля MAC

**Требование:** Протестируйте интерфейсы UART для модуля MAC, чтобы убедиться, что модуль MAC работает в обычном режиме.

**Метод тестирования:** (Не загружайте встроенное ПО в FPGA для этого теста)

1. Используйте кабель micro USB для подключения между ПК <> J43 и платой.

2. На разъеме SW1 установите перемычку 6,3 в положение Off, чтобы подключить FT232 UART к MAC UART.

3. На главном компьютере установите последовательный порт: скорость передачи данных = 57600 бод, бит данных = 8, бит контроля четности = нет, стоп-бит = 1, управление потоком = нет.

4. Протестируйте с помощью интерфейса командной строки, а затем запишите журнал последовательного вывода из модуля MAC.

**Тестовый результат**

|  |  |  |
| --- | --- | --- |
| **Тестовый элемент** | **Результат** | **Положит/Отриц** |
| Протестируйте интерфейс UART для модуля MAC |  | Pass. |

* + 1. Выходной сигнал модуля MAC 1PPS

**Требование:** Проверьте сигналы 1PPS для MAC-модуля, чтобы убедиться, что модуль MAC работает в обычном режиме.

Метод проверки: Используйте осциллограф для регистрации сигналов 1PPS.

\* Это измерение выполняется с помощью несимметричного датчика. Это измерение просто подтверждает вывод тактовой частоты.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| Выходной сигнал 1PPS от модуля MAC  Пин P10 MAC\_PPS\_OUTP | Pulse width: FPGA\_IN\_MAC\_PPS\_OUTP    Period: FPGA\_IN\_MAC\_PPS\_OUTP    //===================================  Pulse width: FPGA\_IN\_MAC\_PPS\_OUTN    Period: FPGA\_IN\_MAC\_PPS\_OUTN | Pass. |

* + 1. Радиочастотный выходной сигнал модуля MAC (10 МГц).

**Требование:** Проверьте выходной сигнал MAC-модуля на частоте 10 МГц, чтобы убедиться, что он работает в обычном режиме.

Метод проверки: с помощью осциллографа зафиксируйте выходной сигнал на частоте 10 МГц.

Результат проверки.

|  |  |  |
| --- | --- | --- |
| **Тестовый сигнал** | **Результат** | **Да/Нет** |
| Выходной сигнал 10 МГц от модуля MAC |  | Pass. |

4.5.4 USB-интерфейсы модуля MAC.

**Требование:** Протестируйте USB-интерфейсы для модуля MAC, чтобы убедиться, что модуль MAC работает в обычном режиме.

Test method:

1. Используйте кабель micro USB для подключения между ПК <> J43 и платой.

2. На главном компьютере запустите ОС Linux, а затем с помощью команды “lsusb -v” введите список USB-устройств.

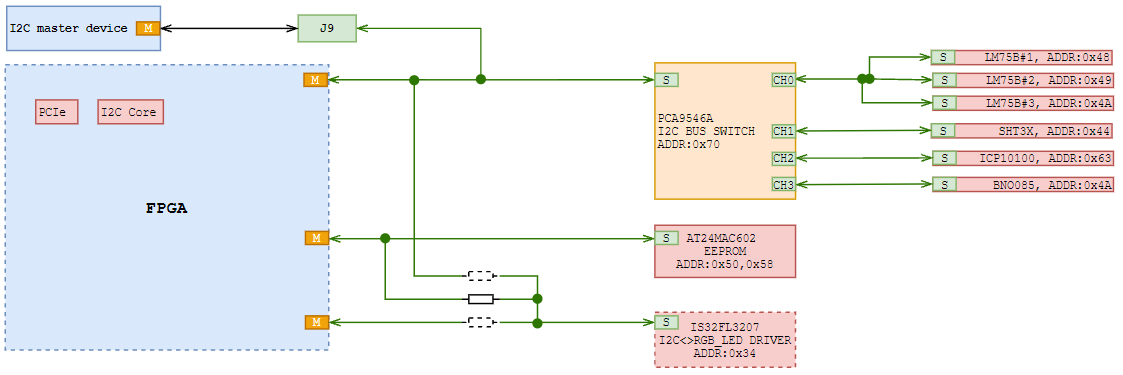
3. Проверьте выводимое сообщение, операционная система должна обнаружить USB-модуль MAC в обычном режиме.

Результат проверки.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| USB интерфейс тест |  |  |

Сенсорные устройства.

**Требование:** Протестируйте интерфейсы I2C для всех сенсорных устройств, чтобы убедиться, что все сенсорные устройства работают в обычном режиме.

****

**Рисунок 4: Схема подключения I2C для сенсорных устройств.**

Метод тестирования: (Не загружайте встроенное ПО в FPGA для этого тестового примера)

1. Подключите главное устройство I2C к разъему J9.

2. На главном ПК запустите ОС Linux, а затем с помощью команды составьте список устройств I2C. В нем должны быть найдены все датчики, а именно: 3 датчика температуры, 1 датчик влажности, 1 датчик давления воздуха и 1 датчик вибрации.

3. Затем попробуйте прочитать/записать данные на каждый датчик, чтобы еще раз проверить, что все датчики могут работать с I2C в обычном режиме, а затем зафиксировать выходное сообщение.

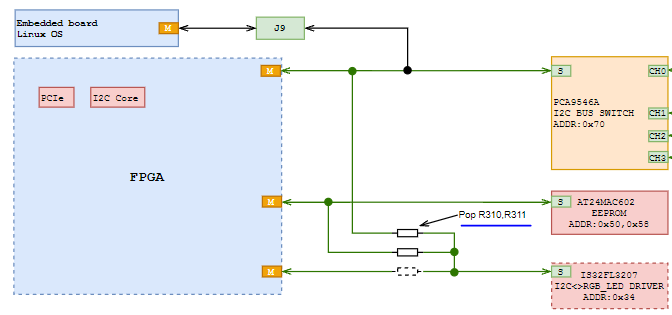
Результат тестирования.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| Температурный сенсор | I2C device list and command to test the Temperature sensors:  LM75B#1, ADDR: 0x48.  LM75B#2, ADDR: 0x49.  LM75B#3, ADDR: 0x4A.    //===================================  Попробуйте прочитать регистр: Reg 0x02 = 0x4B00, Reg 0x03 = 0x5000.  //===================================  LM75B#1, ADDR: 0x48.    //===================================  LM75B#2, ADDR: 0x49.    //===================================  LM75B#3, ADDR: 0x4A. | Pass. |
| Датчик влажности | Список устройств I2C и команда для проверки датчика влажности: SHT3X, адрес: 0x44. | Pass. |
| Датчик давления | Список устройств I2C и команда для проверки датчика давления: ICP10100, адрес: 0x63. | Pass. |
| Датчик вибрации | Список устройств I2C и команда для проверки датчика вибрации: BNO085, адрес: 0x4A. | Pass. |

* 1. EEPROM
     1. I2C интерфейс для EEPROM.

**Требование: Протестируйте интерфейсы I2C для микросхемы EEPROM, чтобы убедиться, что микросхема EEPROM работает в обычном режиме**

Метод тестирования: (Не загружайте встроенное ПО в FPGA для этого тестового примера)

1. Подключите ведущую шину I2C к устройству EEPROM I2C. Мы можем использовать разъем J9 для подключения к I2C master и pop R310, R311 для подключения EEPROM к шине I2C.

**Рисунок 5: Схема подключения I2C для устройства EEPROM.**

1. На главном компьютере запустите ОС Linux, а затем с помощью команды укажите устройство I2C, которое должно быть найдено на устройстве EEPROM.

2. Затем попробуйте считывать/записывать данные с каждого датчика, чтобы еще раз проверить, может ли устройство EEPROM работать с I2C в обычном режиме, а затем записать выходное сообщение.

3. После завершения проверки выньте резисторы R310, R311.

Результат теста.

|  |  |  |
| --- | --- | --- |
| **Тестовый узел** | **Результат** | **Да/нет** |
| EEPROM | Список устройств I2C и команда для проверки EEPROM:  АДРЕС электронной памяти AT24MAC602: 0x50, 0x58. | Pass. |
|  |  |  |

4.5.2 Интерфейс I2C для 2-го устройства EEPROM.

Требование: Протестируйте интерфейсы I2C для 2-й микросхемы EEPROM, чтобы убедиться, что 2-я микросхема EEPROM работает в обычном режиме.

Метод тестирования:

1. Включите UUT, а затем запустите ОС Linux. ≈

2. На сервере используйте команду I2C для сканирования, чтения/записи 2-й EEPROM-памяти, чтобы убедиться, что интерфейс I2C работает нормально.

Test Результат.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| I2C interface for 2nd EEPROM device. | 24LC64T EEPROM ADDR: 0x51 | Pass. |

4.6 Драйвер RGB LED и RGB LEDI2C interface for RGB LED driver.

Требование: Протестируйте интерфейсы I2C для микросхемы драйвера RGB LED, чтобы убедиться, что микросхема драйвера I2C <> RGB LED работает в обычном режиме.

Метод тестирования:

1. Включите UUT, а затем запустите ОС Linux.

2. С помощью команды введите список устройств I2C, в котором должна быть найдена микросхема драйвера RGB LED.

Результат теста.

|  |  |  |
| --- | --- | --- |
| **Тестовый узел** | **Результат** | **Да/нет** |
| I2C<> RGB LED driver. |  | Pass. |

4.5.2 Проверка светодиодов RGB.

Требование: Протестируйте каждый светодиод RGB, управляемый микросхемой драйвера RGB LED, чтобы убедиться, что мы можем управлять яркостью, включая / выключая каждый светодиод RGB с помощью команды I2C.

Метод тестирования:

1. Включите UUT, а затем запустите ОС Linux.

2. Используйте команду для отображения списка устройств I2C, в нем должно быть найдено устройство EEPROM.

3. Используйте команду I2C для управления включением/выключением яркости каждого светодиода RGB.

4. Осмотрите каждый светодиод RGB, чтобы убедиться, что все светодиоды RGB исправны.

Результат теста.

|  |  |  |
| --- | --- | --- |
| **Тестовый узел** | **Результат** | **Да/нет** |
| RGB LEDs test. | Observe each RGB LED when sending the command to control the RGB LED.  Set SL (Scaling Register: 0x4b ~0x5b) = 0xff.  //============ LED R =====================  Set PWM reg H\_L for all the RGB LED pin = 0x00.  Set PWM reg H\_L for the LED R pin = 0xff.  Set GCC (Global Current Control Register: 0x6e) = 0xff; and then Observe the LED. Test Результат = **Pass.**  Set GCC (Global Current Control Register: 0x6e) = 0x5f; and then Observe the LED. Test Результат = **Pass.**  Set GCC (Global Current Control Register: 0x6e) = 0x0f; and then Observe the LED. Test Результат = **Pass.**  //============ LED G =====================  Set PWM reg H\_L for all the RGB LED pin = 0x00.  Set PWM reg H\_L for the LED G pin = 0xff.  Set GCC (Global Current Control Register:0x6e) = 0xff; and then Observe the LED. Test Результат = **Pass.**  Set GCC (Global Current Control Register:0x6e) = 0x5f; and then Observe the LED. Test Результат = **Pass.**  Set GCC (Global Current Control Register: 0x6e) = 0x0f; and then Observe the LED. Test Результат = **Pass.**  //============ LED B =====================  Set PWM reg H\_L for all the RGB LED pin = 0x00.  Set PWM reg H\_L for the LED R pin = 0xff.  Set GCC (Global Current Control Register: 0x6e) = 0xff; and then Observe the LED. Test Результат = **Pass.**  Set GCC (Global Current Control Register: 0x6e) = 0x5f; and then Observe the LED. Test Результат = **Pass.**  Set GCC (Global Current Control Register: 0x6e) = 0x0f; and then Observe the LED. Test Результат = **Pass.** | Pass. |
|  |  |  |

4.6 ПЛИС

4.6.1 Загрузите образ ПЛИС в SPI flash с помощью JTAG

Требования: Для SPI flash ПЛИС необходимо запрограммировать встроенное ПО с помощью JTAG-программатора ПЛИС, как обычно.

Метод тестирования:

1. Подключите кабель программатора FPGA к компьютеру и разъему J14 на плате.

2. Откройте пользовательский интерфейс “Vivado” для Xilinx FPGA, чтобы загрузить файл изображения, а затем отсканируйте устройство FPGA.

3. Щелкните правой кнопкой мыши на устройстве FPGA, которое отображается на панели, чтобы добавить конфигурационную память, затем выберите SPI flash part = MT25QI128, а затем выберите файл конфигурации для FPGA и затем запрограммируйте.

**Результаты теста**

|  |  |  |
| --- | --- | --- |
| **Тестовый узел** | **Результат** | **Да/нет** |
| Загрузите прошивку в U24 FPGA с помощью JTAG. |  | Pass. |
|  |  |  |

4.2.1 Проверка интерфейса PCIe для ПЛИС

Требование: При тестировании интерфейса PCIe на ПЛИС, операционная система может обнаружить устройство на ПЛИС, и состояние соединения по протоколу PCIe должно соответствовать ширине соединения = (x1, x2 или x4), скорость соединения = 2,5 Гбит/с, как обычно.

Метод тестирования:

1. Загрузите встроенное ПО ПЛИС, поддерживающее функцию PCIe, на SPI-флэш-память ПЛИС.

2. Подключите плату к основному компьютеру

3. На главном компьютере запустите ОС Linux, а затем используйте команду “lspci -vv”, чтобы указать шину PCI и устройство.

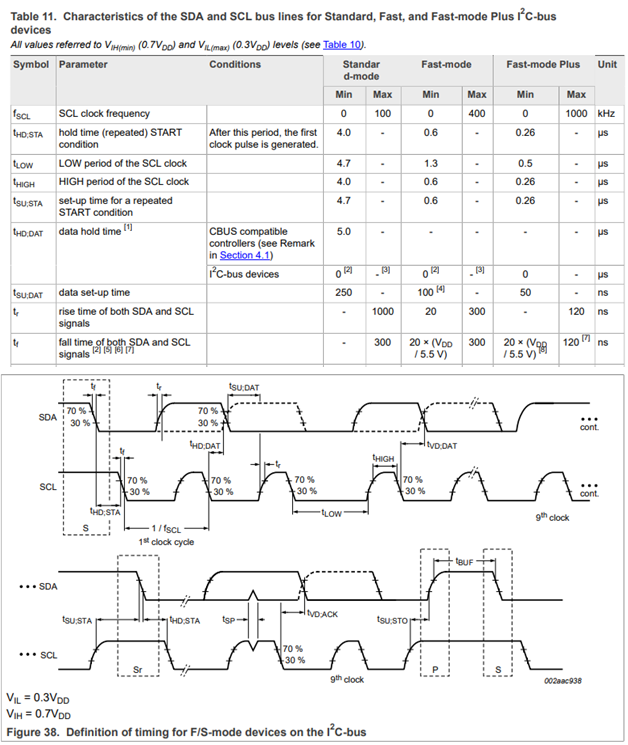
4. Проверьте выводимое сообщение, мы должны найти устройство FPGA.

Результат теста.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **ДА/НЕт** |
| Проверьте состояние подключения PCIe к ПЛИС | **Используйте версию образа ПЛИС для x4 PCIe.**  **Результат теста: скорость соединения = 2,5 Гбит/с, ширина соединения = x4.** | Pass. |
|  |  |  |

4.6 Результат измерения тактовой частоты шины I2C/данных Tr, Tf.

Требование: Протестируйте тактовую частоту шины I2C/время нарастания/спада передачи данных для всех устройств I2C.



**Рисунок 6: Спецификация шины I2C.**

Метод тестирования: (\* Для этого тестового примера мы подключили EEPROM I2C к той же шине с коммутатором PCA9546A I2C.)

1. Загрузите встроенное по ПЛИС, поддерживающее функции PCIe и FPGA I2C master, на SPI-флэш-накопитель ПЛИС.

2. Подключите плату к основной плате.

3. На главном компьютере запустите ОС Linux, а затем используйте команду I2C для доступа к устройствам I2C.

4. Измерьте параметры Tr,Tf для синхронизации сигналов I2C и данных.

\*При этом измеряются только параметры Tr, Tf.

Результат теста.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| Измерено на U35  P/N: IMBD,118  АДРЕС I2C: 0x49.  Состояние при 100 кГц:  Tr < 1000 нс.  Tf < 300 нс. |  |  |
| Измерено на U27  P/N: SHT31A  АДРЕС I2C: 0x44.  Состояние при 100 кГц:  Tr < 1000 нс.  Tf < 300 нс. |  |  |
| Измерено на U29  P/N: BNO055  АДРЕС I2C: 0x4A.  Состояние при 100 кГц:  Tr < 1000 нс.  Tf < 300 нс. |  |  |
| Измерено на U28  P/N: ICP-100  АДРЕС I2C: 0x63.  Состояние при 100 кГц:  Tr < 1000 нс.  Tf < 300 нс. |  | . |
| Измерено на U7  P/N: AT24MAC602  АДРЕС I2C: 0x30, 0x50, 0x58  Частота при 100 кГц:  Время ожидания < 1000 нс.  Tf < 300нс | - - Для этого тестового примера мы подключили I2C eeprom к той же шине с помощью переключателя I2C и зарядного устройства 1991, R19 Rpull-up для I2C eeprom с 4,7 кбайт до 100 кбайт.  - Установите переключатель PCA9546 I2C для всех каналов в положение off.  - - Таким образом, Rsum(подключаемая шина I2C) = 4,48кбайт. | Pass. |

4.6 Падение напряжения на выводе VCC буфера микросхемы для сигналов ввода-вывода FPGA<>SMA.

Требование: Измерьте падение напряжения на выводе VCC буфера микросхемы для сигналов ввода-вывода SMA при срабатывании выхода 1PPS.

Метод испытания:

1. Подключите датчик 1 для измерения выходного сигнала, поступающего с ПЛИС, к разъему SMA.

2. Установите триггер сигнала для осциллографа, нажав триггер на переднем крае сигнала.

3. Измерьте напряжение на выводе VCC буфера микросхемы в точке измерения, расположенной рядом с выводом VCC.

4. Зафиксируйте сигнал.

Результат теста.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| Измерьте напряжение на шине питания вблизи вывода VCC буфера микросхемы U15. | **Осциллограф CH-1: Сигнал частотой 10 МГц, измеренный на разъеме J3 SMA.**  **Осциллограф CH-2: Уровень напряжения, измеренный на выводе VCC буфера микросхемы (U15).**  //======================================  Увеличьте масштаб нарастающего фронта сигнала.    /======================================  Напряжение пульсации, измеренное на выводе VCC буфера микросхемы U15. V (пульсация) < 100 мВ/с-p | Pass. |
| Измерьте напряжение на шине питания вблизи вывода VCC буфера микросхемы U16. | **Осциллограф CH-1: Сигнал CH-1: 1 PPS, измеренный на разъеме J4 SMA.**  **Осциллограф CH-2: Напряжение уровня, измеренное на выводе VCC буфера микросхемы (U16).**  //======================================  Увеличьте масштаб нарастающего фронта сигнала.  //======================================  Напряжение пульсации, измеренное на выводе VCC буфера микросхемы U16. V (пульсация) < 100 мВ/с-p | Pass. |
|  |  |  |

4.6 Подключение USB FT4232 к UART/MPSSE .

4.6.1 Проверка интерфейса USB2.0 для FT4232.

Требование: Протестируйте интерфейсы USB2.0 для микросхемы FT4232 USB<>MPSSE, чтобы убедиться в нормальной работе интерфейса.

Метод тестирования:

1. Подключите Quantum PCIe к серверу.

2. Включите питание сервера.

3. Подключите кабель USB-C между Quantum PCIe и ПК-хостом.

4. На ПК-хосте запустите операционную систему Linux и затем используйте команду “lsusb”, чтобы убедиться, что ПК-хост нашел устройство FT4232 в обычном режиме.

**Результат**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| Тест интерфейса USB2.0 для FT4232. |  | Pass. |
|  |  |  |

4.2.1 FT4232: модуль MPSSE для тестирования интерфейса JTAG .

Требование: Протестируйте интерфейсы MPSSE JTAG для модуля MPSSE FT4232, чтобы убедиться, что интерфейс JTAG работает нормально.

Метод тестирования:

1. Подключите карту к серверу.

2. Включите питание сервера.

3. Подключите кабель USB-C между картой и главным компьютером.

4. На главном компьютере используйте API, поддерживающий JTAG-сканирование, для сканирования ПЛИС на плате Timecard, чтобы убедиться, что модуль MPSSE engine для интерфейса JTAG работает нормально.

Результат тестирования.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| Движок MPSSE для интерфейса JTAG  API: ftjrev с github. (Немного доработан для поддержки FT4232 и gpiol) | Parameter: ID Code = 13631093 = xc7a100t. | Pass. |
|  |  |  |

4.2.1 FT4232: Channel-модуль MPSSE для тестирования интерфейса SPI .

Требование: Протестируйте интерфейсы MPSSE SPI для модуля MPSSE FT4232, чтобы убедиться, что интерфейс JTAG работает нормально.

Метод тестирования:

1. Подключите Карту к серверу.

2. Включите питание сервера.

3. Подключите кабель USB-C между картой и главным компьютером.

4. На главном компьютере используйте API, поддерживающий SPI-интерфейс между чипом FTDI <> SPI flash on Quantum PCIe, чтобы убедиться, что движок MPSSE для интерфейса SPI работает нормально.

Результат тестирования.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| Движок MPSSE для интерфейса SPI  API: инструмент flashrom.  \* Поскольку инструмент flashrom еще не поддерживает текущую версию SPI flash, которую мы используем. Затем, для этого теста, мы изменили SPI flash на p/n: MX25L12835F | Read    Write    Erase | Pass. |
|  |  |  |

4.4.2 FT4232: Модуль MPSSE канала B для тестирования интерфейса I2C.

Требование: Протестируйте интерфейсы MPSSE I2C для модуля FT4232 MPSSE, чтобы убедиться, что интерфейс I2C работает нормально.

Метод тестирования:

1. Включите сервер HPE.

2. Подключите кабель USB-C между Quantum PCIe и ПК.

3. На сервере HPE запустите команду i2cdump и запишите результат.

4. На главном компьютере используйте API, поддерживающий интерфейс I2C между чипом FTDI <> Устройствами I2C на плате Timecard, чтобы убедиться, что модуль MPSSE для интерфейса I2C работает нормально.

5. Сравните результат.

Результат тестирования.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| Движок MPSSE для тестирования интерфейса I2C.  Используйте исходный файл "ftdi-i2c/i2cget.c" на C с github, а затем измените его для поддержки микросхемы FT4232 и добавьте сигнал FT4232\_MUX3\_SEL на CH-A для управления мультиплексорным переключателем для устройств FTDI<> I2C. |  | Pass. |
|  |  |  |

4.4.2 FT4232: Проверка интерфейса UART канала C.

Требование: Проверьте интерфейсы UART для движка, чтобы убедиться, что UART FT4232 работает нормально.

Метод тестирования: (Не загружайте встроенное ПО в ПЛИС для этого тестового примера)

1. На временной карте подключите сетевое имя “UART\_FT4232\_TX\_FPGA\_RX” к контакту J10 pin2, “UART\_FT4232\_TX\_FPGA\_TX” к контакту J10 pin6. Это соединение FT4232 CH-C UART с модулем UART для MAC.

2. Подключите источник питания 12 В к разъему P3.

3. Подключите кабель USB-C между Quantum PCIe и ПК-хостом.

4. На главном КОМПЬЮТЕРЕ откройте API терминала UART. Установите последовательный терминал: скорость передачи данных = 57600 бод, бит данных = 8, бит четности = нет, стоп-бит = 1, управление потоком = нет.

5. Используя API терминала UART, отправьте символ “c” в модуль MAC.

6. Запишите результат, который отправляется из модуля MAC.

Результат тестирования.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| FT4232: Тест интерфейса UART Channel-C. | CH-C = COM29 |  |
|  |  |  |

4.4.2 FT4232: Проверка интерфейса UART канала D.

Требование: Проверьте интерфейсы UART для движка, чтобы убедиться, что UART FT4232 работает нормально.

Метод тестирования: (Не загружайте встроенное ПО в FPGA для этого тестового примера)

1. На Quantum PCIe на разъеме J10 установите перемычку для соединения выводов 1 и 2 и выводов 5 и 6. Это соединение FT4232 CH-D UART с модулем UART для MAC.

2. Подключите источник питания 12 В к разъему P3.

3. Подключите кабель USB-C между Quantum PCIe и ПК-хостом.

4. На главном КОМПЬЮТЕРЕ откройте API терминала UART. Установите последовательный терминал: скорость передачи данных = 57600 бод, бит данных = 8, бит четности = нет, стоп-бит = 1, управление потоком = нет.

5. Используя API терминала UART, отправьте символ “c” в модуль MAC.

6. Запишите результат, отправленный из модуля MAC.

Результат тестирования.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| FT4232: Проверка интерфейса Channel-D UART. | CH-D = COM30 |  |

4.5 Функция программируемого времени задержки U34

Требование: Протестируйте функцию программируемого времени задержки на U34, которая регулирует время задержки с помощью кнопки R C на выводе CT.

Метод тестирования:

1. Включите питание сервера.

2. С помощью осциллографа измерьте значение FPGA\_IN\_RST\_DLY\_N, чтобы проверить время задержки, запрограммированное выводом CT на U34 (на схеме: CT= 100 НФ, TD = 572 мс.).

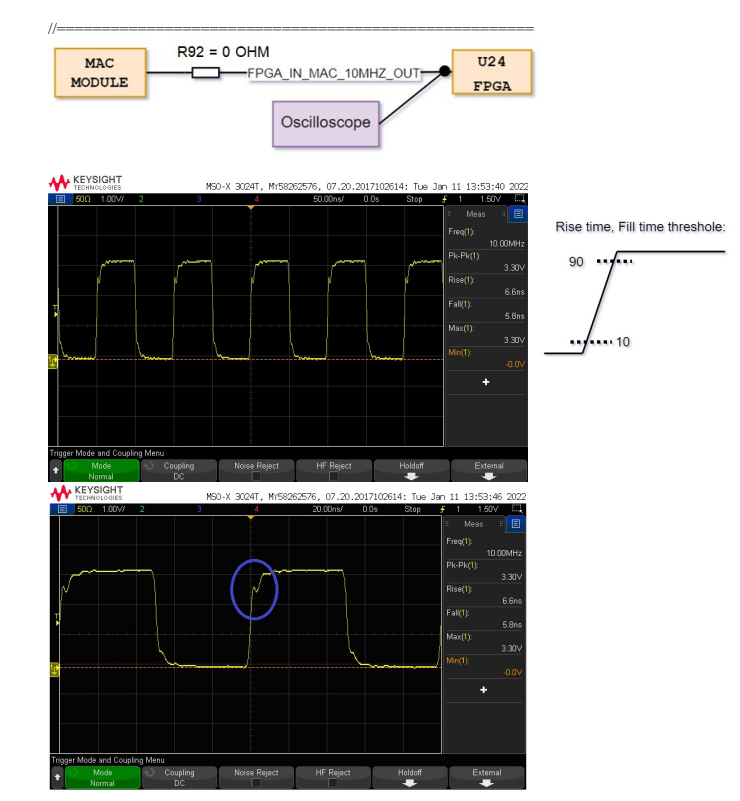
Тест

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| U34 программируемая функция времени задержки. | TD ≈ 565ms |  |
|  |  |  |

RC для уменьшения отражения сигнала.

4.7.1 R-оконечник для 10 МГц от MAC-модуля к FPGA.

Требование: На этапе EVR мы обнаружили, что имя сигнала FPGA\_IN\_MAC\_10 MHZ\_OUT должно регулировать R-сопретевление (R92), чтобы улучшить/уменьшить отражение сигнала. Для фазы версии PCI v1.1 выполните измерение в сигнальной сети с именем FPGA\_IN\_MAC\_10 MHZ\_OUT и затем проверьте результат.



Метод тестирования:

1. Включите сервер.

2. Установите осциллограф на пороговые значения Tr, Tf: Tr=30%, Tf=70%

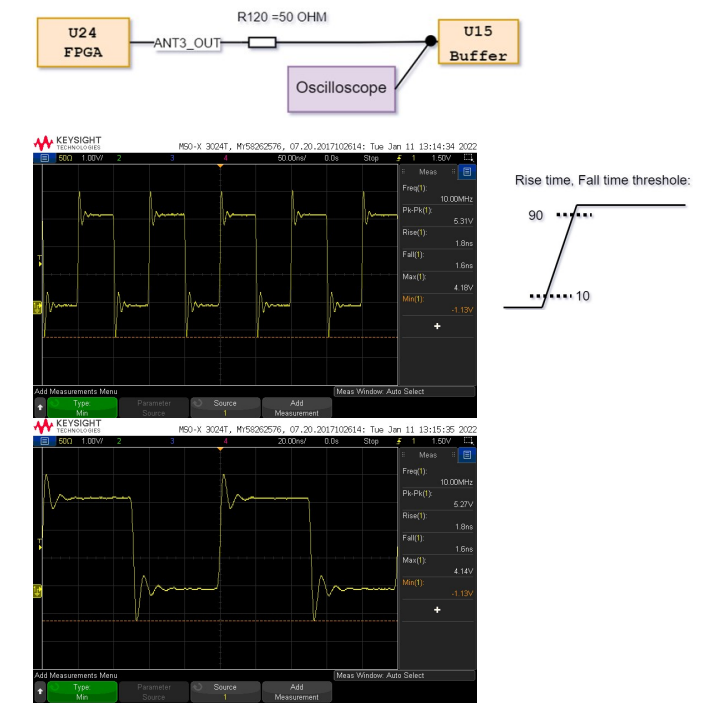
3. С помощью осциллографа измерьте сетевое значение сигнала FPGA\_IN\_MAC\_10MHZ\_OUT, точку измерения, расположенную рядом с выводом ПЛИС.

Результат теста.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| U34 программируемая функция времени задержки. | ================EVT===================      ================DVT=================== |  |

4.6.1 RC для передачи 1PPS из ПЛИС в буфер микросхемы

Требование: На этапе мы нашли название сигнала, И [4:1]\_OUT должен настроить R-серию и добавить параллельное завершение RC-серии, чтобы улучшить/уменьшить отражение сигнала. Для фазы v1.1 измерьте сетевое значение сигнала ANT[4:1]\_OUT и затем проверьте результат.



**Метод тестирования:**

1. Включите сервер.

2. Перейдите по пути: /sys/class/timecard/ocp 0 и затем используйте команду , чтобы установить все out , с помощью команды:

echo OUT: 10Mhz > sma 1

echo OUT: 10Mhz > sma 2

echo OUT: 10 Mhz > sma 3

echo OUT: 10 Mhz > sma4

3. Установите осциллограф на пороговое значение Tr, Tf: Tr=30%, Tf=70%

4. Используйте осциллограф для измерения входного сигнала ANT[4:1]\_OUT, точки измерения, близкой к контакту 2 буфера микросхемы.

**Результат теста.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| ANT1\_OUT | ================EVT===================    ================DVT=================== | Pass. |
| ANT2\_OUT | ================EVT===================    ================DVT=================== | Pass. |
| ANT3\_OUT | ================EVT===================    ================DVT=================== | Pass. |
| ANT4\_OUT | ================DVT=================== | Pass. |
|  |  |  |

5 Функциональный тест

В этом разделе нам потребуется прошивка FPGA и команда DIAG для тестирования всех функций, таких как

Интерфейс ядра FPGA I2C к EEPROM I2C и сенсорным устройствам.

● GPIO.

● Драйвер светодиодов и RGB-светодиодов.

● Сигналы ввода-вывода FPGA<>SMA на внешние устройства.

● Интерфейс ядра FPGA UART к модулю GPS и MAC.

● Интерфейсы USB, FPGA<>MAC-модуль.

● и т.д.,

Тест 5.2 PCIe-интерфейсов FPGA

Требование: Используйте команду DIAG для тестирования интерфейса PCIe на ПЛИС. Операционная система может обнаружить устройство FPGA, и состояние соединения PCIe должно соответствовать ширине соединения = (x1, x2 или x4), скорость соединения = 2,5 Гбит/с, как обычно.

Метод тестирования:

1. Загрузите встроенное ПО ПЛИС, поддерживающее функцию PCIe, на SPI-флэш-накопитель ПЛИС.

2. Подключите плату Quantum PCIe к основной плате.

3. На главном компьютере запустите ОС Linux, а затем с помощью команды DIAG протестируйте шину PCI и устройство.

4. Проверьте выходное сообщение, а затем запишите его.

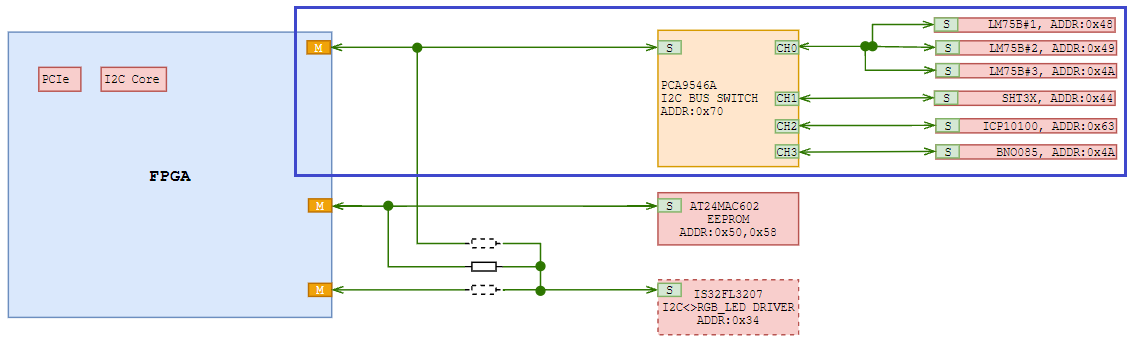
Результат проверки.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| FPGA PCIe interfaces |  | PASS. |

4.7 Проверка интерфейса FPGA I2C master <> I2C устройств

4.7.1 Проверка интерфейса FPGA I2C master <> I2C сенсорных устройств

Требование: Протестируйте интерфейсы I2C для всех сенсорных устройств, используя FPGA в качестве ведущего устройства I2C, чтобы убедиться, что функция FPGA I2C master<> Интерфейсы сенсорных устройств I2C работают нормально.



**Рисунок 7: Интерфейс FPGA I2C master <> I2C sensor devices.**

Метод тестирования:

1. Подключите плату Quantum PCIe к основной плате.

2. включите питание, затем загрузитесь в ОС Linux.

3. На главном ПК запустите ОС Linux и установите пакет драйверов, поддерживающий FPGA I2C master<>I2C-устройство.

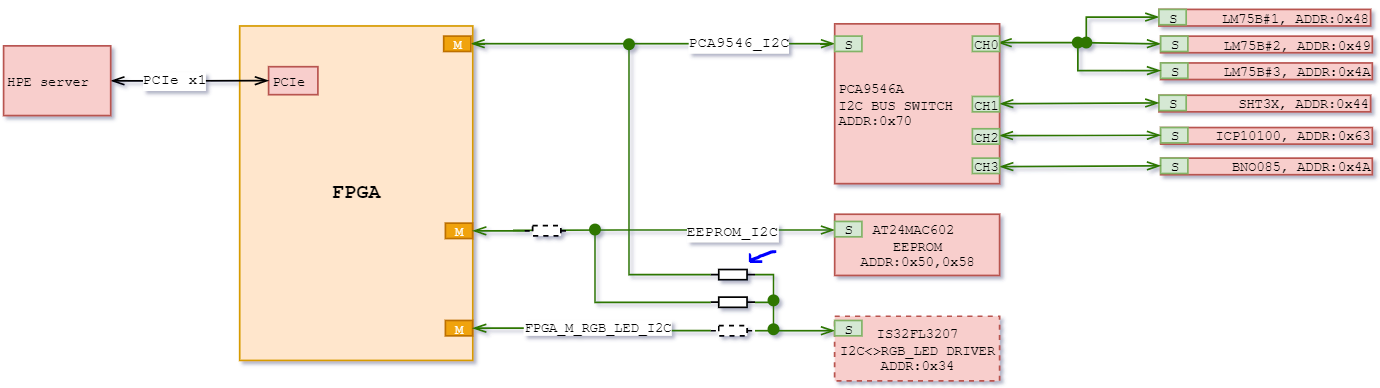
4. Используйте команду DIAG для тестирования каждого датчика, чтобы убедиться, что мы можем получить доступ ко всем датчикам через FPGA I2C master<>I2C в обычном режиме, а затем записать выходное сообщение.

Результат тестирования.

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| Temperature sensors |  | PASS. |
| Humidity sensor |  | PASS. |
| Pressure sensor |  | PASS. |
| Vibration sensor |  | PASS. |
|  |  |  |

* + 1. FPGA I2C master <> I2C EEPROM devices interface test

**Requirement:** Test I2C interfaces for all the sensor devices by using FPGA as I2C master to confirm that the function FPGA I2C master<> I2C EEPROM interface is working as normally. This test, we have tested by using bus I2C-0. Then, pop R310, R311 to connect the EEPROM I2C bus to the I2C bus I2C-0.



**Figure 8: FPGA I2C master <> I2C EEPROM devices interface.**

**Test method:**

1. Plug the Quantum PCIe to the main board.
2. Power on the then boot to Linux OS
3. At the PC host, run the Linux OS and install the package and driver that support the FPGA I2C master<>I2C device.
4. Use the DIAG command to test the EEPROM to confirm that we can access the EEPROM by FPGA I2C master<>I2C EEPROM device as normally and then capture the output message.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| EEPROM |  | PASS. |
|  |  |  |

* 1. FPGA LED driver for RGB\_LED test

**Requirement:** Test FPGA<> LED indicator to confirm that the FPGA can control the USR\_LED, RGB\_LED as normally.

**Test method:**

1. Plug the Quantum PCIe to the main board.
2. Power on the then boot to Linux OS
3. At the PC host, run the Linux OS and install the package, and driver that support the FPGA LED driver function.
4. Use the DIAG command to test the RGB LED and observe the difference to confirm that FPGA<> RGB\_LED: DS14, DS15, DS16, DS17, DS18 function are working as normally.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| RGB\_LED |  | PASS. |
|  |  |  |

* 1. FPGA UART core interface to GPS and MAC module
     1. FPGA UART<> GPS module UART interfaces test

**Requirement:** Test FPGA UART<> GPS UART interfaces to confirm that the FPGA UART core can communicate with.

**Test method:**

1. Plug the Quantum PCIe to the main board.
2. Power on the then boot to Linux OS
3. At the PC host, run the Linux OS and install the package, driver that support the FPGA UART core function.
4. Use the DIAG command to test the FPGA UART core <> GPS UART interfaces to confirm that it is working as normally and then capture the output message.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| FPGA UART<> GPS UART interfaces |  | PASS. |
|  |  |  |

* + 1. FPGA UART<> MAC module UART interfaces test

**Requirement:** Test FPGA UART<> MAC module UART interfaces to confirm that the FPGA UART core can communicate with.

**Test method:**

1. Plug the Quantum PCIe to the main board.
2. Power on the then boot to Linux OS.
3. At the PC host, run the Linux OS and install the package, driver that support the FPGA UART core function.
4. Use the DIAG command to test the FPGA UART core <> MAC module UART interfaces to confirm that it is working as normally and then capture the output message.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| FPGA UART<> MAC module UART interfaces |  | PASS. |

* 1. Single Client PTP Test by phc2sys command

**Test method:**

1. Connect SFP+ cable to SFP+ port of Calnex Sentinel to the QSFP port on the Mellanox NIC (installed on the server).
2. Set-up the Calnex Sentinel equipment for PTP test.
3. At the HPE server, switch the console port to MAC module by using the command “tio –b 57600 /dev/ttyS6” and then set the parameter to:

\{set,Disciplining,1}

\{set,PpsWidth,80000000}

\{set,TauPps0,10000}

\{set,PpsOffset,-30}

\{set,DisciplineThresholdPps0,20}

\{store}

1. Warm –up the HPC server ~18 hrs. After that, at the HPE server, run ptp4l: ptp4l -i ens1f0 -f unicast-master.cfg –m and phc2sys: phc2sys -s /dev/ptp6 -c ens1f0 -O 0 -m.
2. Run test for 12 hrs.
3. Compare PTP times and verify that they are < ±2.5usec.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| 12 hrs single client PTP test by phc2sys command.  Timecard = Disciplining.  Sentinel = Disciplining (Internal Reference Disciplining Mode: Always, Disciplining source: GNSS).  \* GNSS antenna of Quantum PCIe and Sentinel equipment = inside the building close to the window. |  |  |
|  |  |  |

* 1. Single Client PTP Test by phc2sys command

**Test method:**

1. Connect SFP+ cable to SFP+ port of Calnex Sentinel to the QSFP port on the Mellanox NIC (installed on the server).
2. Set-up the Calnex Sentinel equipment for PTP test.
3. At the HPE server, switch the console port to MAC module by using the command “tio –b 57600 /dev/ttyS6” and then set the parameter to:

\{set,Disciplining,1}

\{set,PpsWidth,80000000}

\{set,TauPps0,10000}

\{set,PpsOffset,-30}

\{set,DisciplineThresholdPps0,20}

\{store}

1. Warm –up the HPC server ~18 hrs. After that, at the HPE server, run run ptp4l: ptp4l -i ens1f0 -f unicast-master.cfg –m and phc2sys: phc2sys -s /dev/ptp6 -c ens1f0 -O 0 -m. Run test for 12 hrs.
2. Compare PTP times and verify that they are < ±2.5usec.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| 12 hrs single client PTP test by phc2sys command.  Timecard = Disciplining.  Sentinel = Disciplining (Internal Reference Disciplining Mode: Always, Disciplining source: GNSS).  \* GNSS antenna of Quantum PCIe and Sentinel equipment = outside the building close to the window. | //====================================  //======Результат for: \{set,PpsOffset,-30}=======  PTP 2WayTE mean value ≈ - 2.501us, exceeding the criteria ≈ 0.001us.      //======================================  //======Результат for: \{set,PpsOffset,-2500}=======  PTP 2WayTE mean value ≈ 9.069ns. | PASS with condition by adjust the PpsOffset parameter on the MAC module. |

* 1. Single Client PTP Test by ts2phc command

**Test method:**

1. Connect SFP+ cable to SFP+ port of Calnex Sentinel to the QSFP port on the Mellanox NIC (installed on the HPE server).
2. Connect 1PPS output from Timecard to input of NIC card.
3. Set-up the Calnex Sentinel equipment for PTP test.
4. At the HPE server, switch the console port to MAC module by using the command “tio –b 57600 /dev/ttyS6” and then set the parameter to:

\{set,Disciplining,1}

\{set,PpsWidth,80000000}

\{set,TauPps0,1000}

\{set,PpsOffset, -30}

\{set,DisciplineThresholdPps0,20}

\{store}

1. Warm –up the HPC server ~18 hrs. After that, at the HPE server, run phc2sys to sync the TOD clock from Quantum PCIe to system clock on HPE server by command “phc2sys -s /dev/ptp6 -c CLOCK\_REALTIME -O 0 -m”.
2. Set the 1PPS input pin for the NIC card by command: ./testptp -d /dev/ptp4 -L 0,1.
3. Stop the phc2sys command and then run ts2phc command to align the timestamp by command ts2phc -s generic -c ens1f0 -m -l 7.
4. Run ptp4l command “ptp4l -i ens1f0 -f unicast-master.cfg –m” to test PTP protocol with the Sentinel Equipment.
5. Run test for 12 hrs.
6. Compare PTP times and verify that they are < ± 500 ns.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| 12 hrs single client PTP test by ts2phc command.  Timecard = Disciplining.  Sentinel = Disciplining (Internal Reference Disciplining Mode: Always, Disciplining source: GNSS).  \* GNSS antenna of Quantum PCIe and Sentinel equipment = inside the building close to the window. |  | PASS. |
|  |  |  |

* 1. Single Client PTP Test by ts2phc command

**Test method:**

1. Connect SFP+ cable to SFP+ port of Calnex Sentinel to the QSFP port on the Mellanox NIC (installed on the HPE server).
2. Connect 1PPS output from Timecard to input of NIC card.
3. Set-up the Calnex Sentinel equipment for PTP test.
4. At the HPE server, switch the console port to MAC module by using the command “tio –b 57600 /dev/ttyS6” and then set the parameter to:

\{set,Disciplining,1}

\{set,PpsWidth,80000000}

\{set,TauPps0,1000}

\{set,PpsOffset, -30}

\{set,DisciplineThresholdPps0,20}

\{store}

1. Warm –up the HPC server ~18 hrs. After that, at the HPE server, run phc2sys to sync the TOD clock from Quantum PCIe to system clock on HPE server by command “phc2sys -s /dev/ptp6 -c CLOCK\_REALTIME -O 0 -m”.
2. Set the 1PPS input pin for the NIC card by command: ./testptp -d /dev/ptp4 -L 0,1.
3. Stop the phc2sys command and then run ts2phc command to align the timestamp by command ts2phc -s generic -c ens1f0 -m -l 7.
4. Run ptp4l command “ptp4l -i ens1f0 -f unicast-master.cfg –m” to test PTP protocol with the Sentinel Equipment.
5. Run test for 12 hrs.
6. Compare PTP times and verify that they are < ± 500 ns.

|  |  |  |
| --- | --- | --- |
| **Test Результат.Тест** | **Результат** | **Да/нет** |
| 12 hrs single client PTP test by ts2phc command.  Timecard = Disciplining.  Sentinel = Disciplining (Internal Reference Disciplining Mode: Always, Disciplining source: GNSS).  \* GNSS antenna of Quantum PCIe and Sentinel equipment = outside the building close to the window. |  | PASS. |
|  |  |  |

* 1. 1PPS 24 Hour Holdover Test

**Requirement**: Verify that 1PPS from the timing card does not drift more than ±500usec over a 24 hr period when GNSS signal is lost.

**Test method:**

1. Connect the SMA cable from SMA3 (SMA3 the default = OUT PHC) of the Timecard module to input CH-C of the Calnex sentinel equipment.
2. Set-up the Calnex Sentinel equipment for 1PPS test.
3. At the HPE server, switch the console port to MAC module by using the command “tio –b 57600 /dev/ttyS6” and then set the parameter to:

\{set,Disciplining,1}

\{set,PpsWidth,80000000}

\{set,TauPps0,10000}

\{set,PpsOffset,-30}

\{set,DisciplineThresholdPps0,20}

\{store}

1. Warm-up the HPE server ~18 hrs. After that, set the parameter = \{set,Disciplining,0} for holdover mode.
2. Run test for 24 hrs.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| 1PPS 24 hrs holdover test.  Timecard = Stop disciplining.  Sentinel = Disciplining (Internal Reference Disciplining Mode: Always, Disciplining source: GNSS).  \* GNSS antenna of Quantum PCIe and Sentinel equipment = inside the building close to the window. |  | PASS. |
|  |  |  |

* 1. 1PPS 24 Hour Holdover Test

**Requirement**: Verify that 1PPS from the timing card does not drift more than ±500usec over a 24 hr period when GNSS signal is lost.

**Test method:**

1. Connect the SMA cable from SMA4 (For the DVT, the default for SMA4 = OUT PHC) of the Timecard module to input CH-B of the Calnex sentinel equipment.
2. Set-up the Calnex Sentinel equipment for 1PPS test.
3. At the HPE server, switch the console port to MAC module by using the command “tio –b 57600 /dev/ttyS6” and then set the parameter to:

\{set,Disciplining,1}

\{set,PpsWidth,80000000}

\{set,DisciplineThresholdPps0,20}

\{set,TauPps0, 10} ; Set Tau = 10 run 100sec.

\{set,TauPps0, 100} ; Set Tau = 100 run 20min.

\{set,TauPps0, 5000} ; Set Tau = 5,000 discipline time >= 6hr 56min).

1. Discipline time >= 6hr 56min. this test disciplining time ~12 hrs. After that, set the parameter = \{set,Disciplining,0} for holdover mode.
2. Run test for 24 hrs.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| 1PPS 24 hrs holdover test.  Timecard = Stop disciplining.  Sentinel = Disciplining (Internal Reference Disciplining Mode: Stop discipline during measurement, Disciplining source: GNSS).  \* GNSS antenna of Quantum PCIe and Sentinel equipment = outside the building near to the window.  \*\* This test, we have monitored the digital tuning value and average value for the digital tuning. When discipline time > 5Tau, send command to stop the disciplining at the digital tuning value move close to midpoint [average line]. |  | PASS. |
|  |  |  |

* 1. Single PCIE delay test.

**Requirement**: Verify that the offset when synchronizing the timing card with the NIC over the PCIe bus is <+-5usec)

**Test method:**

1. Connect Calnex Sentinel to the PPS output of the timing card.
2. Connect the other port of Calnex sentinel to PPS output of Mellanox NIC.
3. Run the utility phc2sys (phc2sys -s /dev/ptp6 -c ens1f0 -O 0 -m).
4. Observe the difference.

**Test Результат.**

|  |  |  |
| --- | --- | --- |
| **Тест** | **Результат** | **Да/нет** |
| ens1f0 PHC offset < ±5usec | PHC offset <± 5us | Pass. |
|  |  |  |