

JEDEC STANDARD

SPD5118 Hub and Serial Presence Detect Device Standard

JESD300-5B Version 1.5

July 2022

JEDEC SOLID STATE TECHNOLOGY ASSOCIATION



NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or refer to www.jedec.org under Standards and Documents for alternative contact information.

Published by
©JEDEC Solid State Technology Association 2022
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2108

JEDEC retains the copyright on this material. By downloading this file the individual agrees not to charge for or resell the resulting material.

PRICE: Contact JEDEC

Printed in the U.S.A.
All rights reserved

PLEASE!

DON'T VIOLATE
THE
LAW!

This document is copyrighted by JEDEC and may not be reproduced
without permission.

For information, contact:

JEDEC Solid State Technology Association
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

or refer to www.jedec.org under Standards-Documents/Copyright Information.

This page intentionally left blank

SPD5118 HUB and SERIAL PRESENCE DETECT DEVICE STANDARD**Contents**

| | Pages |
|---|--------------|
| 1 Scope | 1 |
| 2 Definition of SPD5 5118 Hub Device for Memory Module Applications | 1 |
| 2.1 Device Standard | 1 |
| 2.1.1 Description | 1 |
| 2.1.2 Common Features Summary: | 2 |
| 2.2 Device Power Up | 2 |
| 2.3 Device Reset and Initialization | 3 |
| 2.4 I2C and I3C Basic Operation | 3 |
| 2.5 Device Interface - IO Voltage Configuration | 4 |
| 2.5.1 Open Drain Interface with Internal On Die Pullup Resistor | 4 |
| 2.5.2 Open Drain Interface with External Pullup Resistor | 4 |
| 2.5.3 Push Pull Interface with Internal On Die Pullup Resistor | 5 |
| 2.6 Device Interface - Protocol | 5 |
| 2.6.1 Serial Address of the SPD5 Hub Device | 6 |
| 2.6.2 Serial Address of the Local Devices | 6 |
| 2.6.3 Switch from I2C Mode to I3C Basic Mode | 6 |
| 2.6.4 Switch from I3C Basic Mode to I2C Mode | 6 |
| 2.6.5 SPD5 Hub Device Selection | 7 |
| 2.6.6 Local Device Selection Through the SPD5 Hub Device (Before SETHID CCC) | 7 |
| 2.6.7 Local Device Selection Through the SPD5 Hub Device (After SETHID CCC) | 9 |
| 2.6.8 I2C Target Protocol - Host to SPD5 Hub Device | 9 |
| 2.6.9 I2C Target Protocol - Host to Local Device Through SPD5 Hub Device | 13 |
| 2.6.10 I3C Basic Target Protocol - Host to SPD5 Hub Device | 18 |
| 2.6.11 I3C Basic Target Protocol - Host to Local Device (Through SPD5 Hub Device) | 27 |
| 2.6.12 In Band Interrupt (IBI) | 41 |
| 2.6.13 Packet Error Check (PEC) Function | 55 |
| 2.6.14 Parity Error Check Function | 55 |
| 2.6.15 Packet Error Check and Parity Error Handling | 55 |
| 2.6.16 CCC Packet Error Handling | 60 |
| 2.6.17 Error Reporting | 60 |
| 2.6.18 I3C Basic Common Command Codes (CCC) | 60 |
| 2.6.19 IO Operation | 75 |
| 2.6.20 Bus Clear | 75 |
| 2.6.21 Bus Reset | 76 |
| 2.6.22 Command Truth Table | 76 |
| 2.7 SPD5 Hub Device Pin Definition | 78 |
| 2.8 SPD5 Hub Device - Write and Read Access | 78 |
| 2.8.1 Write and Read Access - NVM Memory | 79 |
| 2.8.2 Write and Read Access - Register Memory | 79 |
| 2.9 Write Protection of Non-Volatile Memory | 79 |
| 2.9.1 Normal Run Time Operation (HSA Pin is Tied to GND via a Resistor Value) | 79 |

Contents (cont'd)

| | |
|--|------------|
| 2.9.2 Offline Tester Operation (HSA Pin is Tied Directly to GND, No Resistor Value). | 79 |
| 2.9.3 Suggested Steps to Program SPD5 Hub Devices | 80 |
| 2.10 AC Timing Definition..... | 80 |
| 2.10.1 I2C or I3C Basic Bus Timing | 80 |
| 2.10.2 Hub Propagation Delay..... | 84 |
| 2.11 Parametric Characteristics | 86 |
| 2.11.1 Absolute Maximum Ratings | 86 |
| 2.11.2 Operating Condition, Measurement Condition, and DC and AC Characteristics | 86 |
| 2.11.3 Temperature Sensor Performance | 91 |
| 2.12 Device Package and Pinout..... | 91 |
| 2.12.1 Package Pinout..... | 91 |
| 2.12.2 Recommended Landing Pattern..... | 92 |
| 2.12.3 Mechanical Outline Drawing..... | 92 |
| 3 Volatile Registers Space | 94 |
| 3.1 Access Mechanism | 94 |
| 3.1.1 Register Attribute Definition | 94 |
| 3.2 Registers..... | 95 |
| 3.2.1 Register Map..... | 95 |
| 3.2.2 Thermal Sensor Registers Read Out Mechanism | 96 |
| 3.2.3 Register Description | 96 |
| 4 Annex A — (Informative) Differences Between Revisions | 113 |
| 4.1 Differences Between Version 1.1 and 1.0 | 113 |
| 4.2 Differences Between Version 1.2 and 1.1 | 113 |
| 4.3 Differences Between Version 1.3 and 1.2 | 114 |
| 4.4 Differences Between Version 1.4 and 1.3 | 114 |
| 4.5 Differences Between Version 1.5 and 1.4 | 114 |

Tables

| | |
|---|----|
| Table 1 — Device Part Numbers and Feature Summary | 2 |
| Table 2 — 7-bit Address of SPD5 Hub Device | 6 |
| Table 3 — 7-bit Address of the Local Devices (e.g., PMIC Device) | 6 |
| Table 4 — 7-bit Address of Each Hub Devices on I2C/I3C Bus | 7 |
| Table 5 — 7-bit Address of Local Devices on I2C/I3C Bus | 7 |
| Table 6 — Write Command Data Packet; Table 13, “MR11” [3] = ‘0’ | 10 |
| Table 7 — Write Command Data Packet; Table 13, “MR11” [3] = ‘1’ | 11 |
| Table 8 — Read Command Data Packet; Table 13, “MR11” [3] = ‘0’ | 11 |
| Table 9 — Read Command Data Packet; Table 13, “MR11” [3] = ‘1’ | 12 |
| Table 10 — Read Command Data Packet with Default Address Pointer Mode | 12 |
| Table 11 — Write Command Data Packet (e.g., TS) | 13 |
| Table 12 — Write Command Data Packet (e.g., PMIC) | 13 |
| Table 13 — Write Command Data Packet (e.g., RCD; PEC Disabled) | 14 |
| Table 14 — Write Command Data Packet (e.g., RCD; PEC Enabled) | 14 |

Contents (cont'd)

| | |
|---|----|
| Table 15 —Read Command Data Packet (e.g., TS) | 15 |
| Table 16 —Read Command Data Packet (e.g., PMIC) | 15 |
| Table 17 —Read Command Data Packet (e.g., RCD; PEC Disabled; Legacy Format) | 16 |
| Table 18 —Read Command Data Packet (e.g., RCD; PEC Enabled; Legacy Format) | 17 |
| Table 19 —Read Command Data Packet (e.g., RCD; PEC Disabled; Optimized Format) | 18 |
| Table 20 —Write Command Data Packet; PEC Disabled | 19 |
| Table 21 —Write Command Data Packet; PEC Enabled | 19 |
| Table 22 —Write Command Data Packet with IBI Header; PEC Disabled | 20 |
| Table 23 —Write Command Data Packet with IBI Header; PEC Enabled | 21 |
| Table 24 —Read Command Data Packet; PEC Disabled | 22 |
| Table 25 —Read Command Data Packet; PEC Enabled | 23 |
| Table 26 —Read Command Data Packet with IBI Header; PEC Disabled | 24 |
| Table 27 —Read Command Data Packet with IBI Header; PEC Enabled | 25 |
| Table 28 —Read Command Data Packet with Read Address Pointer Mode; PEC Disabled .. | 26 |
| Table 29 —Read Command Data Packet with Read Address Pointer Mode; PEC Enabled ... | 26 |
| Table 30 —Read CMD Data Packet with Read Address Pointer Mode and IBI Header; PEC Disabled | 26 |
| Table 31 —Read CMD Data Packet with Read Address Pointer Mode and IBI Header; PEC Enabled | 27 |
| Table 32 —Write Command Data Packet (e.g., TS); PEC Disabled | 27 |
| Table 33 —Write Command Data Packet (e.g., TS); PEC Enabled | 28 |
| Table 34 —Write Command Data Packet (e.g., PMIC); PEC Disabled | 28 |
| Table 35 —Write Command Data Packet (e.g., PMIC); PEC Enabled | 29 |
| Table 36 —Write Command Data Packet (e.g., RCD); PEC Disabled | 29 |
| Table 37 —Write Command Data Packet (e.g., RCD); PEC Enabled | 30 |
| Table 38 —Read Command Data Packet (e.g., TS); PEC Disabled | 31 |
| Table 39 —Read Command Data Packet (e.g., TS); PEC Enabled | 32 |
| Table 40 —Read Command Data Packet (e.g., PMIC); PEC Disabled | 33 |
| Table 41 —Read Command Data Packet (e.g., PMIC); PEC Enabled | 34 |
| Table 42 —Read Command Data Packet (e.g., RCD); PEC Disabled | 35 |
| Table 43 —Read Command Data Packet (e.g., RCD); PEC Enabled | 36 |
| Table 44 —Hub IBI Payload Packet; PEC is Disabled | 42 |
| Table 45 —Hub IBI Payload Packet; PEC is Enabled | 43 |
| Table 46 —Target Device IBI Payload Packet; PEC is Disabled | 44 |
| Table 47 —Target Device IBI Payload Packet; PEC is Enabled | 45 |
| Table 48 —Interrupt Arbitration - Among SPD5 Hub Devices | 48 |
| Table 49 —Interrupt Arbitration - Among Local Target Devices | 49 |
| Table 50 —Interrupt Arbitration - Among Local Target and SPD5 Hub Devices | 51 |
| Table 51 —Interrupt Arbitration - Best Host and Local Target and SPD5 Hub Devices | 53 |
| Table 52 —Write Command Data Packet; PEC Disabled | 56 |
| Table 53 —Read Command Data Packet; PEC Disabled | 57 |
| Table 54 —Write Command Data Packet; PEC Enabled | 58 |
| Table 55 —Read Command Data Packet; PEC Enabled | 59 |
| Table 56 —SPD5 Hub CCC Support Requirement | 61 |
| Table 57 —ENEC CCC - Broadcast | 62 |

Contents (cont'd)

| | |
|---|----|
| Table 58 — ENEC CCC - Broadcast with PEC | 62 |
| Table 59 — ENEC CCC - Direct | 62 |
| Table 60 — ENEC CCC - Direct with PEC | 63 |
| Table 61 — ENEC CCC - Byte Encoding | 63 |
| Table 62 — DISEC CCC - Broadcast | 63 |
| Table 63 — DISEC CCC - Broadcast with PEC | 64 |
| Table 64 — DISEC CCC - Direct | 64 |
| Table 65 — DISEC CCC - Direct with PEC | 64 |
| Table 66 — DISEC CCC Byte Encoding | 65 |
| Table 67 — RSTDAA CCC - Broadcast | 65 |
| Table 68 — RSTDAA CCC - Broadcast with PEC | 65 |
| Table 69 — SETAASA CCC - Broadcast | 66 |
| Table 70 — GETSTATUS CCC - Direct | 66 |
| Table 71 — GETSTATUS CCC - Direct with PEC1 | 67 |
| Table 72 — GETSTATUS CCC Byte Encoding | 67 |
| Table 73 — DEVCAP CCC - Direct | 68 |
| Table 74 — DEVCAP CCC - Direct with PEC | 68 |
| Table 75 — DEVCAP CCC Byte Encoding | 68 |
| Table 76 — SETHID CCC - Broadcast | 69 |
| Table 77 — DEVCTRL CCC - Broadcast | 70 |
| Table 78 — DEVCTRL CCC - Broadcast with PEC1 | 70 |
| Table 79 — DEVCTRL CCC Command Definition | 71 |
| Table 80 — DEVCTRL CCC Data Payload Definition | 72 |
| Table 81 — DEVCTRL CCC Example - Multicast Command to '1001' and '0110' Devices | 72 |
| Table 82 — DEVCTRL CCC Example - Broadcast Command to All Devices | 73 |
| Table 83 — DEVCTRL CCC Example - Unicast Command to PMIC on DIMM5 | 73 |
| Table 84 — DEVCTRL CCC Example - Multicast Command to '0010' and '1001' Devices | 74 |
| Table 85 — DEVCTRL CCC Example - Multicast Command to '1001' Devices | 74 |
| Table 86 — SPD5 Hub Device Dynamic IO Operation Mode Switching | 75 |
| Table 87 — For I3C Basic Mode Only with PEC Enabled - Command Truth Table | 77 |
| Table 88 — Pin Description - SPD5118 | 78 |
| Table 89 — HSA Pin Resistor Value and ID - SPD5 Family | 78 |
| Table 90 — HSCL Slope Reversal Parameters | 84 |
| Table 91 — Absolute Maximum Ratings | 86 |
| Table 92 — Operating Conditions | 86 |
| Table 93 — Write Endurance and Data Retention Characteristics | 87 |
| Table 94 — AC Measurement Conditions1 | 87 |
| Table 95 — Input Parameters | 87 |
| Table 96 — Output Ron Specification | 88 |
| Table 97 — DC Characteristics | 88 |
| Table 98 — AC Characteristics | 89 |
| Table 99 — Temperature Sensor Performance | 91 |
| Table 100 — Register Base Attributes | 94 |
| Table 101 — Register Attribute Modifier | 94 |
| Table 102 — Register Map | 95 |

Contents (cont'd)

| | |
|--|-----|
| Table 103 —Thermal Register - Low Byte and High Byte | 96 |
| Table 104 —Thermal Register Examples | 96 |
| Table 105 —MR0 | 96 |
| Table 106 —MR1 | 97 |
| Table 107 —MR2 | 97 |
| Table 108 —MR3 | 97 |
| Table 109 —MR4 | 97 |
| Table 110 —MR5 | 98 |
| Table 111 —MR6 | 98 |
| Table 112 —MR11 | 99 |
| Table 113 —MR12 | 100 |
| Table 114 —MR13 | 101 |
| Table 115 —MR14 | 102 |
| Table 116 —MR15 | 102 |
| Table 117 —MR18 | 103 |
| Table 118 —MR19 | 104 |
| Table 119 —MR20 | 104 |
| Table 120 —MR26 | 105 |
| Table 121 —MR27 | 106 |
| Table 122 —MR28 | 107 |
| Table 123 —MR29 | 107 |
| Table 124 —MR30 | 107 |
| Table 125 —MR31 | 108 |
| Table 126 —MR32 | 108 |
| Table 127 —MR33 | 108 |
| Table 128 —MR34 | 109 |
| Table 129 —MR35 | 109 |
| Table 130 —MR48 | 110 |
| Table 131 —MR49 | 111 |
| Table 132 —MR50 | 111 |
| Table 133 —MR51 | 111 |
| Table 134 —MR52 | 112 |

Figures

| | |
|---|----|
| Figure 1 —Device Power Up Sequence | 3 |
| Figure 2 —Open Drain Interface; Internal Pullup Resistor | 4 |
| Figure 3 —Open Drain Interface; External On Board Pullup Resistor | 5 |
| Figure 4 —Push Pull Interface; On Die Pullup | 5 |
| Figure 5 —Example: Host Accessing RCD on DIMM 0 | 8 |
| Figure 6 —Example: Host Accessing Temperature Sensor 0 on DIMM 3 | 9 |
| Figure 7 —Target Open Drain to Host Push Pull Hand Off Operation | 37 |

Contents (cont'd)

| | |
|---|----|
| Figure 8 —Controller Open Drain (ACK) to Target Push Pull Hand Off Operation | 38 |
| Figure 9 — Controller Push Pull to Target Open Drain Hand Off Operation | 39 |
| Figure 10 —T=1; Host Ends Read with Repeated START and STOP Waveform | 40 |
| Figure 11 —T=0; Target Ends Read; Host Generates STOP | 41 |
| Figure 12 —SPD5 Hub Interrupt; Host Ack Followed by SPD5 Hub Device IBI Payload | 43 |
| Figure 13 —SPD5 Hub Interrupt; Host NACK Followed by STOP | 43 |
| Figure 14 —Local Device Interrupt, Host Ack followed by Target Device IBI Payload | 46 |
| Figure 15 —Local Device Interrupt; Host NACK followed by STOP | 47 |
| Figure 16 —I2C or I3C Basic Bus Reset - SPD5 Hub Device | 76 |
| Figure 17 — I2C or I3C Basic Bus AC Input Timing Parameter Definition | 80 |
| Figure 18 —I3C Basic Bus AC Data Output Timing Parameter Definition | 80 |
| Figure 19 —I2C Bus AC Data Output Timing Parameter Definition | 81 |
| Figure 20 —Output Slew Rate and Output Timing Reference Load | 81 |
| Figure 21 —Output Slew Rate Measurement Points | 81 |
| Figure 22 —Rise and Fall Timing Parameter Definition | 82 |
| Figure 23 —Slope Reversal on HSCL at VIH | 83 |
| Figure 24 —Slope Reversal on HSCL Between VIL and VIH | 83 |
| Figure 25 —Slope Reversal on HSCL at VIL | 84 |
| Figure 26 —Propagation Delay Through the SPD5 Hub Device | 85 |
| Figure 27 —AC Measurement Waveform | 87 |
| Figure 28 —Pinout for SPD5 Hub Devices | 91 |
| Figure 29 —Common Landing Array | 92 |
| Figure 30 —Mechanical Outline Drawing | 93 |

SPD5118 HUB and SERIAL PRESENCE DETECT DEVICE STANDARD

From JEDEC Board Ballot JCB-22-32, formulated under the cognizance of JC-40.1, Digital Logic Families and Applications, item 600-02A.

1 Scope

This standard defines the specifications of interface parameters, signaling protocols, and features for DDR5 Serial Presence Detect EEPROM with Hub function (SPD5 Hub) and integrated Temperature Sensor (TS) as used for memory module applications. The Hub feature allows isolation of a local bus from a Controller host bus. The designation SPD5118 or generic term SPD5 Hub refers to the devices specified by this standard.

The purpose is to provide a standard for the SPD5 Hub devices for uniformity, multiplicity of sources, elimination of confusion, ease of device specification, and ease of use.

Unless otherwise noted in the document, any illegal operation is not allowed and device operations is not guaranteed.

The designation SPD5118 refers to the part designation of a series of commercial logic parts common in the industry. This number is normally preceded by a series of manufacturer specific characters to make up a complete part designation.

2 Definition of SPD5 5118 Hub Device for Memory Module Applications

2.1 Device Standard

2.1.1 Description

The SPD5 Hub device family contains 1024 bytes of non-volatile memory arranged as 16 blocks of 64 bytes per block. Each block may optionally be write protected via software command. Write protection for each block may be overridden in an offline programmer environment while overrides are prevented in normal use. The SPD5 Hub device operate from 1.8 V nominal power supply input. The SPD5 Hub device is intended to operate up to 12.5 MHz on a 1.0V I²C Basic bus or up to 1 MHz on a 1.0 V to 3.3 V I²C bus. The SPD5 Hub devices are intended to interface to I²C/I³C Basic buses which have multiple devices on a shared bus, and must be uniquely addressed with fixed addressing on the same bus.

All SPD5 Hub devices respond to specific pre-defined I²C/I³C Basic device select codes on a host interface bus. The SPD5118 devices also incorporate a second local I²C/I³C Basic bus and pass through of commands from the host bus onto the local bus for addressing of I²C/I³C Basic devices on the local bus (Hub function).

The SPD5118 device incorporates thermal sensing capability which is controlled and read over I²C/I³C Basic bus.

2.1.2 Common Features Summary:

Table 1 — Device Part Numbers and Feature Summary

| Part Number | Hub Function | Internal Thermal Sensor | Non Volatile Memory (Bytes) | Temperature Range |
|-------------|--------------|-------------------------|-----------------------------|-------------------|
| SPD5118 | Yes | Yes | 1024 | -40 °C to +125 °C |

- Two wire I²C or I3C Basic bus serial interface
- Single device load on the Host bus
- Up to 12.5 MHz transfer rate
- 1.8 V, 1.0 V power supply input
- 1.0 V Push Pull I/O levels
- 1.0 V, 1.1 V, 1.2 V, 1.8 V, 2.5 V, and 3.3 V Open Drain IO levels
- 1024 Bytes of non-volatile memory (16 blocks; 64 bytes per block)
- Hub Function
- Integrated Temperature Sensor
- Packet Error Check (PEC) Function
- Parity Error Check Function
- Bus Reset Function
- Up to 8 unique addressing
- In Band Interrupt (IBI)
- Write protection for each block of NVM
- 9-pin thermally enhanced DFN package

2.2 Device Power Up

The SPD5 Hub device has one VDDSPD supply input.

In order to prevent inadvertent operations during power up, a Power On Reset (POR) circuit is included. On cold power on, VDDSPD input supply must rise monotonically between VPON and VDDSPD_{min} and VDDIO input supply must rise monotonically to VDDIO_{min} without ringback to ensure proper startup.

The SPD5 Hub device uses VDDIO supply for its IO levels and it must reach VDDIO_{min} to ensure proper operation of I²C or I3C Basic bus interface.

Once the VDDSPD and VDDIO supply is valid and stable, the SPD5 Hub device shall:

1. Once VDDSPD supply is valid and stable, within t_{Sense_HSA} time, sense its HSA pin to determine if SPD5 Hub device is in application environment or in an offline tester program mode. Depending on what it sense on HSA pin, the SPD5 Hub device configures its HID code automatically based on what it detects on HSA pin at power up.
2. Enable I²C interface within t_{INIT} time and be ready to receive the command from the host. The SPD5 Hub device is ready for operation after t_{INIT} time.

2.2 Device Power Up (cont'd)

The host interface HSDA and HSCL signal pins are pulled up to a pullup voltage through a pullup resistor on the platform or on the host controller. The pullup voltage can be available before or after VDDSPD or VDDIO is valid and stable. If HSDA and HSCL pullup voltage is available before VDDSPD or VDDIO is available, the HSDA and HSCL signal is high and remains high with no leakage path or damage to the SPD Hub device.

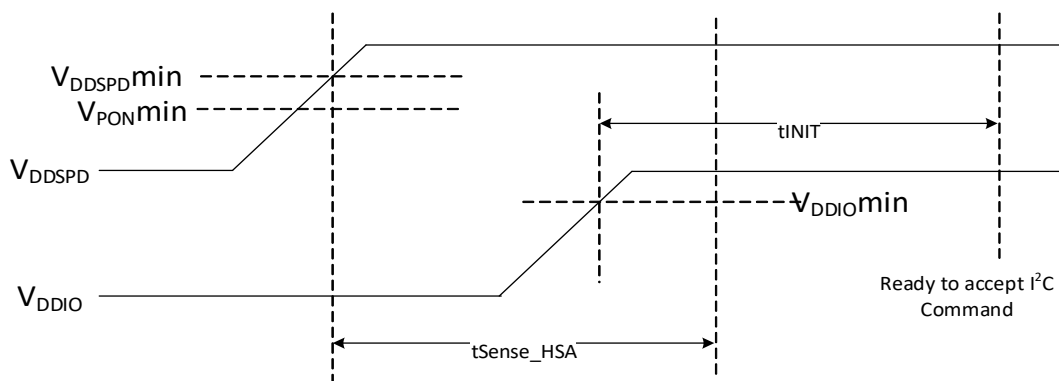


Figure 1 — Device Power Up Sequence

2.3 Device Reset and Initialization

At power down (phase during which VDDSPD input supply decreases continuously), as soon as VDDSPD input supply drops below the V_{DDSPD_min} , the SPD5 Hub device does not guarantee the operation.

On warm power cycling, the VDDSPD and VDDIO input supply must remain below V_{POFF} for t_{POFF} and must meet cold power on reset timing when restoring the power.

2.4 I²C and I3C Basic Operation

At power on, by default, the SPD5 Hub device comes up in I²C mode of operation. Following applies in I²C mode:

1. The maximum operation speed is limited to 1 MHz.
2. In-band interrupts are not supported.
3. Bus reset is supported.
4. Parity check is not supported except for supported CCCs.
5. Packet Error check is not supported.

The SPD5 Hub device shall operate in the I²C mode until put into I3C Basic mode via command.

The host may put the SPD5 Hub device in I3C Basic mode by issuing SETAASA CCC.

Following applies in I3C mode.

1. The maximum operation speed is up to 12.5 MHz
2. In-band interrupts are supported
3. Bus reset is supported.
4. Parity check is always enabled by default.
5. Packet error check is supported and by default is disabled.

2.5 Device Interface - IO Voltage Configuration

The SPD5 Hub device supports configurable Open Drain and Push Pull IO levels to accommodate broad range of DDR5 platform.

2.5.1 Open Drain Interface with Internal On Die Pullup Resistor

Figure 2 shows the SPD5 Hub device configuration options for Open Drain interface for both Host side and local side of the device. In this configuration, the SPD5 Hub device supports Open Drain IO levels on both Host and Local side. However, the IO voltage levels on Host side and Local side are independent and can be different.

On Host side, the SPD5 device can support IO levels from 1.0 V to 3.3 V depending on the supply rail Host may have pulled up the resistor to. The host side pullup resistor can be on motherboard or on die inside the host logic device.

On local side, the SPD5 device can support IO levels from 1.0 V to 1.2 V and register Table 115, “MR14” [5] must be configured to ‘0’.

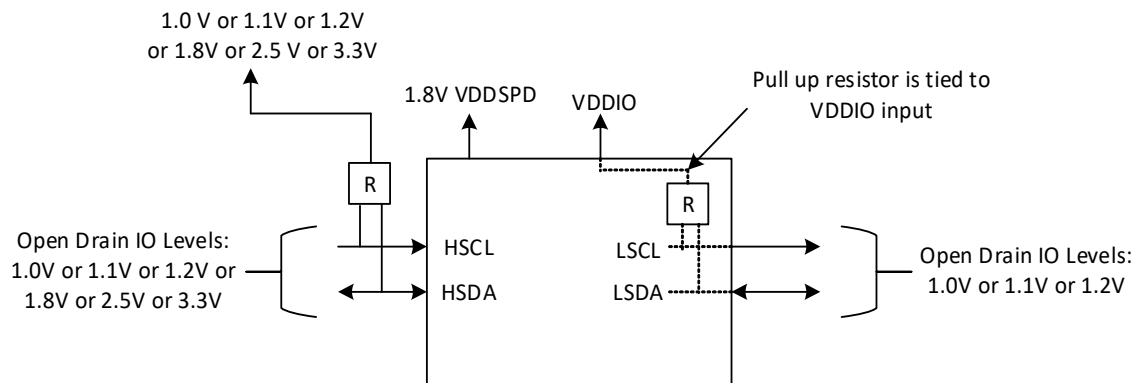


Figure 2 — Open Drain Interface; Internal Pullup Resistor

2.5.2 Open Drain Interface with External Pullup Resistor

Figure 3 shows the SPD5 Hub device configuration options for Open Drain interface for both Host side and local side of the device. In this configuration, the SPD5 device supports Open Drain IO levels on both Host and Local side. However, the IO voltage levels on Host side and Local side are independent and can be different.

On Host side, the SPD5 Hub device can support IO levels from 1.0 V to 3.3 V depending on the supply rail Host may have pulled up the resistor to. The host side pullup resistor can be on motherboard or on die inside the host logic device.

On local side, the SPD5 device can support IO levels from 1.0 V to 3.3 V and Table 115, “MR14” [5] must be configured to ‘1’.

2.5.2 Open Drain Interface with External Pullup Resistor (cont'd)

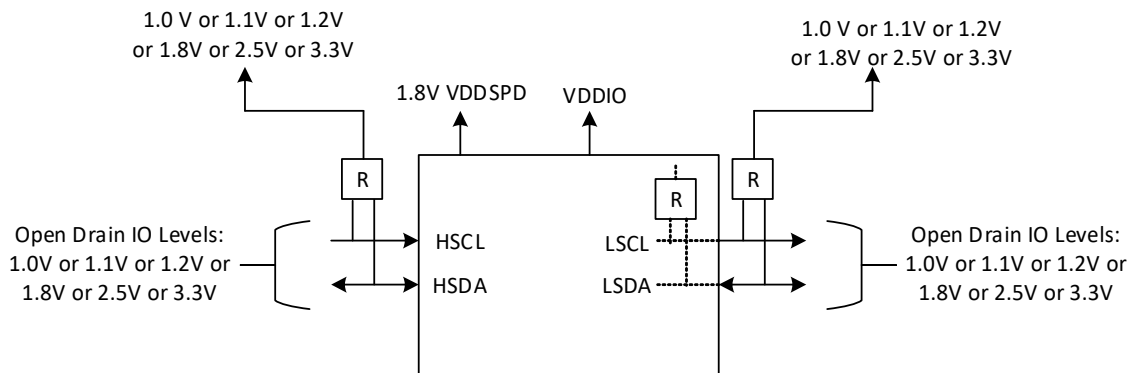


Figure 3 — Open Drain Interface; External On Board Pullup Resistor

2.5.3 Push Pull Interface with Internal On Die Pullup Resistor

The configuration option shown in Figure 4 is only supported when device is in I3C Basic mode.

Figure 4 shows the SPD5 Hub device configuration options for Push Pull interface for both Host side and local side of the device. In this configuration, the SPD5 Hub device supports Push Pull IO levels on both Host and Local side. However, the pullup IO voltage levels on Host side and Local side are independent and can be different.

On Host side, the SPD5 Hub device can support IO levels from 1.0 V to 1.2 V. The host side pullup resistor can be on motherboard or on die inside the host logic device.

On local side, the SPD5 Hub device can support IO levels from 1.0 V to 1.2V and Table 115, “MR14” [5] can be configured to ‘0’ or ‘1’.

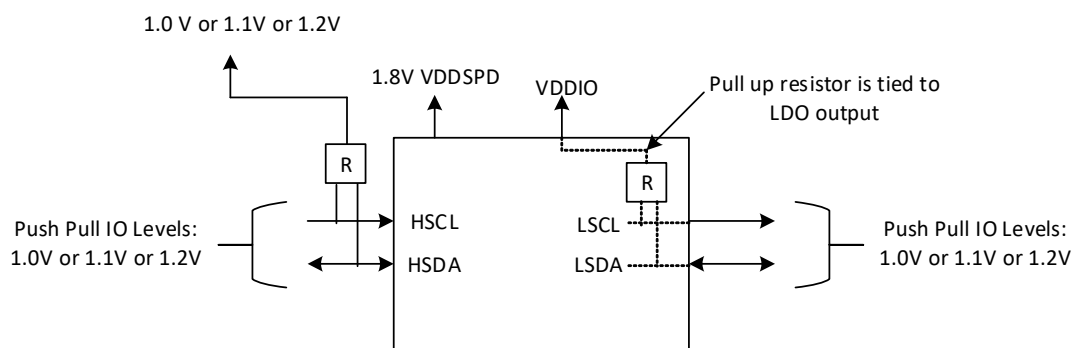


Figure 4 — Push Pull Interface; On Die Pullup

2.6 Device Interface - Protocol

The 7-bit serial address of the SPD5 Hub device and all local devices behind the SPD5 Hub applies to both I²C and I3C Basic mode of operation identically.

2.6.1 Serial Address of the SPD5 Hub Device

The SPD5 Hub device type ID is 4-bit binary value of 1010b.

The SPD5 Hub device samples the status of the HSA pin on power up. The sampled status of the HSA pin determine the unique host ID (HID) of the device. The host identifier value (HID) is merged with the SPD5 Hub device type 1010 xxxb to establish the 7-bit address (SPD5 Hub DevID[6:0]) for the device on the I²C or I3C Basic bus as shown in Table 2. For example, if the value sensed on HSA pin identifier is 2 (010 binary), then the unique 7-bit address for this device is 1010 010b.

Table 2 — 7-bit Address of SPD5 Hub Device

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------------------------|-------|-------|-------|---------------|--------|--------|------------|
| 1 | 0 | 1 | 0 | HID[2] | HID[1] | HID[0] | R/W |
| SPD5 Hub Device Type ID (LID) | | | | Host ID (HID) | | | Read/Write |

2.6.2 Serial Address of the Local Devices

Each local device behind the SPD5 Hub device has a unique 4 bit local device ID (LID) code. For example the PMIC local device type ID is 4-bit binary value of 1001b.

The 3 HID bits of the local device type has a default value of ‘111’ and is merged with its unique 4 bit local ID code to establish the 7-bit address (DevID[6:0]) for the local device on the I²C or I3C Basic bus as shown in Table 3. For example, the PMIC local device behind the SPD5 Hub has 7-bit address of 1001 111b.

Table 3 — 7-bit Address of the Local Devices (e.g., PMIC Device)

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|----------------------------|-------|-------|-------|---------------|-------|-------|------------|
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | R/W |
| Local Device Type ID (LID) | | | | Host ID (HID) | | | Read/Write |

2.6.3 Switch from I²C Mode to I3C Basic Mode

By default, when SPD5 Hub first powers on, it operates in I²C mode. The SPD5 Hub device shall operate in I²C mode until put into I3C Basic mode via command.

In I²C mode, the host is allowed to issued only 3 CCCs (DEVCTRL, SETHID, SETAASA). All other CCCs are not supported and the SPD5 Hub device may simply ignore it. The Host must issue DEVCTRL and SETHID CCC first (if required) followed by SETAASA CCC.

The Host puts the SPD5 Hub device in I3C Basic mode by issuing SETAASA CCC.

When SETAASA CCC is registered by the SPD5 Hub device, it updates the Table 117, “MR18” [5] to ‘1’.

When SETHID CCC is registered by the SPD5 Hub device, it stops the 3-bit HID translation for the local Target device as explained in clause 2.6.6.

2.6.4 Switch from I3C Basic Mode to I²C Mode

The Host can put the SPD5 Hub device back in I²C mode from I3C mode at any time by issuing RSTDAA CCC.

When RSTDAA CCC is registered by the SPD5 Hub device, it updates the Table 117, “MR18” [5] to ‘0’.

2.6.5 SPD5 Hub Device Selection

The host can access any SPD5 Hub device as shown in Table 4 in both I²C mode and I3C Basic mode. The last 3 bits represent the HID bits.

Table 4 — 7-bit Address of Each Hub Devices on I²C/I3C Bus

| Host Access to: | 7-bit Address |
|-----------------|---------------|
| DIMM 0 SPD Hub | 1010 000 |
| DIMM 1 SPD Hub | 1010 001 |
| DIMM 2 SPD Hub | 1010 010 |
| DIMM 3 SPD Hub | 1010 011 |
| DIMM 4 SPD Hub | 1010 100 |
| DIMM 5 SPD Hub | 1010 101 |
| DIMM 6 SPD Hub | 1010 110 |
| DIMM 7 SPD Hub | 1010 111 |

2.6.6 Local Device Selection Through the SPD5 Hub Device (Before SETHID CCC)

By default, the SPD5 Hub device and all local Target devices powers up in I²C mode of operation, Prior to Host issuing SETHID CCC, the host can access the any local device behind the SPD5 Hub device. To access the local device, the host sends 7-bit address that is made up of 4 bits of local device code (LID) followed by the 3 bits of HID code. The LID code represents which local device host is intended to target. The HID code represents which DIMM the local device resides on. Table 5 shows an example of four different local device address codes behind SPD5 Hub on each DIMM.

Table 5 — 7-bit Address of Local Devices on I²C/I3C Bus

| Host Access to: | 7-bit Address LID = 1011 (RCD) | 7-bit Address LID = 1001 (PMIC) | 7-bit Address LID = 0010 (TS0) | 7-bit Address LID = 0110 (TS1) |
|---------------------|--------------------------------------|---------------------------------------|--------------------------------------|--------------------------------------|
| DIMM 0 Local Device | 1011 000 | 1001 000 | 0010 000 | 0110 000 |
| DIMM 1 Local Device | 1011 001 | 1001 001 | 0010 001 | 0110 001 |
| DIMM 2 Local Device | 1011 010 | 1001 010 | 0010 010 | 0110 010 |
| DIMM 3 Local Device | 1011 011 | 1001 011 | 0010 011 | 0110 011 |
| DIMM 4 Local Device | 1011 100 | 1001 100 | 0010 100 | 0110 100 |
| DIMM 5 Local Device | 1011 101 | 1001 101 | 0010 101 | 0110 101 |
| DIMM 6 Local Device | 1011 110 | 1001 110 | 0010 110 | 0110 110 |
| DIMM 7 Local Device | 1011 111 | 1001 111 | 0010 111 | 0110 111 |

The SPD5 Hub Device monitors the LID code coming from the host. When it detects the host access is for the Target device, it compares the last 3 bits of the HID information (shown in Red) coming from the host against its own unique HID code that it has stored at power on. It compares each 3 bits one at a time. If there is a match, the SPD5 Hub device substitutes that bit with ‘1’ and forward it to the local device interface. If there is a mismatch, the SPD5 Hub device substitutes that bit with ‘0’ and forwards it to the local device interface. As a result, only the targeted local device will see its last three HID bits as ‘111’ and all non-targeted local devices will see its last three HID bits as anything other than ‘111’ which is not a valid code as shown in Table 3.

2.6.6 Local Device Selection Through the SPD5 Hub Device (Before SETHID CCC) (cont'd)

There are two exceptions when SPD5 Hub device does not substitute its own HID code when it forwards it to the local Target interface:

1. Host issues Start followed by 7'h7E with W=0 (or Host issues Start followed by 0xFC).
2. After SPD5 Hub executes SETHID CCC command that Host issues. See clause 2.6.7.

Figure 5 gives an example of Host accessing local RCD Device on DIMM0. The figure shows Host sends 7-bit address '1011 000'. Each SPD5 Hub devices receives this address. Each SPD5 Hub device forward first four bits of binary address '1011' (LID) on the local device interface. Each SPD5 Hub compares last 3 bits of binary address '000' from the host against its own unique HID code and substitutes the bit on the local device interface.

| | Hub SPD | | RCD | | PMIC0 | | PMIC1 | | PMIC2 | | TS0 | | TS1 | |
|------------|----------|----|----------|----|----------|----|----------|----|----------|----|----------|----|----------|----|
| DIMM0 | 101 0000 | 50 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 101 1000 | 58 | | | | | | | | | | | | |
| Hub Sends | | | 101 1111 | 5F | 101 1111 | 5F | 101 1111 | 5F | 101 1111 | 5F | 101 1111 | 5F | 101 1111 | 5F |
| DIMM1 | 101 0001 | 51 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 101 1000 | 58 | | | | | | | | | | | | |
| Hub Sends | | | 101 1110 | 5E | 101 1110 | 5E | 101 1110 | 5E | 101 1110 | 5E | 101 1110 | 5E | 101 1110 | 5E |
| DIMM2 | 101 0010 | 52 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 101 1000 | 58 | | | | | | | | | | | | |
| Hub Sends | | | 101 1101 | 5D | 101 1101 | 5D | 101 1101 | 5D | 101 1101 | 5D | 101 1101 | 5D | 101 1101 | 5D |
| DIMM3 | 101 0011 | 53 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 101 1000 | 58 | | | | | | | | | | | | |
| Hub Sends | | | 101 1100 | 5C | 101 1100 | 5C | 101 1100 | 5C | 101 1100 | 5C | 101 1100 | 5C | 101 1100 | 5C |
| DIMM4 | 101 0100 | 54 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 101 1000 | 58 | | | | | | | | | | | | |
| Hub Sends | | | 101 1011 | 5B | 101 1011 | 5B | 101 1011 | 5B | 101 1011 | 5B | 101 1011 | 5B | 101 1011 | 5B |
| DIMM5 | 101 0101 | 55 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 101 1000 | 58 | | | | | | | | | | | | |
| Hub Sends | | | 101 1010 | 5A | 101 1010 | 5A | 101 1010 | 5A | 101 1010 | 5A | 101 1010 | 5A | 101 1010 | 5A |
| DIMM6 | 101 0110 | 56 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 101 1000 | 58 | | | | | | | | | | | | |
| Hub Sends | | | 101 1001 | 59 | 101 1001 | 59 | 101 1001 | 59 | 101 1001 | 59 | 101 1001 | 59 | 101 1001 | 59 |
| DIMM7 | 101 0111 | 57 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 101 1000 | 58 | | | | | | | | | | | | |
| Hub Sends | | | 101 1000 | 58 | 101 1000 | 58 | 101 1000 | 58 | 101 1000 | 58 | 101 1000 | 58 | 101 1000 | 58 |

Figure 5 — Example: Host Accessing RCD on DIMM 0

Figure 6 gives another example of Host accessing local Temperature Sensor 0 device on DIMM3. The figure shows Host sends 7-bit address '0010 011'. Each SPD5 hub devices receives this address. Each SDP5 Hub device forward first four bits of binary address '0010' (LID) on the local device interface. Each SPD5 Hub compares last 3 bits of binary address '011' from the host against its own unique HID code and substitutes the bit on the local device interface.

2.6.6 Local Device Selection Through the SPD5 Hub Device (Before SETHID CCC) (cont'd)

| | Hub SPD | | RCD | | PMIC0 | | PMIC1 | | PMIC2 | | TS0 | | TS1 | |
|------------|----------|----|----------|----|----------|----|----------|----|----------|----|----------|----|----------|----|
| DIMM0 | 101 0000 | 50 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 001 0011 | 13 | | | | | | | | | | | | |
| Hub Sends | | | 001 0100 | 14 | 001 0100 | 14 | 001 0100 | 14 | 001 0100 | 14 | 001 0100 | 14 | 001 0100 | 14 |
| DIMM1 | 101 0001 | 51 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 001 0011 | 13 | | | | | | | | | | | | |
| Hub Sends | | | 001 0101 | 15 | 001 0101 | 15 | 001 0101 | 15 | 001 0101 | 15 | 001 0101 | 15 | 001 0101 | 15 |
| DIMM2 | 101 0010 | 52 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 001 0011 | 13 | | | | | | | | | | | | |
| Hub Sends | | | 001 0110 | 16 | 001 0110 | 16 | 001 0110 | 16 | 001 0110 | 16 | 001 0110 | 16 | 001 0110 | 16 |
| DIMM3 | 101 0011 | 53 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 001 0011 | 13 | | | | | | | | | | | | |
| Hub Sends | | | 001 0111 | 17 | 001 0111 | 17 | 001 0111 | 17 | 001 0111 | 17 | 001 0111 | 17 | 001 0111 | 17 |
| DIMM4 | 101 0100 | 54 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 001 0011 | 13 | | | | | | | | | | | | |
| Hub Sends | | | 001 0000 | 10 | 001 0000 | 10 | 001 0000 | 10 | 001 0000 | 10 | 001 0000 | 10 | 001 0000 | 10 |
| DIMM5 | 101 0101 | 55 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 001 0011 | 13 | | | | | | | | | | | | |
| Hub Sends | | | 001 0001 | 11 | 001 0001 | 11 | 001 0001 | 11 | 001 0001 | 11 | 001 0001 | 11 | 001 0001 | 11 |
| DIMM6 | 101 0110 | 56 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 001 0011 | 13 | | | | | | | | | | | | |
| Hub Sends | | | 001 0010 | 12 | 001 0010 | 12 | 001 0010 | 12 | 001 0010 | 12 | 001 0010 | 12 | 001 0010 | 12 |
| DIMM7 | 101 0111 | 57 | 101 1111 | 5F | 100 1111 | 4F | 100 0111 | 47 | 110 0111 | 67 | 001 0111 | 17 | 011 0111 | 37 |
| Host Sends | 001 0011 | 13 | | | | | | | | | | | | |
| Hub Sends | | | 001 0011 | 13 | 001 0011 | 13 | 001 0011 | 13 | 001 0011 | 13 | 001 0011 | 13 | 001 0011 | 13 |

Figure 6 — Example: Host Accessing Temperature Sensor 0 on DIMM 3

2.6.7 Local Device Selection Through the SPD5 Hub Device (After SETHID CCC)

When SETHID CCC is registered by the SPD5 Hub device, it stops the 3-bit HID translation for the local Target device as explained in clause 2.6.6. After Host sends SETHID CCC, the Host still access the all Target devices behind the SPD5 Hub as shown in Table 5. There is no change in how Host access the SPD5 Hub device and all local Target devices behind the SPD5 Hub device before or after SETHID CCC.

2.6.8 I²C Target Protocol - Host to SPD5 Hub Device

The SPD5 Hub devices operate on a standard I²C serial interface. Transactions where the SPD5 Hub device is the targeted Target device begin with the Host issuing a START condition followed by a 7-bit SPD5 Hub device address then a read or write bit, RW. All data are transmitted with the most significant bit MSB first. During the address followed by R/W bit transmission, the SPD5 Hub device typically replies with an ACK unless there are conditions when it may passively assert a NACK.

2.6.8 I²C Target Protocol - Host to SPD5 Hub Device (cont'd)

The SPD5 Hub device does not allow any read or write operations to its non-volatile memory when it is busy with internal write operation to non-volatile memory. The host should either check Table 130, “MR48” [3] register status or ensure that Write time (t_w) parameter is satisfied prior to performing another write or read to operations non-volatile memory. If host violates this and performs either write or read operation to non-volatile memory then SPD5 Hub device ACKs the address byte to allow possible volatile register access. However, the SPD5 Hub device NACKs the subsequent write or read operation if MemReg = ‘1’ and sets the Table 134, “MR52” [7] = ‘1’.

The SPD5 Hub device does allow any read or write operation to its volatile memory (i.e., MemReg = ‘0’) when it is busy with internal write operation to non-volatile memory.

To allow compatibility with legacy I²C host controller, the SPD5 Hub device offers two ways to address 1024 bytes of non volatile memory when it is operating in I²C mode only. By default, the SPD5 Hub device accepts 1 byte of address which covers first 128 bytes of non-volatile memory. The host must select the page pointer through volatile register Table 112, “MR11” [2:0] to address the entire 1024 bytes of non-volatile memory.

Alternatively, at initial power on, the host can set the Table 112, “MR11” [3] = ‘1’ to address the entire 1024 bytes of non-volatile memory with 2 bytes of address and hence not required to go through page selection to address entire non-volatile memory.

This 1 byte address mode is only applicable to SPD5 Hub device and it is not applicable to PMIC or TS or RCD device in I²C mode.

The SPD5 Hub device volatile register space does not require the page selection process as all volatile registers are within first 128 bytes.

2.6.8.1 Write Operation - Data Packet

The MemReg bit determines if the target of the transaction is an NVM location (MemReg = ‘1’) or an internal register (MemReg = ‘0’). When MemReg = ‘0’, there is no concept of “Block Address”; Block Address bits are treated simply Upper Address bits.

Table 6 — Write Command Data Packet; Table 13, “MR11” [3] = ‘0’

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|----------------------|--|--------------|---------------|-------|-------|-------|-------|-------|-----|------|
| S or Sr ¹ | 1 | 0 | 1 | 0 | HID | | | W=0 | A | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | A | |
| | Data | | | | | | | | A | |
| | ... | | | | | | | | A | |
| | Data | | | | | | | | A | |
| NOTE 1 | In I2C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I2C mode. Any other operation including another Repeat Start is considered an illegal operation. | | | | | | | | | |

2.6.8.1 Write Operation - Data Packet (cont'd)

Table 7 — Write Command Data Packet; Table 112, “MR11” [3] = ‘1’

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|----------------------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|-----|------|
| S or Sr ¹ | 1 | 0 | 1 | 0 | HID | | | W=0 | A | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | A | |
| | 0 | 0 | 0 | 0 | Blk Addr [4:1] ² | | | A | | |
| | Data | | | | | | | | A | |
| | ... | | | | | | | | A | |
| | Data | | | | | | | | A | |
| NOTE 1 | In I2C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I2C mode. Any other operation including another Repeat Start is considered an illegal operation. | | | | | | | | | |
| NOTE 2 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |

2.6.8.2 Read Operation - Data Packet

The MemReg bit determines if the target of the transaction is an NVM location (MemReg = ‘1’) or an internal register (MemReg = ‘0’). When MemReg = ‘0’, there is no concept of “Block Address”; Block Address bits are treated simply Upper Address bits.

Table 8 — Read Command Data Packet; Table 112, “MR11” [3] = ‘0’

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|----------------------|--|--------------|---------------|-------|-------|-------|-------|-------|----------------|------|
| S or Sr ¹ | 1 | 0 | 1 | 0 | HID | | | W=0 | A | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | A | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A ² | |
| | Data | | | | | | | | A | |
| | ... | | | | | | | | A | |
| | Data | | | | | | | | N | |
| NOTE 1 | In I ² C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation including another Repeat Start is considered an illegal operation. | | | | | | | | | |
| NOTE 2 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. The device may eventually ACK. | | | | | | | | | |

2.6.8.2 Read Operation - Data Packet (cont'd)

Table 9 — Read Command Data Packet; Table 112, “MR11” [3] = ‘1’

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|----------------------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|----------------|---------|
| S or Sr ¹ | 1 | 0 | 1 | 0 | HID | | | W=0 | A | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | A | |
| | 0 | 0 | 0 | 0 | Blk Addr [4:1] ² | | | A | | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A ³ | |
| | Data | | | | | | | | A | |
| | ... | | | | | | | | A | |
| | Data | | | | | | | | N | Sr or P |
| NOTE 1 | In I ² C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation including another Repeat Start is considered an illegal operation. | | | | | | | | | |
| NOTE 2 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 3 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times as it may desire. The device may eventually ACK. | | | | | | | | | |

2.6.8.3 Default Read Address Pointer Mode

During normal operation of the DDR5 DIMM, the host periodically may poll critical information from the same location. An example may be the SPD5 Hub device's temperature readout. To help improve the efficiency of the I²C bus protocol, the SPD5 Hub offers a default read address pointer mode so that whenever the SPD5 Hub device sees the STOP operation on its HSCL and HSDA bus, its read address pointer is always resets to default address. The default read pointer address mode is enabled through register Table 117, “MR18” [4] and default starting address for read operation is selectable through register Table 117, “MR18” [3:2]. This allows host to read the read command data packet as shown in Table 10 compared to as shown in Table 9. The default read address pointer reduces the packet overhead by 3 bytes. The host typically enables this mode when the normal operation of the DDR5 DIMM begins. The default read address pointer mode is only applicable to volatile register space (i.e., MemReg = ‘0’).

Table 10 — Read Command Data Packet with Default Address Pointer Mode

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|-----|---------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A | |
| | Data | | | | | | | | A | |
| | ... | | | | | | | | A | |
| | Data | | | | | | | | N | Sr or P |

2.6.9 I²C Target Protocol - Host to Local Device Through SPD5 Hub Device

2.6.9.1 Write Operation - Data Packet

Table 11 — Write Command Data Packet (e.g., TS)

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|---|---------------|-------|-------|-------|-------|-------|-------|-------|-----|---------|
| S or Sr ¹ | 0 | 0 | 1 | 0 | HID | | | W=0 | A | |
| | Address [7:0] | | | | | | | | A | |
| | Data | | | | | | | | A | |
| | Data | | | | | | | | A | |
| | ... | | | | | | | | A | |
| | Data | | | | | | | | A | |
| | | | | | | | | | | Sr or P |
| NOTE 1 In I ² C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation, including another Repeat Start, is considered an illegal operation. | | | | | | | | | | |

Table 12 — Write Command Data Packet (e.g., PMIC)

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|----------------------|--|-------|-------|-------|-------|-------|-------|-------|-----|---------|
| S or Sr ¹ | 1 | 0 | 0 | 1 | HID | | | W=0 | A | |
| | Address [7:0] | | | | | | | | A | |
| | Data | | | | | | | | A | |
| | Data | | | | | | | | A | |
| | ... | | | | | | | | A | |
| | Data | | | | | | | | A | |
| | | | | | | | | | | Sr or P |
| NOTE 1 | In I ² C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation, including another Repeat Start, is considered an illegal operation. | | | | | | | | | |

When Host makes any write request to the RCD device, the RCD protocol has PEC check in the last byte. The SPD simply treats the last byte as Data like any other bytes as shown in Table 14. The RCD device requires valid stable input clock (DCK_t, DCK_c), Reset_n and DCS_n to allow any read or write access on its I²C interface.

2.6.9.1 Write Operation - Data Packet (cont'd)**Table 13 — Write Command Data Packet (e.g., RCD; PEC Disabled)**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|---|-------------------------------------|-------|-------|-------|-------|-------|-------|-------|---------|------|
| S or Sr ¹ | 1 | 0 | 1 | 1 | HID | | | W=0 | A | |
| | Data (I ² C Bus Command) | | | | | | | | A | |
| | Data (Byte Count=8) | | | | | | | | A | |
| | Data (Reserved; 0x00) | | | | | | | | A | |
| | Data (Dev/Channel Num) | | | | | | | | A | |
| | Data (Page Num [7:0]) | | | | | | | | A | |
| | Data (Reg Num [7:0]) | | | | | | | | A | |
| | Data (Wr Data [31:24]) | | | | | | | | A | |
| | Data (Wr Data [23:16]) | | | | | | | | A | |
| | Data (Wr Data [15:8]) | | | | | | | | A | |
| Data (Wr Data [7:0]) | | | | | | | | A | Sr or P | |
| NOTE 1 In I ² C mode, Start or Repeat Start operation followed by 7 ^h 7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation, including another Repeat Start, is considered an illegal operation. | | | | | | | | | | |

Table 14 — Write Command Data Packet (e.g., RCD; PEC Enabled)

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|---|-------------------------------------|-------|-------|-------|-------|-------|-------|-------|-----|---------|
| S or Sr ¹ | 1 | 0 | 1 | 1 | HID | | | W=0 | A | |
| | Data (I ² C Bus Command) | | | | | | | | A | |
| | Data (Byte Count=8) | | | | | | | | A | |
| | Data (Reserved; 0x00) | | | | | | | | A | |
| | Data (Dev/Channel Num) | | | | | | | | A | |
| | Data (Page Num [7:0]) | | | | | | | | A | |
| | Data (Reg Num [7:0]) | | | | | | | | A | |
| | Data (Wr Data [31:24]) | | | | | | | | A | |
| | Data (Wr Data [23:16]) | | | | | | | | A | |
| | Data (Wr Data [15:8]) | | | | | | | | A | |
| | Data (Wr Data [7:0]) | | | | | | | | A | |
| | Data (PEC [7:0]) | | | | | | | | A | |
| NOTE 1 In I ² C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation, including another Repeat Start, is considered an illegal operation. | | | | | | | | | | Sr or P |

2.6.9.2 Read Operation - Data Packet

Table 15 — Read Command Data Packet (e.g., TS)

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|---|---------------|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S or Sr ¹ | 0 | 0 | 1 | 0 | HID | | | W=0 | A | |
| | Address [7:0] | | | | | | | | A | |
| Sr | 0 | 0 | 1 | 0 | HID | | | R=1 | A ² | |
| | Data | | | | | | | | A | |
| | Data | | | | | | | | A | |
| | ... | | | | | | | | A | |
| | Data | | | | | | | | N | |
| NOTE 1 In I ² C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation including, another Repeat Start, is considered an illegal operation. | | | | | | | | | | |
| NOTE 2 If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. | | | | | | | | | | |

Table 16 — Read Command Data Packet (e.g., PMIC)

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|----------------------|--|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S or Sr ¹ | 1 | 0 | 0 | 1 | HID | | | W=0 | A | |
| | Address [7:0] | | | | | | | | A | |
| Sr | 1 | 0 | 0 | 1 | HID | | | R=1 | A ² | |
| | Data | | | | | | | | A | |
| | Data | | | | | | | | A | |
| | ... | | | | | | | | A | |
| | Data | | | | | | | | N | |
| NOTE 1 | In I2C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I2C mode. Any other operation, including another Repeat Start, is considered an illegal operation. | | | | | | | | | |
| NOTE 2 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. | | | | | | | | | |

When Host makes any read request to the RCD device, the RCD protocol has PEC check for the RCD address information as well as data returned by the RCD. The SPD5 Hub simply treats the PEC information as data byte like any other data byte as shown in Table 18. The SPD5 Hub device does not check for the PEC. The RCD device requires valid stable input clock (DCK_t, DCK_c), Reset_n, and DCS_n to allow any read or write access on its I²C interface.

2.6.9.2 Read Operation - Data Packet (cont'd)**Table 17 — Read Command Data Packet (e.g., RCD; PEC Disabled; Legacy Format)**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|---|-------------------------------------|-------|-------|-------|-------|-------|-------|-------|----------------|---------|
| S or Sr ¹ | 1 | 0 | 1 | 1 | HID | | | W=0 | A | |
| | Data (I ² C Bus Command) | | | | | | | | A | |
| | Data (Byte Count=4) | | | | | | | | A | |
| | Data (Reserved; 0x00) | | | | | | | | A | |
| | Data (Dev/Channel Num) | | | | | | | | A | |
| | Data (Page Num [7:0]) | | | | | | | | A | |
| | Data (Reg Num [7:0]) | | | | | | | | A | |
| S | 1 | 0 | 1 | 1 | HID | | | W=0 | A | |
| | Data (I ² C Bus Command) | | | | | | | | A | |
| Sr | 1 | 0 | 1 | 1 | HID | | | R=1 | A ² | |
| | Data (Byte Count) | | | | | | | | A | |
| | Data (Status) | | | | | | | | A | |
| | Data (Rd Data [31:24]) | | | | | | | | A | |
| | Data (Rd Data [23:16]) | | | | | | | | A | |
| | Data (Rd Data [15:8]) | | | | | | | | A | |
| | Data (Rd Data [7:0]) | | | | | | | | A | Sr or P |
| NOTE 1 In I ² C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation, including another Repeat Start, is considered an illegal operation. | | | | | | | | | | |
| NOTE 2 If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. | | | | | | | | | | |

2.6.9.2 Read Operation - Data Packet (cont'd)

Table 18 — Read Command Data Packet (e.g., RCD; PEC Enabled; Legacy Format)

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|---|-------------------------------------|-------|-------|-------|-------|-------|-------|-------|----------------|---------|
| S or Sr ¹ | 1 | 0 | 1 | 1 | HID | | | W=0 | A | |
| | Data (I ² C Bus Command) | | | | | | | | A | |
| | Data (Byte Count=4) | | | | | | | | A | |
| | Data (Reserved; 0x00) | | | | | | | | A | |
| | Data (Dev/Channel Num) | | | | | | | | A | |
| | Data (Page Num [7:0]) | | | | | | | | A | |
| | Data (Reg Num [7:0]) | | | | | | | | A | |
| | Data (PEC [7:0]) | | | | | | | | A | P |
| S | 1 | 0 | 1 | 1 | HID | | | W=0 | A | |
| | Data (I ² C Bus Command) | | | | | | | | A | |
| Sr | 1 | 0 | 1 | 1 | HID | | | R=1 | A ² | |
| | Data (Byte Count) | | | | | | | | A | |
| | Data (Status) | | | | | | | | A | |
| | Data (Rd Data [31:24]) | | | | | | | | A | |
| | Data (Rd Data [23:16]) | | | | | | | | A | |
| | Data (Rd Data [15:8]) | | | | | | | | A | |
| | Data (Rd Data [7:0]) | | | | | | | | A | |
| | Data (PEC [7:0]) | | | | | | | | A | Sr or P |
| NOTE 1 In I ² C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation, including another Repeat Start, is considered an illegal operation. | | | | | | | | | | |
| NOTE 2 If Target device NACKs during Repeat Start for any reason, the host my re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. | | | | | | | | | | |

2.6.9.2 Read Operation - Data Packet (cont'd)

Table 19 — Read Command Data Packet (e.g., RCD; PEC Disabled; Optimized Format)

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N | Stop |
|----------------------|--|-------|-------|-------|-------|-------|-------|-------|----------------|---------|
| S or Sr ¹ | 1 | 0 | 1 | 1 | HID | | | W=0 | A | |
| | Data (I ² C Bus Command) | | | | | | | | A | |
| | Data (Byte Count=4) | | | | | | | | A | |
| | Data (Reserved; 0x00) | | | | | | | | A | |
| | Data (Dev/Channel Num) | | | | | | | | A | |
| | Data (Page Num [7:0]) | | | | | | | | A | |
| | Data (Reg Num [7:0]) | | | | | | | | A | |
| Sr | 1 | 0 | 1 | 1 | HID | | | R=1 | A ² | |
| | Data (Byte Count) | | | | | | | | A | |
| | Data (Status) | | | | | | | | A | |
| | Data (Rd Data [31:24]) | | | | | | | | A | |
| | Data (Rd Data [23:16]) | | | | | | | | A | |
| | Data (Rd Data [15:8]) | | | | | | | | A | |
| | Data (Rd Data [7:0]) | | | | | | | | A | Sr or P |
| NOTE 1 | In I ² C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I ² C mode. Any other operation, including another Repeat Start, is considered an illegal operation. | | | | | | | | | |
| NOTE 2 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. | | | | | | | | | |

2.6.10 I3C Basic Target Protocol - Host to SPD5 Hub Device

2.6.10.1 Write Operation Data Packet

The SPD5 Hub devices operate on a standard I3C serial interface. Transactions where the SPD5 Hub device is the targeted Target device begin with the Host issuing a START condition followed by a 7-bit SPD5 Hub device address then a write bit, RW. All data are transmitted with the most significant bit MSB first. During the address followed by R/W bit transmission, the SPD5 Hub device typically replies with an ACK unless there are conditions when it may passively assert a NACK. The “T” bit carries Parity information from the Host for each byte.

The SPD5 Hub device does not allow any read or write operations to its non-volatile memory when it is busy with internal write operation to non-volatile memory. The host should either check Table 130, “MR48” [3] register status or ensure that Write time (t_w) parameter is satisfied prior to performing another write or read to operations non-volatile memory. If host violates this and performs either write or read operation to non-volatile memory then SPD5 Hub device always ACKs the address byte. In a subsequent byte, the device ignores the write operation if MemReg = ‘1’ and sets the Table 134, “MR52” [7] = ‘1’ and requests IBI if IBI function is enabled. Subsequently, if host continues to do the read operation with Repeat Start, the SPD5 Hub device NACKs.

The SPD5 Hub device does allow any read or write operation to its volatile memory (i.e., MemReg = ‘0’) when it is busy with internal write operation to non-volatile memory.

2.6.10.1 Write Operation Data Packet (cont'd)

The Packet Error Code (PEC) function is disabled by default when the SPD5 Hub device is put in I3C Basic mode. The host may optionally enable this function through Table 117, “MR18” [7] or DEVCTRL CCC. If enabled, the PEC is appended at the end of all transactions. If PEC is enabled, the host must complete the burst length as indicated in CMD field. In other words, the host must not interrupt the burst length pre-maturely for Write operation.

The MemReg bit determines if the target of the transaction is an NVM location (MemReg = ‘1’) or an internal register (MemReg = ‘0’). When MemReg = ‘0’, there is no concept of “Block Address”; Block Address bits are treated simply Upper Address bits.

Table 20 — Write Command Data Packet; PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | 0 | 0 | 0 | 0 | Blk Addr [4:1] ⁴ | | | | T | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | Sr ⁵ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The SPD5 Hub device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 5 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 21 — Write Command Data Packet; PEC Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | CMD | | | W=0 | Blk Addr [4:1] ⁴ | | | | T | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | |
| | PEC | | | | | | | | T | Sr ⁵ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The SPD5 Hub device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation | | | | | | | | | |
| NOTE 4 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 5 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.10.1 Write Operation Data Packet (cont'd)

The host may optionally allow SPD5 Hub device to request IBI. For this case, the transactions to the SPD5 Hub device begin with the I3C Basic host issuing a START condition followed by 7'h7E and then write bit. If SPD5 Hub device has a pending IBI, it transmits its 7-bit device select code followed by R=1. If SPD5 Hub device has no pending IBI, there is no action taken by SPD5 Hub. Table 22 and Table 23 show the I3C bus write command data packet with optional IBI header for PEC disabled and PEC enabled case respectively. Note that in Table 23, PEC calculation does not include IBI header byte (7'h7E followed by W=0).

Table 22 — Write Command Data Packet with IBI Header; PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ^{1,3} | |
| Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{2,3,4} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | 0 | 0 | 0 | 0 | Blk Addr [4:1] ⁵ | | | T | | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start). | | | | | | | | | |
| NOTE 2 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK) and Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 3 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 4 | The SPD5 Hub device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 5 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 6 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.10.1 Write Operation Data Packet (cont'd)

Table 23 — Write Command Data Packet with IBI Header; PEC Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--------|---|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|----------------------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ^{1,3} | |
| Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{2,3,4} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | CMD | | | W=0 | Blk Addr [4:1] ⁵ | | | | T | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | |
| | PEC | | | | | | | | T | Sr ⁶ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start). | | | | | | | | | |
| NOTE 2 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK) and Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 3 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 4 | The SPD5 Hub device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 5 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 6 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.10.2 Read Operation Data Packet

The transactions to SPD5 Hub Target device begin with the I3C Basic Host issuing a START condition followed by a 7-bit SPD5 Hub device type identifier then a write bit. All I3C bus data are transmitted with the most significant bit MSB first. During select code transmission, the SPD5 Hub device typically replies with an ACK unless there are exceptional conditions when it may passively assert a NACK. See Table 24. The “T” bit carries Parity information from the host prior to Repeated START. After Repeated START, “T” bit carries information from SPD5 Hub device to Host indicating Continuous (‘1’) or Stop (0’) whether it is transmitting the last byte or not.

The Packet Error Code (PEC) function is disabled by default when SPD5 Hub device is put in I3C Basic mode. The host may optionally enable this function through Table 117, “MR18” [7] or DEVCTRL CCC. If enabled, the PEC is appended as shown in Table 25. If PEC is enabled, the host must complete the burst length as indicated in CMD field. In other words, the host must not interrupt the burst length pre-maturely for Read operation

The MemReg bit determines if the target of the transaction is an NVM location (MemReg = ‘1’) or an internal register (MemReg = ‘0’). When MemReg = ‘0’, there is no concept of “Block Address”; Block Address bits are treated simply Upper Address bits.

2.6.10.2 Read Operation Data Packet (cont'd)**Table 24 — Read Command Data Packet; PEC Disabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | 0 | 0 | 0 | 0 | Blk Addr [4:1] ⁴ | | | T | | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A/N ^{5,6} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 ^{7,8} | Sr ⁹ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The SPD5 Hub device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | The memory size of SPD5 hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 5 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity errors the SPD5 Hub may eventually ACK. | | | | | | | | | |
| NOTE 6 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 7 | See Figure 10 to see how Host ends Target device operation. | | | | | | | | | |
| NOTE 8 | For NVM read memory access (i.e., MemReg = '1'), when last byte is reached (1024 Byte) or for volatile memory access (i.e., MemReg = '0'), when last byte (i.e., MR255) is reached (extreme rare case), the Target device sends T = '0'. See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 9 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.10.2 Read Operation Data Packet (cont'd)

Table 25 — Read Command Data Packet; PEC Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | CMD | | | R=1 | Blk Addr [4:1] ⁴ | | | | T | |
| | PEC | | | | | | | | T | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A/N ^{5,6} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 | |
| | PEC | | | | | | | | T=0 ⁷ | Sr ⁸ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The SPD5 Hub device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 5 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to PEC error parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity or PEC errors, the SPD5 Hub may eventually ACK. The PEC calculation by the Target device only includes device select code of the ACK response of the Repeat start operation. In other words, if there are more than one Repeat Start operation, the Target device includes device select code of only the last Repeat Start from the Host when it ACKs in PEC calculation and all other NACK responses of the device select code of the Repeat Start are not included in PEC calculation. | | | | | | | | | |
| NOTE 6 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 7 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 8 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

The host may optionally allow SPD5 Hub device to request IBI. For this case, the transactions to the SPD5 Hub device begin with the I3C Basic host issuing a START condition followed by 7'h7E and then write bit. If SPD5 Hub device has a pending IBI, it transmits its 7-bit device select code followed by R=1. If SPD5 Hub device has no pending IBI, there is no action taken by SPD5 Hub. Table 26 and Table 27 show the I3C Basic bus read command data packet with optional IBI header for PEC disabled and PEC enabled case respectively. Note that in Table 27, PEC calculation (from Host to SPD5 Hub) does not include IBI header byte (7'h7E followed by W=0).

2.6.10.2 Read Operation Data Packet (cont'd)**Table 26 — Read Command Data Packet with IBI Header; PEC Disabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|-----------------------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ^{1,3} | |
| Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{2,3,4} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | 0 | 0 | 0 | 0 | Blk Addr [4:1] ⁵ | | | T | | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A/N ^{6,7} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 ^{8,9} | Sr ¹⁰ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start). | | | | | | | | | |
| NOTE 2 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK) and Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 3 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 4 | The SPD5 Hub device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 5 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 6 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 7 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity errors, the SPD5 Hub may eventually ACK. | | | | | | | | | |
| NOTE 8 | See Figure 10 to see how Host ends Target device operation. | | | | | | | | | |
| NOTE 9 | For NVM read memory access (i.e., MemReg = '1'), when last byte is reached (1024 Byte) or for volatile memory access (i.e., MemReg = '0'), when last byte (i.e., MR255) is reached (extreme rare case), the Target device sends T = '0'. See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 10 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.10.2 Read Operation Data Packet (cont'd)

Table 27 — Read Command Data Packet with IBI Header; PEC Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|----------------------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ^{1,3} | |
| Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{2,3,4} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | CMD | | | R=1 | Blk Addr [4:1] ⁵ | | | | T | |
| | PEC | | | | | | | | T | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A/N ^{6,7} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 | |
| | PEC | | | | | | | | T=0 ⁸ | Sr ⁹ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start). | | | | | | | | | |
| NOTE 2 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK) and See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 3 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 4 | The SPD5 Hub device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 5 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 6 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 7 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to PEC error or parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity or PEC errors, the SPD5 Hub may eventually ACK. The PEC calculation by the Target device only includes device select code of the ACK response of the Repeat start operation. In other words, if there are more than one Repeat Start operation, the Target device includes device select of only the last Repeat Start from the Host when it ACKs in PEC calculation and all other NACK responses of the device select code of the Repeat Start are not included in PEC calculation. | | | | | | | | | |
| NOTE 8 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 9 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.10.3 Default Read Address Pointer Mode

This mode works the same exact way as explained in clause 2.6.8.3. Table 28 and Table 29 show the read command data packet for PEC function disabled and enabled respectively. When PEC function is enabled, Table 117, “MR18” [1] sets the number of bytes that SPD5 Hub device sends out followed by the PEC calculation. If PEC is enabled, the host must complete the burst length as indicated in Table 117, “MR18” [1] register. In other words, the host must not interrupt the burst length pre-maturely for default address pointer read operation. The default read address pointer mode is only applicable to volatile register space (i.e., MemReg = ‘0’).

2.6.10.3 Default Read Address Pointer Mode (cont'd)**Table 28 — Read Command Data Packet with Read Address Pointer Mode; PEC Disabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A ¹ | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 ^{2,3} | Sr ⁴ or P |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | See Figure 10 to see how Host ends Target device operation. | | | | | | | | | |
| NOTE 3 | When last byte (i.e., MR255) is reached (extreme rare case), the Target device sends T = '0'. See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 4 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 29 — Read Command Data Packet with Read Address Pointer Mode; PEC Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|------------------|------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A ¹ | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 | |
| | PEC | | | | | | | | T=0 ² | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | See Figure 11 to see how Target device ends the operation followed by STOP operation | | | | | | | | | |
| NOTE 3 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 30 — Read CMD Data Packet with Read Address Pointer Mode and IBI Header; PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--------|--|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ^{1,2} | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A ^{2,3} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 ^{4,5} | Sr ⁶ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 4 | See Figure 10 to see how Host ends Target device operation. | | | | | | | | | |
| NOTE 5 | When last byte (i.e., MR255) is reached (extreme rare case), the Target device sends T = '0'. See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 6 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.10.3 Default Read Address Pointer Mode (cont'd)

Table 31 — Read CMD Data Packet with Read Address Pointer Mode and IBI Header; PEC Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--------|---|-------|-------|-------|-------|-------|-------|-------|------------------|----------------------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ^{1,2} | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A ^{2,3} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 | |
| | PEC | | | | | | | | T=0 ⁴ | Sr ⁵ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 4 | See Figure 11 to see how Target device ends the operation followed by STOP operation | | | | | | | | | |
| NOTE 5 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11 I3C Basic Target Protocol - Host to Local Device (Through SPD5 Hub Device)

2.6.11.1 Write Operation - Data Packet

Table 32 to Table 37 show examples of write command data packet for different types of local devices behind the SPD5 Hub. These example do not show the optional IBI header byte that Host may choose to use. All local devices behind SPD5 Hub device also supports the IBI header byte similar to as shown in Table 22 and Table 23. Refer to the device specific data sheet.

Table 32 — Write Command Data Packet (e.g., TS); PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 0 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | Address [7:0] | | | | | | | | T | |
| | Data | | | | | | | | T | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | Sr ⁴ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Addr; bit [7]). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The TS device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The TS device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.1 Write Operation - Data Packet (cont'd)**Table 33 — Write Command Data Packet (e.g., TS); PEC Enabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 0 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | Address [7:0] | | | | | | | | T | |
| | CMD | | | W=0 | 0 | 0 | 0 | 0 | T | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | |
| | PEC | | | | | | | | T | Sr ⁴ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Addr, bit [7]). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The TS device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The TS device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 34 — Write Command Data Packet (e.g., PMIC); PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 0 | 1 | HID | | | W=0 | A ^{1,2,3} | |
| | Address [7:0] | | | | | | | | T | |
| | Data | | | | | | | | T | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | Sr ⁴ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Addr, bit [7]). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The PMIC device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The PMIC device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.1 Write Operation - Data Packet (cont'd)

Table 35 — Write Command Data Packet (e.g., PMIC); PEC Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 0 | 1 | HID | | | W=0 | A ^{1,2,3} | |
| | Address [7:0] | | | | | | | | T | |
| | CMD | | | W=0 | 0 | 0 | 0 | 0 | T | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | |
| | PEC | | | | | | | | T | Sr ⁴ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Addr, bit [7]). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The PMIC device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The PMIC device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

The RCD device requires valid stable input clock (DCK_t, DCK_c), Reset_n, and DCS_n to allow any read or write access on its I3C Basic interface.

Table 36 — Write Command Data Packet (e.g., RCD); PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 1 | HID | | | W=0 | A ^{1,2,3} | |
| | Data (I3C Bus Command) | | | | | | | | T | |
| | Data (Byte Count=8) | | | | | | | | T | |
| | Data (Reserved; 0x00) | | | | | | | | T | |
| | Data (Dev/Channel Num) | | | | | | | | T | |
| | Data (Page Num [7:0]) | | | | | | | | T | |
| | Data (Reg Num [7:0]) | | | | | | | | T | |
| | Data (Wr Data [31:24]) | | | | | | | | T | |
| | Data (Wr Data [23:16]) | | | | | | | | T | |
| | Data (Wr Data [15:8]) | | | | | | | | T | |
| | Data (Wr Data [7:0]) | | | | | | | | T | Sr ⁴ or P |
| NOTE1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Data, bit [7]). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The RCD device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The RCD device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.1 Write Operation - Data Packet (cont'd)**Table 37 — Write Command Data Packet (e.g., RCD); PEC Enabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 1 | HID | | | W=0 | A ^{1,2,3} | |
| | Data (I3C Bus Command) | | | | | | | | T | |
| | Data (Byte Count=8) | | | | | | | | T | |
| | Data (Reserved; 0x00) | | | | | | | | T | |
| | Data (Dev/Channel Num) | | | | | | | | T | |
| | Data (Page Num [7:0]) | | | | | | | | T | |
| | Data (Reg Num [7:0]) | | | | | | | | T | |
| | Data (Wr Data [31:24]) | | | | | | | | T | |
| | Data (Wr Data [23:16]) | | | | | | | | T | |
| | Data (Wr Data [15:8]) | | | | | | | | T | |
| | Data (Wr Data [7:0]) | | | | | | | | T | |
| | Data (PEC [7:0]) | | | | | | | | T | Sr ⁴ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Data, bit [7]). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The RCD device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The RCD device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.2 Read Operation - Data Packet

Table 38 to Table 43 show examples read command data packet for different types of local devices behind the SPD5 Hub. These example do not show the optional IBI header byte that Host may choose to use. All local devices behind SPD5 Hub device also supports the IBI header byte similar to as shown in Table 26 and Table 27. Refer to the device specific data sheet.

Table 38 — Read Command Data Packet (e.g., TS); PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|--------------------|------|
| S or Sr | 0 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | Address [7:0] | | | | | | | | T | |
| Sr | 0 | 0 | 1 | 0 | HID | | | R=1 | A/N ^{4,5} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 ^{6,7} | |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Addr, bit [7]). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The TS device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The TS device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity errors, the device may eventually ACK. | | | | | | | | | |
| NOTE 5 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 6 | See Figure 10 to see how Host ends Target device operation. | | | | | | | | | |
| NOTE 7 | For volatile register access, when last byte (MR127) is reached (extreme rare case), the Target device sends T = '0'. See Figure 11 to see how Host ends Target device operation. | | | | | | | | | |
| NOTE 8 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.2 Read Operation - Data Packet (cont'd)**Table 39 — Read Command Data Packet (e.g., TS); PEC Enabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 0 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | Address [7:0] | | | | | | | | T | |
| | CMD | | | R=1 | 0 | 0 | 0 | 0 | T | |
| | PEC | | | | | | | | T | |
| Sr | 0 | 0 | 1 | 0 | HID | | | R=1 | A/N ^{4,5} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 | |
| | PEC | | | | | | | | T=0 ⁶ | Sr ⁷ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Addr, bit [7]). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The TS device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The TS device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to PEC error or parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity or PEC errors, the device may eventually ACK. The PEC calculation by the Target device only includes device select code of the ACK response of the Repeat start operation. In other words, if there are more than one Repeat Start operation, the Target device includes device select of only the last Repeat Start from the Host when it ACKs in PEC calculation and all other NACK responses of the device select code of the Repeat Start are not included in PEC calculation. | | | | | | | | | |
| NOTE 5 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 6 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 7 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.2 Read Operation - Data Packet (cont'd)

Table 40 — Read Command Data Packet (e.g., PMIC); PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|--------------------|------|
| S or Sr | 1 | 0 | 0 | 1 | HID | | | W=0 | A ^{1,2,3} | |
| | Address [7:0] | | | | | | | | T | |
| Sr | 1 | 0 | 0 | 1 | HID | | | R=1 | A/N ^{4,5} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 ^{6,7} | |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Addr, bit [7]). | | | | | | | | | |
| NOTE 2 | The PMIC NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The PMIC device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The PMIC device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity errors, the device may eventually ACK. | | | | | | | | | |
| NOTE 5 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 6 | See Figure 10 to see how Host ends Target device operation. | | | | | | | | | |
| NOTE 7 | For volatile register access, when last byte is reached within the region (either Host region or DIMM Vendor region), it will continue to return the data but returned data is 0x00 if there is no valid password for DIMM vendor region or device vendor specific region. Once the address counter reaches R255, it resets to address R00 and it continues to return the data. Only Host can perform the STOP operation. | | | | | | | | | |
| NOTE 8 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.2 Read Operation - Data Packet (cont'd)**Table 41 — Read Command Data Packet (e.g., PMIC); PEC Enabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|--------------------|------|
| S or Sr | 1 | 0 | 0 | 1 | HID | | | W=0 | A ^{1,2,3} | |
| | Address [7:0] | | | | | | | | T | |
| | CMD | | | R=1 | 0 | 0 | 0 | 0 | T | |
| | PEC | | | | | | | | T | |
| Sr | 1 | 0 | 0 | 1 | HID | | | R=1 | A/N ^{4,5} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 | |
| | PEC | | | | | | | | T=0 ⁶ | |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Addr, bit [7]). | | | | | | | | | |
| NOTE 2 | The PMIC NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The PMIC device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The PMIC device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to PEC error or parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity or PEC errors, the device may eventually ACK. The PEC calculation by the Target device only includes device select code of the ACK response of the Repeat start operation. In other words, if there are more than one Repeat Start operation, the Target device includes device select of only the last Repeat Start from the Host when it ACKs in PEC calculation and all other NACK responses of the device select code of the Repeat Start are not included in PEC calculation. | | | | | | | | | |
| NOTE 5 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 6 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 7 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.2 Read Operation - Data Packet (cont'd)

The RCD device requires valid stable input clock (DCK_t, DCK_c), Reset_n, and Chip Select input (DCS_n) to allow any read or write access on its I3C Basic interface.

Table 42 — Read Command Data Packet (e.g., RCD); PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 1 | HID | | | W=0 | A ^{1,2,3} | |
| | Data (I3C Bus Command) | | | | | | | | T | |
| | Data (Byte Count=4) | | | | | | | | T | |
| | Data (Reserved; 0x00) | | | | | | | | T | |
| | Data (Dev/Channel Num) | | | | | | | | T | |
| | Data (Page Num [7:0]) | | | | | | | | T | |
| | Data (Reg Num [7:0]) | | | | | | | | T | |
| Sr | 1 | 0 | 1 | 1 | HID | | | R=1 | A/N ^{4,5} | |
| | Data (Byte Count) | | | | | | | | T=1 | |
| | Data (Status) | | | | | | | | T=1 | |
| | Data (Wr Data [31:24]) | | | | | | | | T=1 | |
| | Data (Wr Data [23:16]) | | | | | | | | T=1 | |
| | Data (Wr Data [15:8]) | | | | | | | | T=1 | |
| | Data (Wr Data [7:0]) | | | | | | | | T=0 ⁶ | Sr ⁷ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Data). | | | | | | | | | |
| NOTE 2 | The RCD NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The RCD device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The RCD device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE4 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity errors, the device may eventually ACK. | | | | | | | | | |
| NOTE 5 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 6 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 7 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.2 Read Operation - Data Packet (cont'd)**Table 43 — Read Command Data Packet (e.g., RCD); PEC Enabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 1 | HID | | | W=0 | A ^{1,2,3} | |
| | Data (I3C Bus Command) | | | | | | | | T | |
| | Data (Byte Count=4) | | | | | | | | T | |
| | Data (Reserved; 0x00) | | | | | | | | T | |
| | Data (Dev/Channel Num) | | | | | | | | T | |
| | Data (Page Num [7:0]) | | | | | | | | T | |
| | Data (Reg Num [7:0]) | | | | | | | | T | |
| Sr | 1 | 0 | 1 | 1 | HID | | | R=1 | A/N ^{4,5} | |
| | Data (Byte Count) | | | | | | | | T=1 | |
| | Data (Status) | | | | | | | | T=1 | |
| | Data (Wr Data [31:24]) | | | | | | | | T=1 | |
| | Data (Wr Data [23:16]) | | | | | | | | T=1 | |
| | Data (Wr Data [15:8]) | | | | | | | | T=1 | |
| | Data (Wr Data [7:0]) | | | | | | | | T=1 | |
| | Data (PEC [7:0]) | | | | | | | | T=0 ⁶ | Sr ⁷ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Data). | | | | | | | | | |
| NOTE 2 | The RCD NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The RCD device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The RCD device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity or PEC errors, the device may eventually ACK. The PEC calculation by the Target device only includes device select code of the ACK response of the Repeat start operation. In other words, if there are more than one Repeat Start operation, the Target device includes device select of only the last Repeat Start from the Host when it ACKs in PEC calculation and all other NACK responses of the device select code of the Repeat Start are not included in PEC calculation. | | | | | | | | | |
| NOTE 5 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 6 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 7 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.11.2 Read Operation - Data Packet (cont'd)

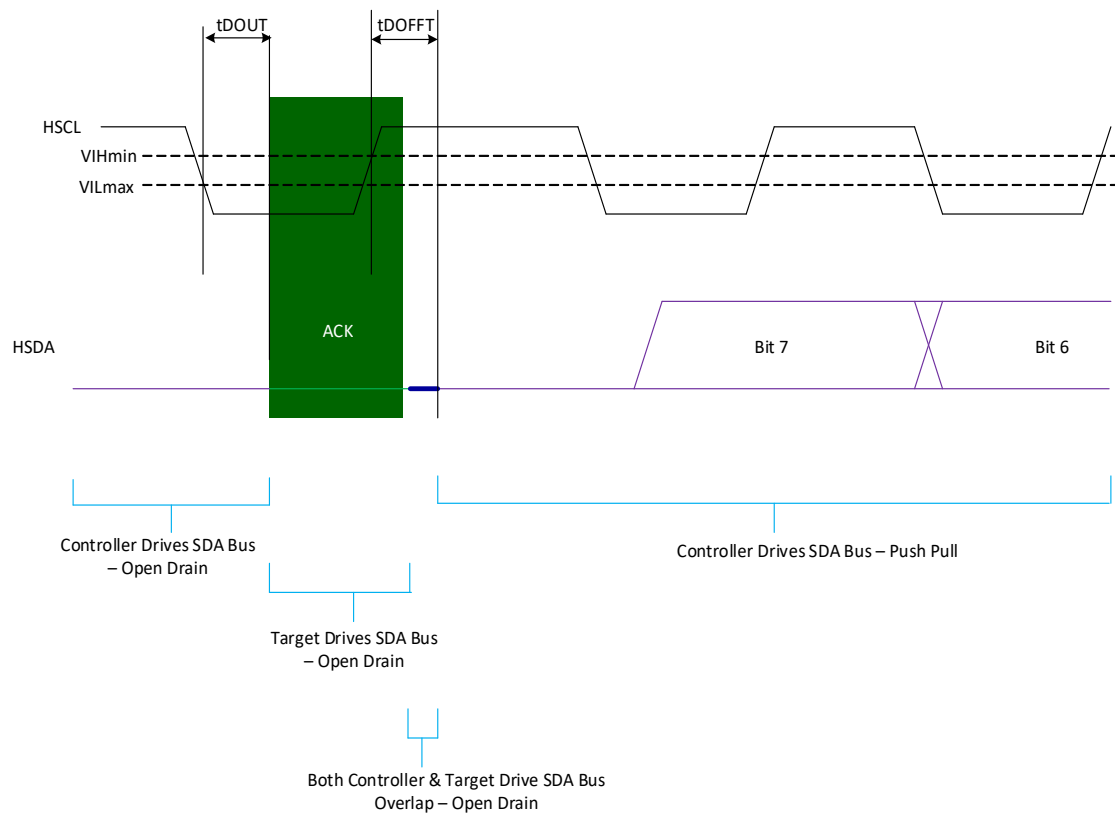
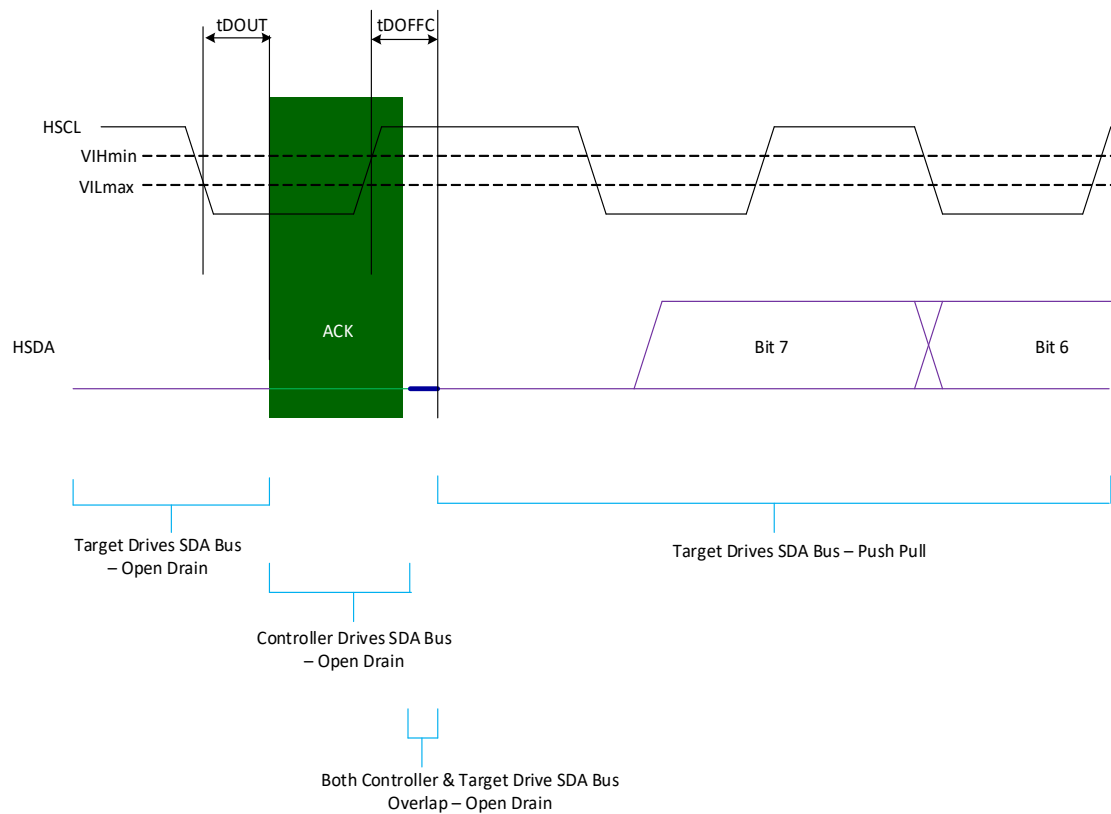


Figure 7 — Target Open Drain to Host Push Pull Hand Off Operation

2.6.11.2 Read Operation - Data Packet (cont'd)**Figure 8 — Controller Open Drain (ACK) to Target Push Pull Hand Off Operation**

2.6.11.2 Read Operation - Data Packet (cont'd)

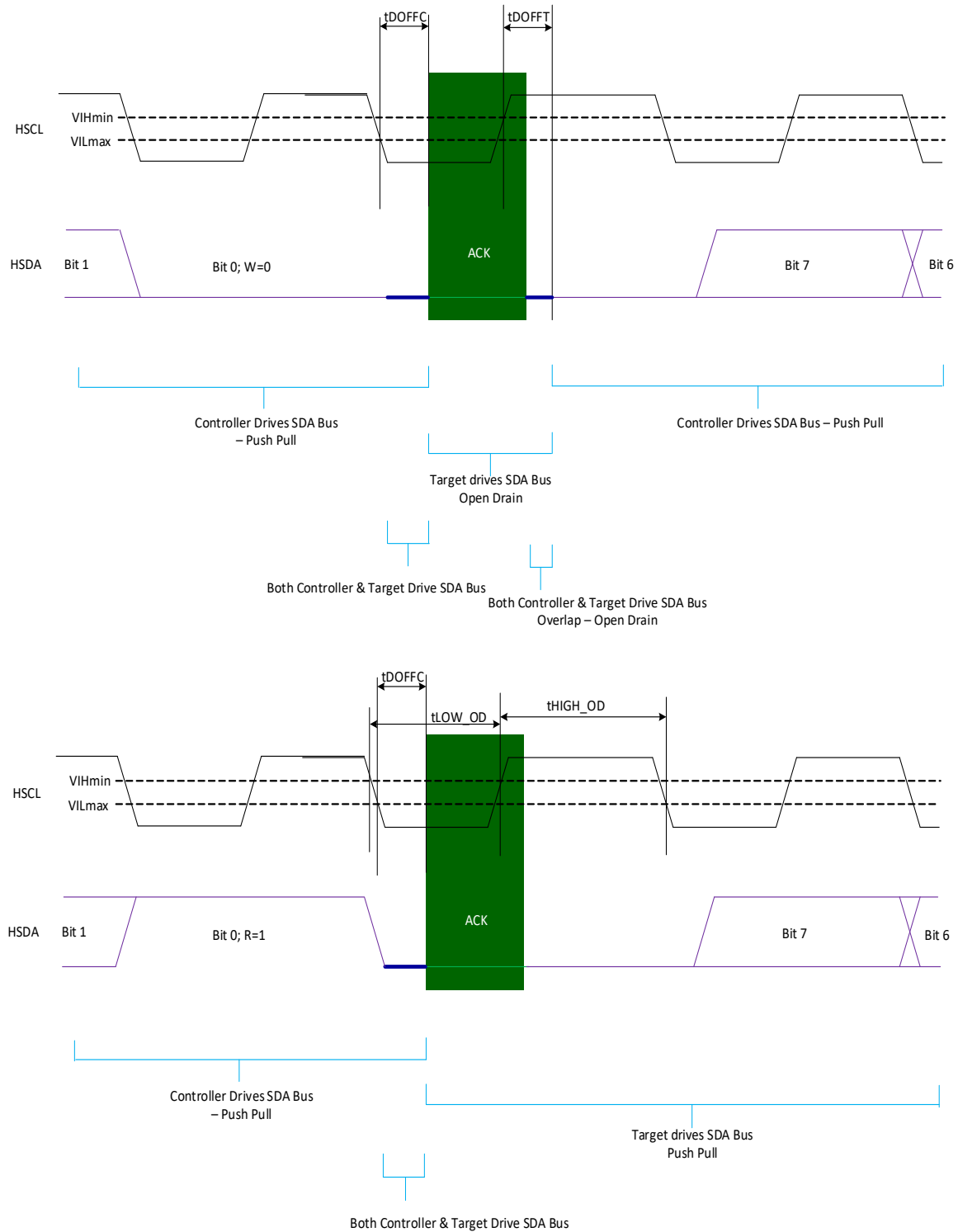


Figure 9 — Controller Push Pull to Target Open Drain Hand Off Operation

2.6.11.2 Read Operation - Data Packet (cont'd)

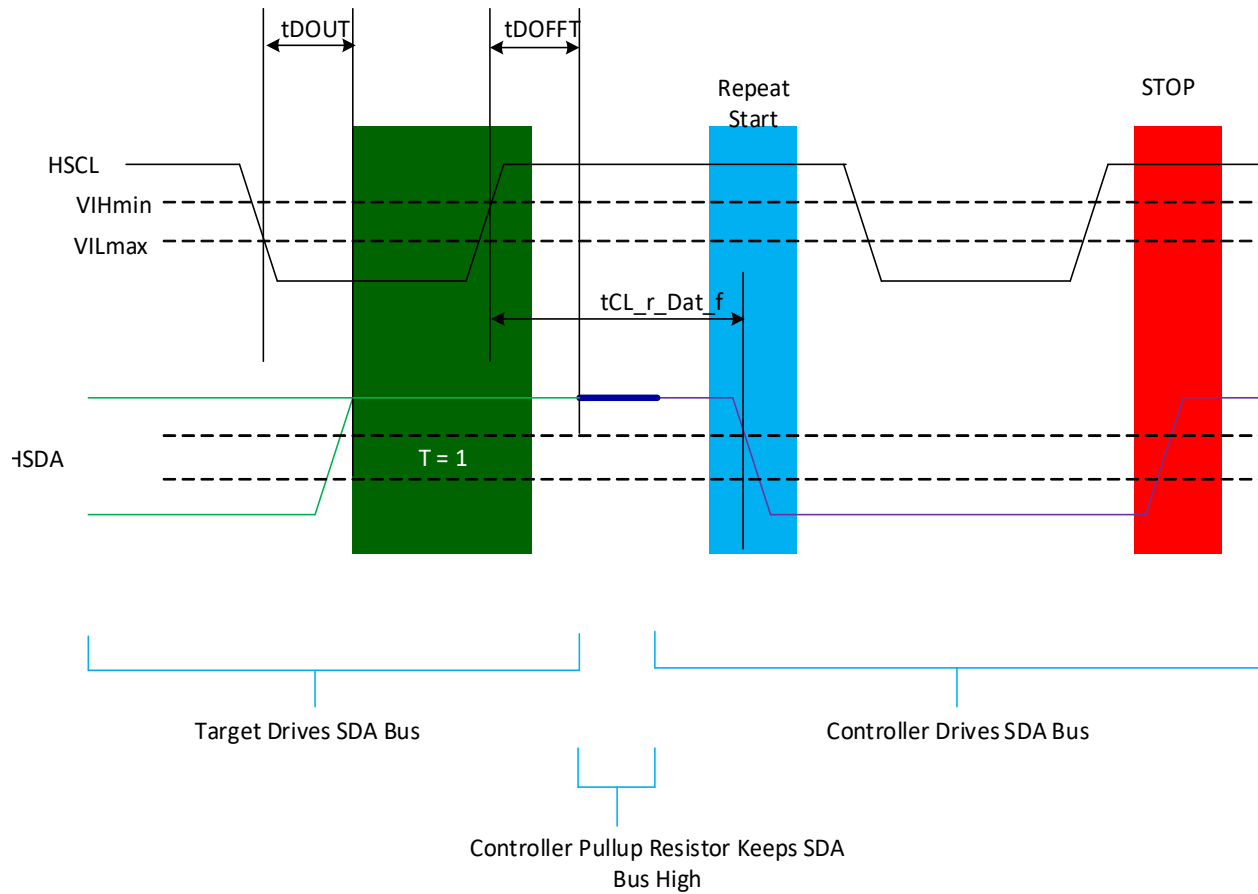


Figure 10 — T=1; Host Ends Read with Repeated START and STOP Waveform

2.6.11.2 Read Operation - Data Packet (cont'd)

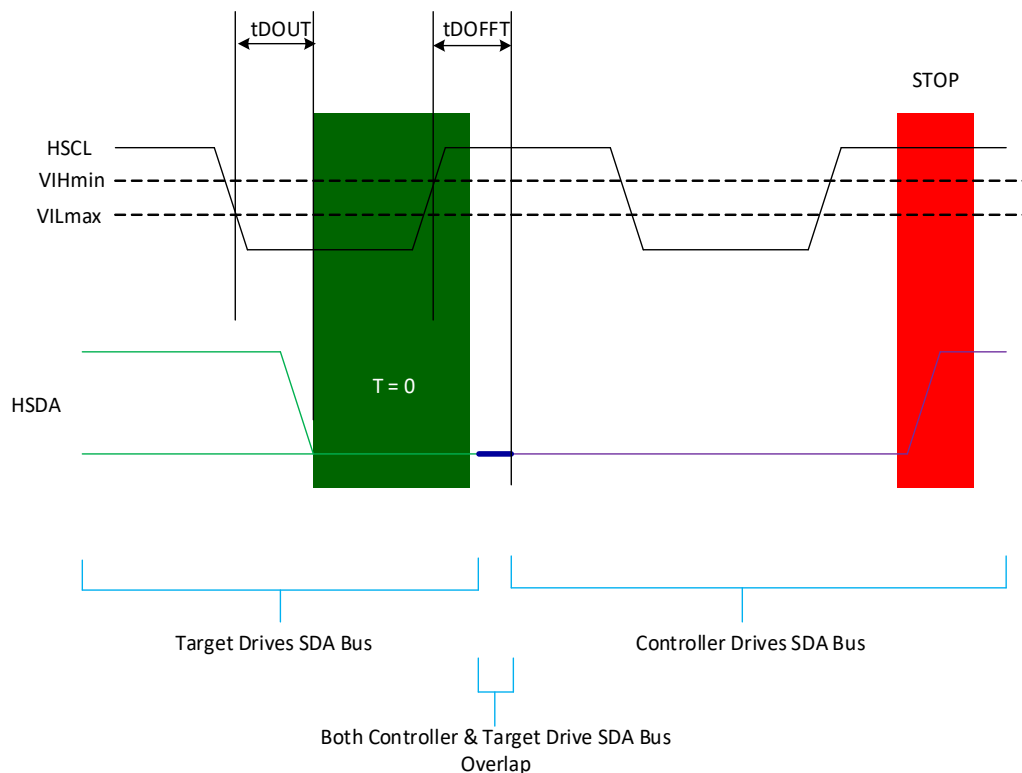


Figure 11 — T=0; Target Ends Read; Host Generates STOP

2.6.12 In Band Interrupt (IBI)

In I²C mode, in band interrupt function is not supported. Only I3C Basic mode supports in band interrupt function.

2.6.12.1 Enabling and Disabling In Band Event Interrupt Function

By default, all interrupt sources are disabled (i.e., set to '0'). The host may enable following interrupts in the SPD5 Hub device. Once enabled, the SPD5 Hub device sends an IBI when that event occurs.

1. Error Interrupt Enable in Table 121, "MR27" [4]:
 - a. When Table 121, "MR27" [4] = '1', the device sends the IBI at next available opportunity when any of the register bit in Table 134, "MR52" [7:5, 1:0] is set to '1' and sets Table 130, "MR48" [7] = '1' and updates Pending Interrupt Bits [3:0] = '0001' for GETSTATUS CCC.
 - b. When Table 121, "MR27" [4] = '0', the device does not send the IBI regardless of the register bit status in Table 134, "MR52" [7:5, 1:0]. However, the device does set Table 130, "MR48" [7] = '1' and updates Pending Interrupt Bits [3:0] = '0001' for GETSTATUS CCC.
2. Temperature Sensor Interrupt Enable in Table 121, "MR27" [3:0]: The host can set any combination of register bits to '1'.
 - a. When any of the register bits in Table 121, "MR27" [3:0] = '1' and if Table 121, "MR27" [4] = '1', the device sends the IBI at next available opportunity when the corresponding register bit in Table 133, "MR51" [3:0] is set to '1' and sets Table 130, "MR48" [7] = '1' and updates Pending Interrupt Bits [3:0] = '0001' for GETSTATUS CCC.

2.6.12.1 Enabling and Disabling In Band Event Interrupt Function (cont'd)

- b. When any of the register bits in Table 121, “MR27” [3:0] = ‘0’ or Table 121, “MR27” [4] = ‘0’, the device does not send the interrupt regardless of the corresponding register bit status in Table 133, “MR51” [3:0]. However, the device does set Table 130, “MR48” [7] = ‘1’ and updates Pending Interrupt Bits [3:0] = ‘0001’ for GETSTATUS CCC if any of the bits in Table 121, “MR27” [3:0] = ‘1’ and Table 121, “MR27” [4] = ‘0’.

2.6.12.2 Mechanics of Interrupt Generation - SPD5 Hub Device

Event interrupts may be generated by the SPD5 Hub device if IBI is enabled. When there is a pending interrupt (i.e., Table 130, “MR48” [7] = ‘1’ and Table 121, “MR27” [4] = ‘1’), the SPD5 Hub requests an interrupt after detecting START condition by transmitting its 7-bit binary address (LID bits followed by HID bits) followed by R/W = ‘1’ on the SDA bus serially (synchronized by SCL falling transitions).

If the SPD5 Hub detects no START condition but if the Host to the SPD5 Hub device bus (HSDA and HSCL) has been inactive (no edges seen) for t_{AVAL} period, then the SPD5 Hub device may assert HSDA low by $t_{\text{IBI_ISSUE}}$ time to request an interrupt. When the SPD5 Hub device requests an interrupt, the Host toggles the HSCL. The SPD5 Hub device transmits its 7-bit binary address; ‘1010’ followed by 3 HID bits and then sets the R/W bit = ‘1’.

When the SPD5 Hub device requests an interrupt, the host may take one of the two actions below.

- The Host sends ACK on 9th bit to accept the interrupt request. At this point, if the SPD5 Hub confirms that it has won the arbitration, the SPD5 Hub device transmits the IBI payload as shown in Table 44 and Table 45 for PEC disabled and PEC enabled configuration respectively. See Figure 12. Figure 12 just shows only first two data bits of the MDB byte to illustrate the timing. The interrupt payload contains MDB followed by 8-bit register contents of Table 133, “MR51” and Table 134, “MR52” in order. The host then issues the STOP command. Note the timing waveform in Figure 12. The host then accepts the IBI payload if it sends an ACK on 9th bit to accept the interrupt request. The host can interrupt the IBI payload at T. If host stops the IBI payload at T bit in the middle of payload, the SPD5 Hub device retains the IBI status flag Table 130, “MR48” [7] = ‘1’ and Pending Interrupt Bits [3:0] internally and waits for the next opportunity to request an interrupt. If the SPD5 Hub device successfully transmits the entire IBI payload, it then clears IBI status flag Table 130, “MR48” [7] = ‘0’ and Pending Interrupt Bits [3:0] = ‘0000’ on its own and does not request for an IBI again unless there is another different event occurs; for another same event, the device does not request for an IBI.
- The Host sends NACK on the 9th bit as shown in Figure 13 followed by a STOP command. In this case, the SPD5 Hub device does not transmit the IBI payload and waits for the next opportunity to request an interrupt. At this point, though Host sent an NACK, it does have a knowledge of which SPD5 Hub device sent the IBI request. The SPD5 Hub retains the IBI status flag Table 130, “MR48” [7] = ‘1’ and Pending Interrupt Bits [3:0] = ‘0001’.

Table 44 — Hub IBI Payload Packet; PEC is Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--------|---|-------|-------|-------|-------|-------|-------|-------|------------------|------|
| S | 1010 | | | | HID | | | R=1 | A ¹ | |
| | MDB = 0x00 | | | | | | | | T=1 | |
| | MR51 [7:0] | | | | | | | | T=1 | |
| | MR52 [7:0] | | | | | | | | T=0 ² | |
| NOTE 1 | See Figure 8 to see how the transition occurs from Host Open Drain (ACK) to Target Push Pull Operation (1st bit of MDB, bit [7]). | | | | | | | | | |
| NOTE 2 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |

2.6.12.2 Mechanics of Interrupt Generation - SPD5 Hub Device (cont'd)

Table 45 — Hub IBI Payload Packet; PEC is Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--------|---|-------|-------|-------|-------|-------|-------|-------|------------------|------|
| S | 1010 | | | | HID | | | R=1 | A ¹ | |
| | MDB = 0x00 | | | | | | | | T=1 | |
| | MR51 [7:0] | | | | | | | | T=1 | |
| | MR52 [7:0] | | | | | | | | T=1 | |
| | PEC | | | | | | | | T=0 ² | P |
| NOTE 1 | See Figure 8 to see how the transition occurs from Host Open Drain (ACK) to Target Push Pull Operation (1st bit of MDB, bit [7]). | | | | | | | | | |
| NOTE 2 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |

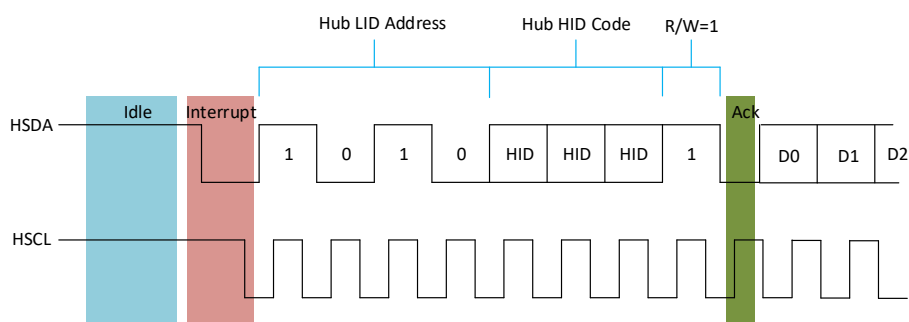


Figure 12 — SPD5 Hub Interrupt; Host Ack Followed by SPD5 Hub Device IBI Payload

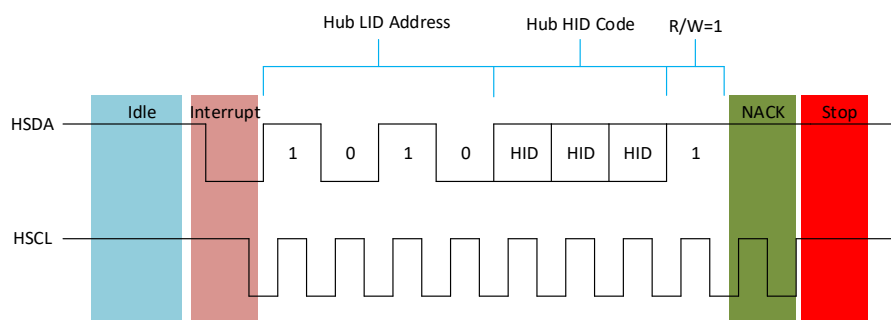


Figure 13 — SPD5 Hub Interrupt; Host NACK Followed by STOP

2.6.12.3 Mechanics of Interrupt Generation - Local Target Device

Event interrupts may be generated by the local device if IBI is enabled. When there is a pending interrupt in any local device and if IBI is enabled, the local device requests an interrupt after detecting START condition by transmitting its 7-bit binary address (LID bits followed by HID bits) followed by R/W = '1' on the SDA bus serially (synchronized by SCL falling transitions).

2.6.12.3 Mechanics of Interrupt Generation - Local Target Device (cont'd)

If the local device detects no START condition but if the Host to the local Target device (through the SPD5 Hub device) bus (LSDA and LSCL) has been inactive (no edges seen) for t_{AVAL} period, then any local device may assert LSDA low by $t_{\text{IBI_ISSUE}}$ time to request an interrupt. When the local device requests an interrupt, the SPD5 Hub device propagates the LSDA to HSDA to Host. The Host toggles the HSCL. The local device transmits its 7-bit binary address (LID bits followed by HID bits) followed by R/W bit = '1' to the SPD5 Hub device. This is shown in Figure 14 top waveform.

The SPD5 Hub device forwards LID bits it receives from the local device to the Host. The SPD5 Hub device substitutes its own HID code in place of the HID bits it receives from the local device and sends to the Host if Host has not issued SETHID CCC previously. This is shown in Figure 14 middle waveform. The SPD5 Hub device forwards HID bits it receives from the local device to the host if Host has issued SETHID CCC previously. This is shown in Figure 14 bottom waveform. The SPD5 Hub device forwards the R/W bit = '1' to the Host.

When the local device requests an interrupt, the host may take one of the two actions below.

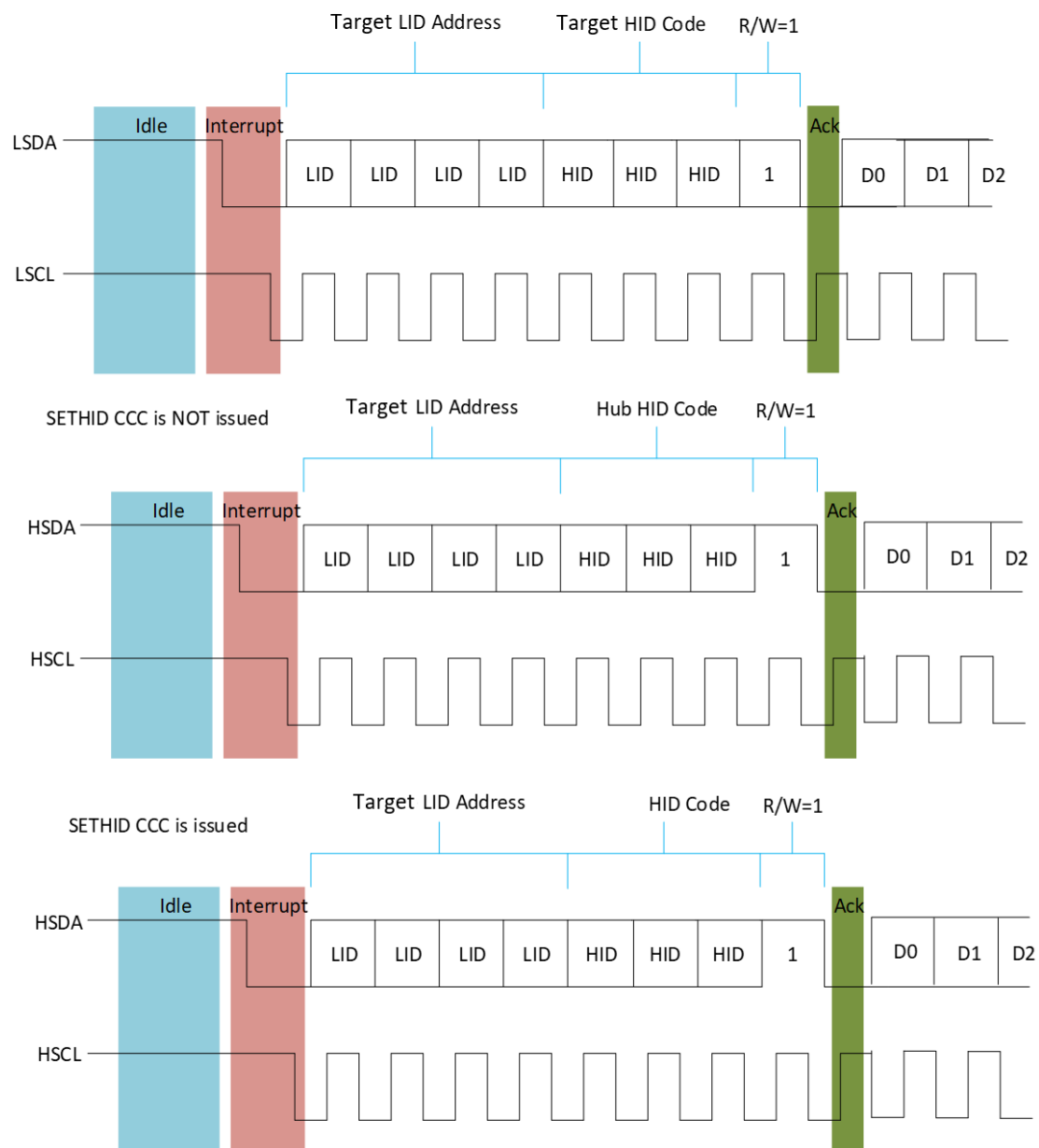
- The Host sends ACK on 9th bit to accept the interrupt request. At this point, if the local device confirms that it has won the arbitration, the local device transmits the IBI payload as shown in Table 46 and Table 47 for PEC disabled and PEC enabled configuration, respectively. See Figure 14. Figure 14 just shows only first two data bits of MDB byte to illustrate the timing. The interrupt payload contains MDB followed by appropriate Target device error register contents in order. The host then issues the STOP command. Note the timing waveform in Figure 14. The host then accepts the IBI payload if it sends an ACK on 9th bit to accept the interrupt request. The host can interrupt the IBI payload at T bit. If host stops the IBI payload at T bit in the middle of payload, the local device retains the IBI status flag and Pending Interrupt Bits [3:0] internally and waits for the next opportunity to request an interrupt. If the local device successfully transmits the entire IBI payload, it then clears IBI status flag and Pending Interrupt Bits [3:0] = '0000' on its own and does not request for an IBI again unless there is another different event occurs; for another same event, the device does not request for an IBI.
- The Host sends NACK on the 9th bit as shown in Figure 15 followed by a STOP command. In this case, the local device does not transmit the IBI payload and waits for the next opportunity to request an interrupt. At this point, though Host sent an NACK, it does have a knowledge of which local device sent the IBI request. The local device retains the IBI status flag and Pending Interrupt Bits [3:0] = '0001'

Table 46 — Target Device IBI Payload Packet; PEC is Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/T | Stop |
|--------|---|-------|-------|-------|-------|-------|-------|-------|------------------|------|
| S | LID | | | | HID | | | R=1 | A ¹ | |
| | MDB = 0x00 | | | | | | | | T=1 | |
| | First Error Code Byte | | | | | | | | T=1 | |
| | Second Error Code Byte | | | | | | | | T=1 | |
| | ... | | | | | | | | T=0 ² | |
| NOTE 1 | See Figure 8 to see how the transition occurs from Host Open Drain (ACK) to Target Push Pull Operation (1st bit of MDB, bit [7]). | | | | | | | | | |
| NOTE 2 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |

2.6.12.3 Mechanics of Interrupt Generation - Local Target Device (cont'd)**Table 47 — Target Device IBI Payload Packet; PEC is Enabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/T | Stop |
|--------|---|-------|-------|-------|-------|-------|-------|-------|------------------|------|
| S | LID | | | | HID | | | R=1 | A ¹ | |
| | MDB = 0x00 | | | | | | | | T=1 | |
| | First Error Code Byte | | | | | | | | T=1 | |
| | Second Error Code Byte | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | PEC | | | | | | | | T=0 ² | |
| NOTE 1 | See Figure 8 to see how the transition occurs from Host Open Drain (ACK) to Target Push Pull Operation (1st bit of MDB, bit [7]). | | | | | | | | | |
| NOTE 2 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |

2.6.12.3 Mechanics of Interrupt Generation - Local Target Device (cont'd)**Figure 14 — Local Device Interrupt, Host Ack Followed by Target Device IBI Payload**

2.6.12.3 Mechanics of Interrupt Generation - Local Target Device (cont'd)

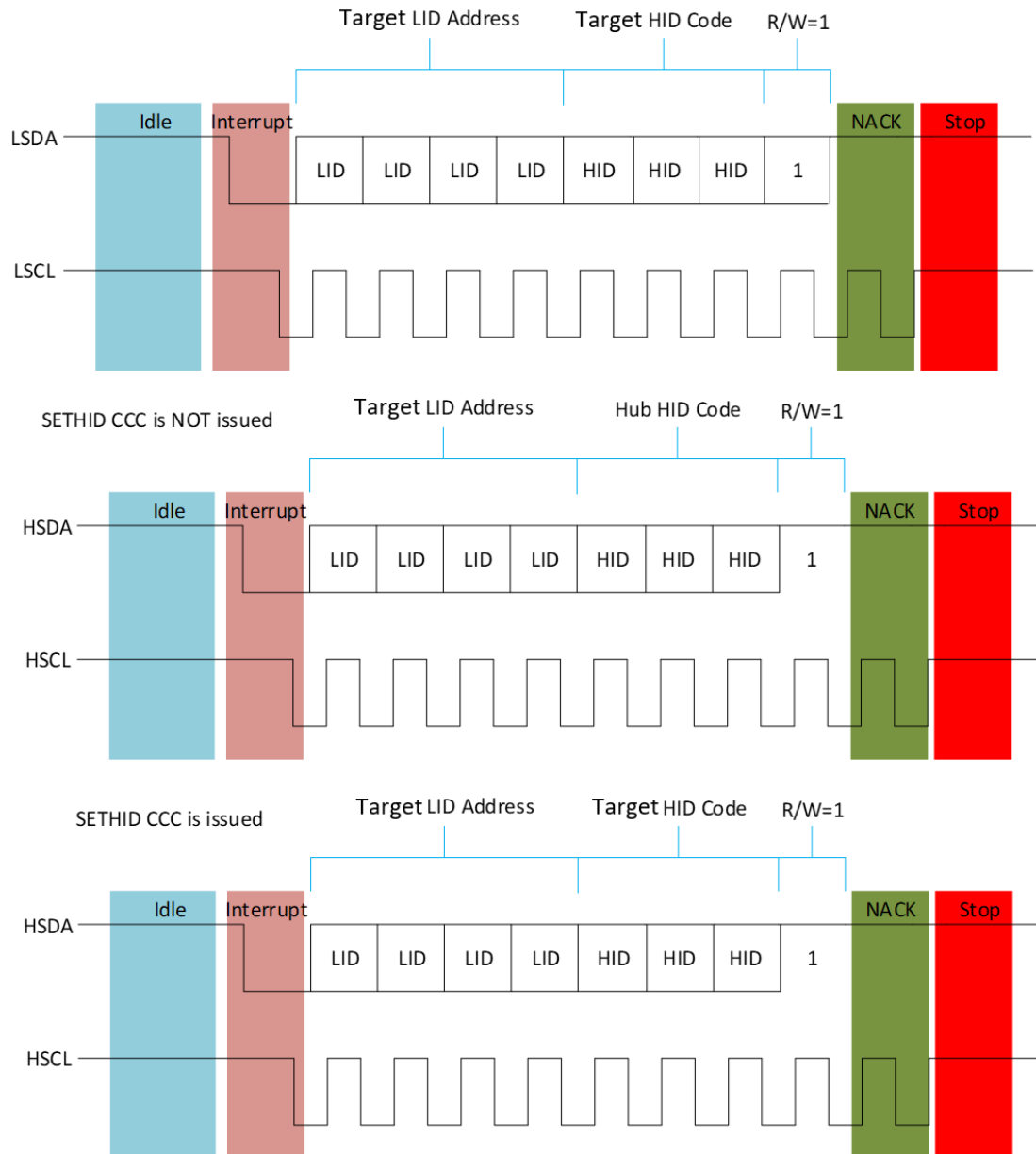


Figure 15 — Local Device Interrupt; Host NACK Followed by STOP

2.6.12.4 Interrupt Arbitration; SETHIDD CCC is Not Issued by Host

As there are multiple local Target devices behind the SPD5 Hub and there are multiple SPD5 Hub devices on I3C Basic bus, multiple device may request an interrupt when the I3C Basic bus is inactive for t_{AVAL} period. Arbitration process is required.

2.6.12.4.1 Interrupt Arbitration - Among SPD5 Hub Devices

This clause explains the interrupt arbitration among the SPD5 Hub devices only. There are up to 8 SPD5 Hub devices on the I3C Basic bus.

As all SPD5 Hub devices have the same 4-bit LID code of '1010', the arbitration is always won by the lower HID code. For example, if one SPD5 Hub device has HID code of '000' and other SPD5 Hub device has a HID code of '011', through the arbitration process, the HID code of '000' wins. The SPD5 Hub device with a HID code of '011' must release the bus and wait for next opportunity to request an interrupt. Table 48 shows the arbitration priority based on the HID code for the SPD5 Hub device.

Table 48 — Interrupt Arbitration - Among SPD5 Hub Devices

| Hub Device LID Code | Hub Device HID Code | Arbitration Priority |
|---------------------|---------------------|----------------------|
| 1010 | 000 | 1 |
| 1010 | 001 | 2 |
| 1010 | 010 | 3 |
| 1010 | 011 | 4 |
| 1010 | 100 | 5 |
| 1010 | 101 | 6 |
| 1010 | 110 | 7 |
| 1010 | 111 | 8 |

2.6.12.4.2 Interrupt Arbitration - Among Local Target Devices Behind One SPD5 Hub Device

This clause explains the interrupt arbitration among the local devices only behind the SPD5 Hub. There are up to 13 local devices behind the SPD5 Hub.

As all local Target devices behind the SPD5 Hub devices have the same 3-bit HID code, Hence the arbitration is always won by the 4-bit LID code. For example, if one local Target device has LID code of '1001' and other local Target device has a LID code of '0010', through the arbitration process, the LID code of '0010' wins. The local device with a LID code of '1001' must release the bus and wait for next opportunity to request an interrupt. Table 49 shows the arbitration priority based on the LID code for the local devices. The Green color cells in Table 49 are the likely devices that will be on a standard DDR5 RDIMM or DDR5 LRDIMM. The Olive color cells in Table 49 do not apply.

During the interrupt arbitration phase, the SPD5 Hub device forwards the winning 4-bit LID code one digit at a time that it receives from the local Target devices to the Host.

The SPD5 Hub device discards the 3-bit '111' HID code received from the local Target device. Instead the SPD5 Hub device forwards its own unique 3-bit HID code one digit at a time to the host. As a result, the Host can identify the winning device based on the 4-bit LID code (Local Target device) and 3-bit HID code (the DIMM). Also, when the SPD5 Hub device substitutes its own unique 3-bit HID code to the host, its own receiver will see that at the input and it compares each 3 bits one at a time as if the host sent those 3 bits. If there is a match, the SPD5 Hub device forwards '1' to the local device interface and if there is a mis-match, the SPD5 Hub device forward '0' to the local device interface. Because the local Target device sends '111' as its 3 bit HID code, the local device knows that it won the arbitration.

2.6.12.4.2 Interrupt Arbitration - Among Local Target Devices Behind One SPD5 Hub Device (cont'd)

Table 49 — Interrupt Arbitration - Among Local Target Devices

| Device | Target Device LID Code | Target Device HID Code | Arbitration Priority |
|---------|------------------------|------------------------|----------------------|
| N/A | 0000 | N/A | N/A |
| RFU | 0001 | 111 | 1 |
| TS0 | 0010 | 111 | 2 |
| RFU | 0011 | 111 | 3 |
| RFU | 0100 | 111 | 4 |
| RFU | 0101 | 111 | 5 |
| TS1 | 0110 | 111 | 6 |
| RFU | 0111 | 111 | 7 |
| PMIC1 | 1000 | 111 | 8 |
| PMIC0 | 1001 | 111 | 9 |
| SPD Hub | 1010 | HID | N/A |
| RCD | 1011 | 111 | 10 |
| PMIC2 | 1100 | 111 | 11 |
| RFU | 1101 | 111 | 12 |
| RFU | 1110 | 111 | 13 |
| N/A | 1111 | N/A | N/A |

2.6.12.4.3 Interrupt Arbitration - Between SPD5 Hub Device and Local Target Devices Behind the Hub

This clause explains the interrupt arbitration among the SPD5 Hub device and all local devices behind that SPD5 Hub. There are up to 13 local devices behind each SPD5 Hub.

As the SPD5 Hub device LID code of '1010', any local Target devices with a lower LID code always wins the arbitration process as shown in Table 49. The local Target devices with a higher LID code than '1010', the SPD5 Hub device wins the arbitration process. As an example, for a typical DDR5 RDIMM/LRDIMM, the RCD always has the lowest priority for winning the interrupt arbitration.

2.6.12.4.4 Interrupt Arbitration - Among Local Target Devices Behind Different SPD5 Hub Devices

This clause explains the interrupt arbitration among the SPD5 Hub devices and all local devices behind any SPD5 Hub devices. There are up to 8 SPD5 Hub devices on the I3C bus and up to 13 local devices behind each SPD5 Hub.

The arbitration process is hybrid of clause 2.6.12.4.1 to clause 2.6.12.4.3. The device with the lowest 4 bit LID code across all local Target devices and across all SPD5 Hub devices always wins the arbitration process. The HID code for that lowest LID code represents the SPD5 Hub device code.

Table 50 shows four examples. In each example, the Target LID code column represent local Target device behind hub which has HID code value of '111'; the Hub HID code column represents the SPD5 Hub device which has LID code of '1010' followed by its own unique HID code; the winning device column represent the final winner among all devices (either SPD Hub device or local Target devices) during the arbitration phase.

2.6.12.4.4 Interrupt Arbitration - Among Local Target Devices Behind Different SPD5 Hub Devices (cont'd)

Example 1: There are total 5 devices (4 local Target devices and 1 Hub device) that are requesting an interrupt. These 5 devices are shown in the RED color text. The devices that are in the Black color text are not requesting an interrupt. The winning device is the local device LID code of '0010' and HID code of '111'. This is because the LID code of '0010' is the lowest among three other local Target device code and its HID code is '111'.

Example 2: There are total 4 devices (2 local Target devices and 2 Hub device) that are requesting an interrupt. These 4 devices are shown in the RED color text. The devices that are in the Black color text are not requesting an interrupt. The winning device is the local device LID code of '1001' and HID code of '100'. This is because the LID code of '1001' is the lowest among one other local Target device code and its HID code is '100'.

Example 3: There are total of 3 devices (2 local Target devices and 1 Hub device) that are requesting an interrupt. These 3 devices are shown in the RED color text. The devices that are in Black color text are not requesting an interrupt. The winning device is the Hub device on DIMM0 with LID code of '1010' and HID code of '000'. This is because LID code '1010' is lower than two other local Target device code and its HID code is '000'.

Example 4: There are total of 2 devices (2 local Target devices) that are requesting an interrupt. These 2 devices are shown in RED color text. The devices that are in Black color text are not requesting an interrupt. This example is unique as two identical Target devices across two different DIMM device is requesting an interrupt. The winning device is the local Target device on DIMM 0 with LID code of '0010' and DIMM 0 HID code of '000'. This is because DIMM0 HID code '000' is lower than DIMM2 HID code of '010'. See clause 2.6.12.4.2.

2.6.12.4.4 Interrupt Arbitration - Among Local Target Devices Behind Different SPD5 Hub Devices (cont'd)

Table 50 — Interrupt Arbitration - Among Local Target and SPD5 Hub Devices

| Example 1 | | | Example 2 | | | Example 3 | | | Example 4 | | |
|------------------------------|--------------|----------------|------------------------------|--------------|----------------|------------------------------|--------------|----------------|------------------------------|--------------|----------------|
| Target LID Code | Hub HID Code | Winning Device | Target LID Code | Hub HID Code | Winning Device | Target LID Code | Hub HID Code | Winning Device | Target LID Code | Hub HID Code | Winning Device |
| 0010 0110 1001 1011 | 000 | | 0010 0110 1001 1011 | 000 | | 0010 0110 1001 1011 | 000 | 1010 000 | 0010 0110 1001 1011 | 000 | 0010 000 |
| 0010 0110 1001 1011 | 001 | | 0010 0110 1001 1011 | 001 | | 0010 0110 1001 1011 | 001 | | 0010 0110 1001 1011 | 001 | |
| 0010 0110 1001 1011 | 010 | | 0010 0110 1001 1011 | 010 | | 0010 0110 1001 1011 | 010 | | 0010 0110 1001 1011 | 010 | |
| 0010 0110 1001 1011 | 011 | | 0010 0110 1001 1011 | 011 | | 0010 0110 1001 1011 | 011 | | 0010 0110 1001 1011 | 011 | |
| 0010 0110 1001 1011 | 100 | | 0010 0110 1001 1011 | 100 | 1001 100 | 0010 0110 1001 1011 | 100 | | 0010 0110 1001 1011 | 100 | |
| 0010 0110 1001 1011 | 101 | | 0010 0110 1001 1011 | 101 | | 0010 0110 1001 1011 | 101 | | 0010 0110 1001 1011 | 101 | |
| 0010 0110 1001 1011 | 110 | | 0010 0110 1001 1011 | 110 | | 0010 0110 1001 1011 | 110 | | 0010 0110 1001 1011 | 110 | |
| 0010 0110 1001 1011 | 111 | 0010 111 | 0010 0110 1001 1011 | 111 | | 0010 0110 1001 1011 | 111 | | 0010 0110 1001 1011 | 111 | |

2.6.12.4.5 Interrupt Arbitration - Between Host and Any SPD5 Hub or Any Local Target Devices Behind Hub

When the bus is idle for t_{AVAL} time, any SPD5 Hub device or any local device behind the Hub can request an interrupt by pulling the SDA bus low.

2.6.12.4.5 Interrupt Arbitration - Between Host and Any SPD5 Hub or Any Local Target Devices Behind Hub (cont'd)

In an uncommon but possible scenario would be that at the exact same time as when the SPD5 Hub or local Target devices are requesting an interrupt, the host is starting an operation to the Hub or local Target devices. When this happens, Host also gets involved in the arbitration process along with the Hub or the local Target devices. During the arbitration phase, there will be always only one winning device and it could be either Hub or the local Target device or the Host.

If the host wins during the arbitration phase, it continues with normal operation. The losing Hub or local Target device waits for next opportunity to send an interrupt.

If the host loses during the arbitration phase, it host must let go of the bus. When Host loses during the arbitration, the host must let the Hub or local Target device finish sending their 4-bit LID code followed by 3-bit HID code followed by R/W = '1'. At this point, during the 9th bit, the host has two options to take the action as noted below:

- Host sends an ACK to accept the interrupt and hence accepts the IBI payload from the winning SPD5 Hub or local Target device. After the IBI payload, the host issues STOP operation.
- Host sends an NACK followed by STOP operation.

A rare but still possible scenario would be that at the exact same time as when the Hub or local Target device is requesting an interrupt, the host is starting an operation to that same exact hub or local Target devices. When this happens, neither Host or nor the hub or local Target device knows it is a winner until the 8th bit and Host always wins. This is because, the hub or local Target device sends R=1 (8th bit) during the interrupt. The host sets W=0 (8th bit) during the operation. As a result, the host wins and the hub or Target device must let go of the bus and wait for the next opportunity to send an interrupt. This is shown as example 3 in Table 51.

Table 51 shows three examples. In each example, the Host is targeting an operation to the device cod. The Target LID code column represent local Target device behind hub which has HID code value of '111'; the Hub HID code column represents the SPD5 Hub device which has LID code of '1010' followed by its own unique HID code; the winning device column represent the final winner among all devices (either Host or SPD5 Hub device or local Target devices) during the arbitration phase.

In example 1, there are total of 5 devices (1 Hub device and 4 local Target devices) are requesting an interrupt at exactly same time as when the Host is starting an operation to hub device on DIMM 3 (1010 011). The winning device is local Target device '0010 111' because it has the lower 4-bit LID code.

In example 2, there are total of 4 devices (2 Hub devices 2 local Target devices) are requesting an interrupt at exactly same time as when the Host is requesting an operation to the local Target device on DIMM5 (0110 101). The host is the winner because it's intended target device has the lower 4-bit LID code than devices that are requesting an interrupt.

In example 3, there are is one 1 hub device on DIMM 2 is requesting an interrupt at exactly the same time as when Host is requesting an operation to the same exact Hub device on DIMM2. In this case, the host is the winner because the 8-bit will be driven low by the Host (W=0) while the Hub device drives it high (R=1) during the interrupt.

An extremely rare but still possible scenario would be that at the exact same time as when the Hub or local Target device is requesting an interrupt, the host is requesting a read operation with default read address pointer mode to the same exact hub or local Target device. When this happens, there is no winning device. This is the only time there is no winning device. This is because, the hub or local Target device sends R=1 (8th bit) during the interrupt and Host also sends R=1 for read request with default read address pointer mode. As a result, there is no winner because all devices, i.e., the Host or the hub or the local Target device, is waiting for other device to ACK. In this case, no device will ACK. Since there is no ACK by any device, the Host must time out and repeats the read request with Repeat Start. When it repeats the read request with Repeat Start, the hub or local Target device does not send an interrupt because of Repeat Start.

2.6.12.4.5 Interrupt Arbitration - Between Host and Any SPD5 Hub or Any Local Target Devices Behind Hub (cont'd)

Table 51 — Interrupt Arbitration - Best Host and Local Target and SPD5 Hub Devices

| Example 1 | | | | Example 2 | | | | Example 3 | | | |
|--------------------|------------------------------|--------------|----------------|--------------------|------------------------------|--------------|----------------------------|--------------------|------------------------------|--------------|----------------------------|
| Host Target Device | Target LID Code | Hub HID Code | Winning Device | Host Target Device | Target LID Code | Hub HID Code | Winning Device | Host Target Device | Target LID Code | Hub HID Code | Winning Device |
| 1010 011 | 0010 0110 1001 1011 | 000 | | 0110 101 | 0010 0110 1001 1011 | 000 | | 1010 010 | 0010 0110 1001 1011 | 000 | |
| | 0010 0110 1001 1011 | 001 | | | 0010 0110 1001 1011 | 001 | | | 0010 0110 1001 1011 | 001 | |
| | 0010 0110 1001 1011 | 010 | | | 0010 0110 1001 1011 | 010 | | | 0010 0110 1001 1011 | 010 | Host operation to 1010 010 |
| | 0010 0110 1001 1011 | 011 | | | 0010 0110 1001 1011 | 011 | | | 0010 0110 1001 1011 | 011 | |
| | 0010 0110 1001 1011 | 100 | | | 0010 0110 1001 1011 | 100 | | | 0010 0110 1001 1011 | 100 | |
| | 0010 0110 1001 1011 | 101 | | | 0010 0110 1001 1011 | 101 | Host operation to 0110 101 | | 0010 0110 1001 1011 | 101 | |
| | 0010 0110 1001 1011 | 110 | | | 0010 0110 1001 1011 | 110 | | | 0010 0110 1001 1011 | 110 | |
| | 0010 0110 1001 1011 | 111 | 0010 111 | | 0010 0110 1001 1011 | 111 | | | 0010 0110 1001 1011 | 111 | |

2.6.12.5 Interrupt Arbitration; SETHID CCC is Issued by Host

The interrupt arbitration process works similar way as defined in clause 2.6.12.4 but with a simplification. It is assumed that in this case, the all Target devices behind the Hub across all DIMMs on I3C Basic bus has a unique 7-bit device select code. Further, it assumes that all devices on a DIMM has the same 3-bit HID code.

The SPD5 Hub device forwards the 4-bit LID code and 3-bit HID code one digit at a time that it receives from the local Target devices to the Host. The SPD5 Hub device's own receiver will see the same input and it forwards it back to the local device interface.

2.6.12.5.1 Interrupt Arbitration - Between SPD5 Hub Device and Local Target Devices Behind Hub

On any given DIMM, the arbitration is always won by the device that has the lowest 4-bit LID code since all devices on the DIMM has same 3-bit HID code.

2.6.12.5.2 Interrupt Arbitration - Between All SPD5 Hub Devices and All Local Target Devices Behind Hub

Across multiple DIMMs, the arbitration is always won by the device that has the lowest 7-bit address (4-bit LID + 3-bit HID).

2.6.12.5.3 Interrupt Arbitration - Between Host and All Devices

An uncommon but possible scenario would be that at the exact same time as when the Target device is requesting an interrupt, the host is starting an operation to the Target device. When this happens, Host also gets involved in the arbitration process along with the Target devices. During the arbitration phase, there will be always only one winning device and it could be either Host or the Target device.

If the host wins during the arbitration phase, it continues with normal operation. The losing Target device waits for next opportunity to send an interrupt.

If the host loses during the arbitration phase, it host must let go of the bus. When Host loses during the arbitration, the host must let the Target device finish sending their 4-bit LID code followed by 3-bit HID code followed by R/W = '1'. At this point, during the 9th bit, the host has two options to take the action as noted below:

- Host sends an ACK to accept the interrupt and hence accepts the IBI payload from the winning Target device. After the IBI payload, the host issues STOP operation.
- Host sends an NACK followed by STOP operation.

A rare but still possible scenario would be that at the exact same time as when the SPD5 Hub device is requesting an interrupt, the host is starting an operation to that same SPD5 Hub device. When this happens, neither Host or nor the SPD5 Hub device knows it is a winner until the 8th bit and Host always wins. This is because, the SPD5 hub device sends R=1 (8th bit) during the interrupt. The host sets W=0 (8th bit) during the operation. As a result, the host wins and the SPD5 Hub device must let go of the bus and wait for the next opportunity to send an interrupt.

An extremely rare but still possible scenario would be that at the exact same time as when the SPD5 Hub device is requesting an interrupt, the host is requesting a read operation with default read address pointer mode to the SPD5 Hub device. When this happens, there is no winning device. This is the only time there is no winning device. This is because, the SPD5 Hub device sends R=1 (8th bit) during the interrupt and Host also sends R=1 for read request with default read address pointer mode. As a result, there is no winner because all devices, i.e., the Host or the SPD5 Hub device, is waiting for other device to ACK. In this case, no device will ACK. Since there is no ACK by any device, the Host must time out and repeats the read request with Repeat Start. When it repeats the read request with Repeat Start, the SPD5 Hub device does not send an interrupt because of Repeat Start.

2.6.12.6 Clearing Device Status and IBI Status Registers

The SPD5 Hub device provides the IBI status in Table 130, "MR48" [7] by setting it to '1'. The SPD5 Hub device clears the IBI status register Table 130, "MR48" [7] to '0' automatically when it sends a complete IBI (including payload and without interruption) and it also clears Pending Interrupt Bits [3:0] to '0000'. Once IBI status register is cleared, the SPD5 Hub does not request for an IBI again unless an another event occurs.

The SPD5 Hub device provides the device status in Table 133, "MR51" and Table 134, "MR52" registers. The status information in Table 133, "MR51" and Table 134, "MR52" are latched and remains set even after the SPD5 Hub device sends IBI payload and clears the IBI status register Table 130, "MR48" [7] to '0'. The host must explicitly clear the status register through Clear command by writing '1' for appropriate status or by issuing a Global clear command.

2.6.12.6 Clearing Device Status and IBI Status Registers (cont'd)

After Host issues a clear command, if the condition is no longer present, the SPD5 Hub device clears the appropriate status register, clears the IBI status register to '0' and Pending Interrupt Bits [3:0] to '0000' even if the SPD5 Hub device has not sent the IBI. After Host issues clear command, if the condition is still present, the device will again set the appropriate status register, sets the IBI status register to '1' and Pending Interrupt Bits [3:0] to '0001' even if the device has already sent the IBI and entire IBI payload.

2.6.13 Packet Error Check (PEC) Function

In I²C mode, packet error checking is not supported. Only I3C Basic mode supports packet error checking.

The SPD5 Hub device implements an 8-bit Packet Error Code (PEC) which is appended at the end of all transactions if PECs is enabled through DEVCTRL CCC or by directly writing '1' to Table 18, "MR18" [7]. The PEC is a CRC-8 value calculated on all the messages bytes except for START, REPEATED START, STOP conditions or T-bits, ACK and NACK and IBI header (7'h7E followed by W=0) bits.

The polynomial for CRC-8 calculations is:

$$\bullet C(X) = X^8 + X^2 + X^1 + 1$$

The seed value for PEC function is all zero.

When Host calculates PEC for SPD5 Hub device, it includes LID and HID bits followed by R/W bit.

2.6.14 Parity Error Check Function

In I²C mode, parity error checking is not supported except for supported CCCs. Only I3C Basic mode supports parity error checking.

By default, when SPD5 Hub device is put in I3C Basic mode, parity function is automatically enabled. The host can disable the function after it is enabled. Host can also disable the parity function with DEVCTRL CCC or by directly writing '1' to Table 117, "MR18" [6]. When parity function is disabled, the SPD5 Hub device simply ignores the "T" bit information from the Host. The host may actually choose to compute the parity and send that information during "T" bit or simply drive static low or high in "T" bit.

The SPD5 Hub device implements ODD parity. If an odd number of bits in the byte are '1', the parity bit value is '0'. If even number of bits in the byte are '1', the parity bit value is '1'. The host computes the parity and sends during "T" bit.

2.6.15 Packet Error Check and Parity Error Handling

There are two types of error checking done by the SPD5 Hub device - Parity error checking and Packet Error checking. By default, the parity error checking is always enabled and packet error checking is disabled when the SPD5 Hub device is put in I3C Basic mode. The host may enable the packet error checking at any time. The parity error is calculated for each byte. The host sends parity error information in "T" bit.

I3C basic defines TE0, TE1, TE2, TE3, TE4, TE5, and TE6 error detection for Target devices. Only TE1 and TE2 error detection is supported by the SPD5 Hub for parity checking. All other errors are not supported and not applicable.

2.6.15.1 Write Command Data Packet Error Handling - PEC Disabled

The SPD5 Hub device checks for the parity error for each byte in a packet except for the device select code byte that it receives from the host as shown in Table 52.

2.6.15.1 Write Command Data Packet Error Handling - PEC Disabled (cont'd)**Table 52 — Write Command Data Packet; PEC Disabled**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | 0 | 0 | 0 | 0 | Blk Addr [4:1] ⁴ | | | | T | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The SPD5 Hub device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 5 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Write command - if no parity error:

- The SPD5 Hub device executes the command.

Write command - if parity error:

- The SPD5 Hub device discards the byte in the packet that had a parity error.
- The SPD5 Hub device discards all subsequent bytes in that packet until the STOP operation. The SPD5 Hub device may or may not check parity for all sub-sequent bytes in that packet.
- Note that as the packet contains more than one byte, if first byte had no parity error but the second byte had a parity error, the SPD5 Hub device may or may not execute the first byte operation but second byte and all subsequent bytes operations are discarded.
- The SPD5 Hub device sets Table 134, “MR52” [0], Table 130, “MR48” [7], and P_Err in GETSTATUS CCC to ‘1’; updates Pending Interrupt Bits [3:0] to ‘0001’; and waits for the next opportunity to send an in band interrupt if IBI is enabled.

2.6.15.2 Read Command Data Packet Error Handling - PEC Disabled

The SPD5 Hub device checks for parity error for each byte in a packet except for the device select code byte that it receives from the host prior to Repeat Start as shown in Table 53.

The SPD5 Hub device does not compute the parity when it sends the data to the Host. The Host does not check for parity error for the bytes that SPD5 Hub device sends. The SPD5 Hub device sends Continuous (‘1’) or Stop (‘0’) information during “T” bit.

2.6.15.2 Read Command Data Packet Error Handling - PEC Disabled (cont'd)

Table 53 — Read Command Data Packet; PEC Disabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | 0 | 0 | 0 | 0 | Blk Addr [4:1] ⁴ | | | T | | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A/N ^{5,6} | |
| | Data | | | | | | | | T=1 | |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 ^{7,8} | Sr ⁹ or P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The SPD5 Hub device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 5 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity errors, the device may eventually ACK. | | | | | | | | | |
| NOTE 6 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 7 | See Figure 10 to see how Host ends Target device operation. | | | | | | | | | |
| NOTE 8 | For NVM memory access (i.e., MemReg = '1'), when last byte is reached within 16 byte boundary or for volatile memory access (i.e., MemReg = '0'), when last byte (i.e., MR 255) is reached (extreme rare case), the Target device sends T = '0'. See Figure 11 to see how Host ends Target device operation. | | | | | | | | | |
| NOTE 9 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Read Command - If no parity error:

- The SPD5 Hub sends ACK back to the host when Host perform Start Repeat operation.
- The SPD5 Hub device executes the command and sends the data as shown in Table 53.

Read Command - If parity error:

- The SPD5 Hub device discards the byte in the packet that had a parity error.
- The SPD5 Hub device discards second byte in that packet if the parity error occurred in first byte. The SPD5 Hub device may or may not check the parity for second byte in that packet.
- The SPD5 Hub sends NACK back to the host when Host performs a Start Repeat operation. This is shown in the **RED color** cell in Table 53 above. The NACK represents either a parity error in one of the two bytes or that SPD5 Hub is not able to start the read operation. The host may re-try Repeat Start again. The host may do the Repeat Start as many times as it may desire. If the SPD5 Hub device NACKs due to parity error in a previous byte from the host, it will always NACK regardless of how many times host tries Repeat Start.
- The SPD5 Hub does not send the data shown in Table 53 and instead expects Host to perform STOP operation.
- The SPD5 Hub device sets Table 134, "MR52" [0], Table 130, "MR48" [7], and P_Err in GETSTATUS CCC to '1'; updates Pending Interrupt Bits [3:0] in GETSTATUS to '0001'; and waits for the next opportunity to send an in band interrupt if IBI is enabled.

2.6.15.3 Write Command Data Packet Error Handling - PEC Enabled

The SPD5 Hub device checks for the parity error for each byte in a packet except for the device select code byte that it receives from the host as shown in Table 54. Further, the SPD5 Hub device checks for the packet error for the entire packet (from Start condition until last byte of Data) that it receives from the host as shown in Table 54.

Table 54 — Write Command Data Packet; PEC Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | CMD | | | W=0 | Blk Addr [4:1] ⁴ | | | | T | |
| | Data | | | | | | | | T | |
| | ... | | | | | | | | T | |
| | Data | | | | | | | | T | |
| | PEC | | | | | | | | T | |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The SPD5 Hub device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 5 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Write command - if no parity error:

- The SPD5 Hub device waits for the entire packet. If no error in packet, the SPD5 Hub device executes the command. If there is an error in the packet, the SPD5 Hub device discards the entire packet and does not execute that packet and waits for STOP, sets the Table 134, “MR52” [1] and Table 130, “MR48” [7] to ‘1’ and PEC_Err in GETSTATUS CCC to ‘1’ and updates Pending Interrupt Bits [3:0] in GETSTATUS CCC to ‘0001’ and waits for the next opportunity to send in band interrupt if IBI is enabled.

Write command - if parity error:

- The SPD5 Hub device discards that byte and the entire packet until STOP operation.
- The SPD5 Hub device sets Table 134, “MR52” [0], Table 130, “MR48” [7], and P_Err in GETSTATUS CCC to ‘1’; updates Pending Interrupt Bits [3:0] in GETSTATUS CCC to ‘0001’ and waits for the next opportunity to send in band interrupt if IBI is enabled.
- The SPD5 Hub device may or may not check the error for the packet. If the SPD5 Hub device checks for the packet error, likely it will detect an error in the packet and the device may also set Table 134, “MR52” [1] and PEC_Err in GETSTATUS CCC to ‘1’ as well.

2.6.15.4 Read Command Data Packet Error Handling - PEC Enabled

The SPD5 Hub device checks for the parity error for each byte in a packet except for the device select code byte that it receives from the host prior to Repeat Start as shown in Table 55.

The SPD5 Hub device does not compute the parity when it sends the data to the Host. The does not check for parity error for the bytes shown in Table 55. The SPD5 Hub device sends Continuous (‘1’) or Stop (‘0’) information during “T” bit when SPD5 Hub device is sending the read data.

2.6.15.4 Read Command Data Packet Error Handling - PEC Enabled (cont’d)

The SPD5 Hub device checks for the PEC error for a packet that it receives from the Host from Start condition to Repeat Start condition (from first device select code followed by the address offset and CMD byte).

The SPD5 Hub device computes the packet error code for the entire packet starting with Repeat Start (device select code and the data SPD5 Hub device transmits back to Host)

Table 55 — Read Command Data Packet; PEC Enabled

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|--------------|---------------|-------|-----------------------------|-------|-------|-------|--------------------|----------------------|
| S or Sr | 1 | 0 | 1 | 0 | HID | | | W=0 | A ^{1,2,3} | |
| | MemReg | Blk Addr [0] | Address [5:0] | | | | | | T | |
| | CMD | | | R=1 | Blk Addr [4:1] ⁴ | | | T | | |
| | PEC | | | | | | | | T | |
| Sr | 1 | 0 | 1 | 0 | HID | | | R=1 | A/N ^{5,6} | |
| | Data | | | | | | | | T=1 | Sr ⁸ or P |
| | ... | | | | | | | | T=1 | |
| | Data | | | | | | | | T=1 | |
| | PEC | | | | | | | | T=0 ⁷ | |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (MemReg bit). | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | The SPD5 Hub device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | The memory size of SPD5 Hub device is limited to 1024 Bytes. SPD5 Hub device ignores Blk Addr [4] bit. | | | | | | | | | |
| NOTE 5 | If Target device NACKs during Repeat Start for any reason, the host may re-try Repeat Start again. The host can do the Repeat Start as many times it may desire. If Target device NACKs due to PEC error or parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity or PEC errors, the device may eventually ACK. The PEC calculation by the Target device only includes device select code of the ACK response of the Repeat start operation. In other words, if there are more than one Repeat Start operation, the Target device includes device select of only the last Repeat Start from the Host when it ACKs in PEC calculation and all other NACK responses of the device select code of the Repeat Start are not included in PEC calculation. | | | | | | | | | |
| NOTE 6 | See Figure 9 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK). | | | | | | | | | |
| NOTE 7 | See Figure 11 to see how Target device ends the operation followed by Host STOP operation. | | | | | | | | | |
| NOTE 8 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Read command - If no parity error and no PEC error:

- The SPD5 Hub sends ACK back to the host when Host perform a Start Repeat operation.
- The SPD5 Hub device executes the command and sends the data as shown in Table 55.
- The SPD5 Hub computes PEC for the bytes (from Start condition to PEC byte prior to Repeat Start) shown in Table 55.

Read command - if parity error or PEC error:

- The SPD5 Hub device discards the byte in the packet that had a parity error.
- The SPD5 Hub device discards second byte in that packet if a parity error occurred in first byte. The SPD5 Hub device may or may not check parity for the second byte in that packet.
- The SPD5 Hub device discards the packet if there is a PEC error.

2.6.15.4 Read Command Data Packet Error Handling - PEC Enabled (cont'd)

- The SPD5 Hub sends NACK back to the host when Host perform Start Repeat operation. This is shown in the **RED color** cell in Table 55. The NACK represents either PEC error or a parity error in one of the three bytes or that SPD5 Hub is not able to start the read operation. The host may re-try Repeat Start again. The host may do the Repeat Start as many times it may desire. The PEC calculation by SPD5 Hub device only includes device select code of the ACK responses of the Repeat Start operation. In other words, if there are more than one Repeat Start operation, the SPD5 Hub device includes the device select of only the last Repeat Start from the Host when it ACKs in PEC calculation and other NACK responses of the device select codes of the Repeat Start are not included in PEC calculation. If the SPD5 Hub device NACKs due to PEC error or a parity error in a previous bytes from Host, it will always NACK regardless of how many times Host tries Repeat Start.
- The SPD5 Hub does not send any data shown in Table 55 and instead expects Host to perform STOP operation.
- The SPD5 Hub device sets Table 134, “MR52” [0], Table 130, “MR48” [7], and P_Err in GETSTATUS CCC to ‘1’ for parity error and Table 134, “MR52” [1] and Table 130, “MR48” [7], and PEC_Err in GETSTATUS CCC to ‘1’ for PEC error. Further, the SPD5 Hub updates Pending Interrupt Bits [3:0] in GETSTATUS CCC to ‘0001’ and waits for the next opportunity to send an in band interrupt if IBI is enabled.

2.6.16 CCC Packet Error Handling

Parity error and PEC error detected in a CCC packet are handled the same way as described for normal Read/Write operations.

2.6.17 Error Reporting

All error conditions including PEC error check and Parity error check detected by the SPD5 Hub devices are captured in Table 133, “MR51” and Table 134, “MR52” registers.

There are three different possible ways error information can be communicated to the host.

1. The Host makes the read request to Table 133, “MR51” and Table 134, “MR52” registers.
2. The Host starts any transaction with 7'h7E IBI header. (Only applicable in I3C Mode)
3. The SPD5 Hub device sends in band interrupt if enabled, when its SCL and SDA input has been idle for t_{AVAIL} time. (Only applicable in I3C Basic Mode).

2.6.18 I3C Basic Common Command Codes (CCC)

The I3C Basic Standard lists large number of Common Command Codes (CCC). Not all CCC are required to be supported. The SPD5 Hub device NACKs for all unsupported CCC. The SPD5 Hub supports CCC as listed in Table 56.

The SPD5 Hub device requires STOP operation in between when switching from CCC operation to private device specific Write or Read or Default Read Address Pointer mode operation and vice versa. In other words, any CCC operation must be followed by STOP operation before continuing to any device specific Write or Read or Default Read Address Pointer mode operation. Similarly, any device specific Write or Read or Default Read Address Pointer mode operation must be followed by STOP operation before continuing to any CCC operation. The SPD5 Hub device also requires STOP operation between any direct CCC to broadcast CCC.

The SPD5 Hub device does allow Repeat Start operation between any direct CCC to any other direct CCC or between any broadcast CCC to any other broadcast CCC or between any private Write or Read or Default Read Address Pointer mode operation to any other private Write or Read or Default Read Address Pointer mode operation.

2.6.18 I3C Basic Common Command Codes (CCC) (cont'd)

CCC is an I3C concept by definition, and shall always conform to I3C SDR timings, irrespective of whether the device has switched from I2C mode or not.

Prior to dynamic address assignment (SETAASA/P), the Target Device(s) may drive the ACK/NACK past the Open Drain SCL rising but before the next SCL falling transition, as a longer overlap in Open Drain is harmless.

Immediately after the Open Drain ACK (upon the next SCL falling edge), the bus should transition to push pull mode (though still at 1Mhz) as described in the MIPI I3C Basic Standard V1.0, clause 5.1.2.3 “Handoff from Address ACK to SDR Controller Write Data” and Figure 32 “I3C Data Transfer – ACK by Target”.

For additional details on how to handle an the ACK transition, please refer to the MIPI I3C Basic Standard V1.0, clause 5.1.2.3.1 “Transition from Address ACK to SDR Controller Write Data”

Table 56 — SPD5 Hub CCC Support Requirement

| CCC | Mode | Code | Description | Note |
|----------------------------|-----------|------|---|------|
| ENEC | Broadcast | 0x00 | Enable Event Interrupts | |
| | Direct | 0x80 | | |
| DISEC | Broadcast | 0x01 | Disable Event Interrupts | |
| | Direct | 0x81 | | |
| RSTDAA | Broadcast | 0x06 | Put the device in I ² C Mode (aka: Reset Dynamic Address Assignment) | |
| SETAASA | Broadcast | 0x29 | Put the device in I3C Basic Mode (aka: Set All Addresses to Static Address) | |
| GETSTATUS | Direct | 0x90 | Get Device Status | |
| DEVCAP | Direct | 0xE0 | Get Device Capability | 1 |
| SETHID | Broadcast | 0x61 | SPD5 Hub updates 3-bit HID field, updates “T” bit with updated parity calculation for all devices behind Hub and stops 3-bit HID translation. | 1 |
| DEVCTRL | Broadcast | 0x62 | Configure SPD5 Hub and all devices behind Hub | 1 |
| NOTE 1 JEDEC specific CCC. | | | | |

2.6.18.1 ENEC CCC

The ENEC CCC is only supported after device is put in I3C Basic mode. In I²C mode, it is illegal for host to issue this CCC. When ENEC CCC is registered by the SPD5 Hub, it updates Table 121, “MR27” [4] = ‘1’ and it takes in effect at the next Start operation (i.e., after STOP condition). Table 57 to Table 60 shows an example of a single ENEC CCC. Table 61 shows the encoding definition for ENEC CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7’h7E with W=0 byte in PEC calculation.

2.6.18.1 ENEC CCC (cont'd)**Table 57 — ENEC CCC - Broadcast**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x00 (Broadcast) | | | | | | | | T | |
| | 0x00 | | | | | | | ENINT | T | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 58 — ENEC CCC - Broadcast with PEC

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x00 (Broadcast) | | | | | | | | T | |
| | 0x00 | | | | | | | ENINT | T | |
| | PEC | | | | | | | | T | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 59 — ENEC CCC - Direct

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|------------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x80 (Direct) | | | | | | | | T | |
| Sr | DevID[6:0] | | | | | | | W=0 | A ^{1,2} | |
| | 0x00 | | | | | | | ENINT | T | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | The SPD5 Hub device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 3 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.18.1 ENEC CCC (cont'd)**Table 60 — ENEC CCC - Direct with PEC**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|------------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x80 (Direct) | | | | | | | | T | |
| | PEC | | | | | | | | T | |
| Sr | DevID[6:0] | | | | | | | W=0 | A ^{1,2} | |
| | 0x00 | | | | | | | ENINT | T | |
| | PEC | | | | | | | | T | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | The SPD5 Hub device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 3 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 61 — ENEC CCC - Byte Encoding

| Bit | Encoding | Notes |
|-------|---|---|
| ENINT | 0 = No Action 1 = Enable IBI Interrupt | It is illegal for Host to issue ENEC CCC with ENINT bit = '0' |

2.6.18.2 DISEC CCC

The DISEC CCC is only supported after device is put in I3C Basic mode. In I²C mode, it is illegal for host to issue this CCC. When DISEC CCC is registered by the SPD5 Hub, it updates Table 121, “MR27” [4] = '0' and it takes in effect at the next Start operation (i.e., after STOP condition). Table 62 to Table 65 shows an example of a single DISEC CCC. Table 66 shows the encoding definition for DISEC CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

Table 62 — DISEC CCC - Broadcast

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|--------|----------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x01 (Broadcast) | | | | | | | | T | |
| | 7'h00 | | | | | | | DISINT | T | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.18.2 DISEC CCC (cont'd)**Table 63 — DISEC CCC - Broadcast with PEC**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|--------|----------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x01 (Broadcast) | | | | | | | | T | |
| | 7'h00 | | | | | | | DISINT | T | |
| | PEC | | | | | | | | T | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 64 — DISEC CCC - Direct

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|--------|------------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x81 (Direct) | | | | | | | | T | |
| Sr | DevID[6:0] | | | | | | | W=0 | A ^{1,2} | |
| | 0x00 | | | | | | | DISINT | T | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | The SPD5 Hub device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 3 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 65 — DISEC CCC - Direct with PEC

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|--------|------------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x81 (Direct) | | | | | | | | T | |
| | PEC | | | | | | | | T | |
| Sr | DevID[6:0] | | | | | | | W=0 | A ^{1,2} | |
| | 0x00 | | | | | | | DISINT | T | |
| | PEC | | | | | | | | T | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | The SPD5 Hub device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The SPD5 Hub device ignores the entire packet until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 3 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.18.2 DISEC CCC (cont'd)**Table 66 — DISEC CCC Byte Encoding**

| Bit | Encoding | Notes |
|--------|--|---|
| DISINT | 0 = No Action 1 = Disable IBI Interrupt | It is illegal for Host to issue DISEC CCC with DISINT bit = '0' |

2.6.18.3 RSTDAA CCC

The RSTDAA CCC is only supported after device is put in I3C Basic mode. In I²C mode, this CCC is ignored. When RSTDAA CCC is registered by the SPD5 Hub, it updates Table 117, “MR18” [5] = ‘0’ and it takes in effect at the next Start operation (i.e., after STOP condition). Further it disables IBI and PEC function (Table 121, “MR27” [4] = ‘0’, Table 117, “MR18” [7] = ‘0’ respectively) and clears parity function Table 117, “MR18” [6] = ‘0’).

Table 67 to Table 68 show an example of a single RSTDAA CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

Table 67 — RSTDAA CCC - Broadcast

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|----------------|----------------------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x06 (Broadcast) | | | | | | | | T | Sr ² or P |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 68 — RSTDAA CCC - Broadcast with PEC

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x06 (Broadcast) | | | | | | | | T | |
| | PEC | | | | | | | | T | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.18.4 SETAASA CCC

The SETAASA CCC is only supported when device is in I²C mode; however, it still follows I3C SDR timings compliant to CCC definitions. In I²C mode, when issues this CCC, to guarantee that this CCC is registered by the device without any error, the host shall limit the maximum speed operation for this CCC to 1 MHz. In I3C Basic mode, this CCC is ignored. When SETAASA CCC is registered by the SPD5 Hub, it updates Table 117, “MR18” [5] = ‘1’ and it takes in effect at the next Start operation (i.e., after STOP condition). Table 69 shows an example of a single SETAASA CCC.

SETAASA CCC does not support PEC function as device is in I²C mode and there is no PEC function in I²C mode.

Table 69 — SETAASA CCC - Broadcast

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|------------------|-------|-------|-------|-------|-------|-------|-------|-------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A | |
| | 0x29 (Broadcast) | | | | | | | | T | P |

2.6.18.5 GETSTATUS CCC

The GETSTATUS CCC is supported in I3C Basic mode. In I²C mode, this CCC is ignored (i.e., it is not executed internally and GETSTATUS CCC code is not acknowledged and host must do STOP operation). Table 70 to Table 71 show an example of a single GETSTATUS CCC

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

Table 70 — GETSTATUS CCC - Direct

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------------------|-------|-------|-------|----------------|----------------------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x90 (Direct) | | | | | | | | T | |
| Sr | DevID[6:0] | | | | | | | R=1 | A ¹ | |
| | PEC_Err | 0 | 0 | 0 | 0 | 0 | 0 | 0 | T | |
| | 0 | 0 | P_Err | 0 | Pending Interrupt | | | | T | Sr ² or P |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.18.5 GETSTATUS CCC (cont'd)

Table 71 — GETSTATUS CCC - Direct with PEC¹

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|-------|-------------------|-------|-------|-------|----------------|----------------------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ² | |
| | 0x90 (Direct) | | | | | | | | T | |
| | PEC | | | | | | | | T | |
| Sr | DevID[6:0] | | | | | | | R=1 | A ¹ | |
| | PEC_Err | 0 | 0 | 0 | 0 | 0 | 0 | 0 | T | |
| | 0 | 0 | P_Err | 0 | Pending Interrupt | | | | T | |
| | PEC | | | | | | | | T | Sr ³ or P |
| NOTE 1 | GETSTATUS CCC with PEC check is only supported in I3C Basic mode. | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKS if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 72 — GETSTATUS CCC Byte Encoding

| Bit | Encoding | Notes |
|-------------------|---|---|
| PEC_Err | 0 = No Error 1 = PEC Error occurred | This register is cleared when Host issues clear command to Table 119, “MR20” [1] for PEC error. |
| P_Err | 0 = No Error 1 = Protocol Error; Parity Error occurred | This register is cleared when Host issues clear command to Table 119, “MR20” [0] Parity error. |
| Pending Interrupt | 0000 = No Pending Interrupt 0001 = Pending Interrupt All other encodings are reserved | This register is cleared when Host issues clear command to any appropriate device status register that causes IBI status register to get cleared. |

When the SPD5 Hub device responds to GETSTATUS CCC, after it completes the response, the PEC_Err, P_Err, and Pending Interrupt Bits [3:0] do not automatically get cleared. The host must explicitly clear the appropriate status register through Clear command b writing ‘1’ to corresponding register or by issuing Global Clear command. Once the SPD5 Hub device clears the appropriate status register, only then PEC_Err, P_err, and Pending Interrupt Bits [3:0] gets cleared.

After host issues clear command, if the condition is still present, the device will again set the appropriate status register, sets the IBI status register to ‘1’ and Pending Interrupt Bits [3:0] to ‘0001’.

2.6.18.6 DEVCAP CCC

The DEVCAP CCC is only supported after device is put in I3C Basic mode. In I²C mode, it is illegal for host to issue this CCC. Table 73 to Table 74 show an example of a single DEVCAP CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

2.6.18.6 DEVCAP CCC (cont'd)**Table 73 — DEVCAP CCC - Direct**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0xE0 (Direct) | | | | | | | | T | |
| Sr | DevID[6:0] | | | | | | | R=1 | A ¹ | |
| | MSB (Each bit defines capability) | | | | | | | | T | |
| | LSB (Each bit defines capability) | | | | | | | | T | |
| | Sr ² or P | | | | | | | | | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 74 — DEVCAP CCC - Direct with PEC

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|----------------------|---|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0xE0 (Direct) | | | | | | | | T | |
| | PEC | | | | | | | | T | |
| Sr | DevID[6:0] | | | | | | | R=1 | A ¹ | |
| | MSB (Each bit defines capability) | | | | | | | | T | |
| | LSB (Each bit defines capability) | | | | | | | | T | |
| | PEC | | | | | | | | T | |
| Sr ² or P | | | | | | | | | | |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 75 — DEVCAP CCC Byte Encoding

| Bit | Encoding | Notes |
|----------|--|----------------------------|
| MSB [7] | RFU | Coded as '0' |
| MSB[6] | RFU | Coded as '0' |
| MSB[5] | RFU | Coded as '0' |
| MSB[4] | RFU | Coded as '0' |
| MSB[3] | RFU | Coded as '0' |
| MSB[2] | 0 = No Support for Timer based Reset 1 = Supports Timer based Reset | SPD5 Hub hard codes to '1' |
| MSB[1:0] | RFU | Coded as '000' |
| LSB[7:0] | RFU | Coded as '000' |

2.6.18.7 SETHID CCC

The SETHID CCC is supported only when device is in I²C mode. In I²C mode, when issues this CCC, to guarantee that this CCC is registered by the device without any error, the host shall limit the maximum speed operation for this CCC to 1 MHz. In I3C Basic mode, it is illegal for host to issue this CCC. When SETHID CCC is registered by the SPD5 Hub, it stops 3-bit HID translation. Table 76 shows an example of a single SETHID CCC. The host must send all '0' in the data byte followed by "T" bit. The SPD5 Hub device forwards bits [7:4, 0] to local devices behind the SPD5 Hub as it receives from the Host. The SPD5 Hub device substitutes its own 3-bit HID code in bits [3:1] and forwards it to the local devices behind the SPD5 Hub. The SPD5 Hub device also re-calculates the parity information and forwards the updated parity information in "T" bit. As the device is in I²C mode when SETHID CCC is issued, the PEC function is not supported.

The Host may issue SETHID CCC more than one time.

Table 76 — SETHID CCC - Broadcast

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|------------------|-------|-------|-------|-------|-------|-------|-------|-------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A | |
| | 0x61 (Broadcast) | | | | | | | | T | |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | T | P |

2.6.18.8 DEVCTRL CCC

On a typical I3C Basic bus, there can be up to 120 devices. For DDR5 DIMM application environment, there are up to 8 SPD5 Hub devices and behind each SPD5 Hub devices, there are 4 local Target devices totaling up to 40 or more devices on I3C Basic bus. For certain operation such as enable or disable functions that are common to all devices (i.e., Packet Error Check), the host must go through one device at a time which takes significant amount of time at initial power up. Further, it requires additional complexity on the host because it must speak different protocol depending on how it may access the device until all devices are configured identically.

To help expedite this configuration operation and to simplify the host complexity, the device supports the DEVCTRL CCC. The DEVCTRL CCC is supported either in I²C mode or I3C Basic mode of operation. In I²C mode, when issues this CCC, to guarantee that this CCC is registered by the device without any error, the host shall limit the maximum speed operation for this CCC to 1 MHz. Table 77 to Table 78 show an example of a single DEVCTRL CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

The host shall pay attention to DEVCTRL CCC. The DEVCTRL CCC is limited to SPD5 Hub device's volatile register space only and should not be used for SPD5 Hub's NVM operation. If DEVCTRL CCC is used to access device specific registers (e.g. RegMod = '1'), the host shall still follow any device specific register restriction. For example, if device specific register require STOP operation for device to take in the effect of the setting, the host must also use STOP operation when using DEVCTRL CCC to access device specific register.

In I²C mode, DEVCTRL CCC must be limited to 1 byte addressing mode for SPD5 Hub device (i.e., Table 112, "MR11" [3] = '0').

2.6.18.8 DEVCTRL CCC (cont'd)**Table 77 — DEVCTRL CCC - Broadcast**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|--|-------|-------|------------------|-------|-------------|-------|--------|----------------|----------------------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x62 (Broadcast) | | | | | | | | T | |
| | AddrMask[2:0] | | | StartOffset[1:0] | | PEC BL[1:0] | | RegMod | T | |
| | DevID[6:0] | | | | | | | 0 | T ² | |
| | Byte 0 Data Payload | | | | | | | | T | |
| | Byte 1 Data Payload | | | | | | | | T | |
| | Byte 2 Data Payload | | | | | | | | T | |
| | Byte 3 Data Payload | | | | | | | | T | Sr ³ or P |
| NOTE 1 | The SPD5 Hub NACKs if there is a parity error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 2 | An exception is made for DEVCTRL CCC where device does report a parity error when it determines 7-bit device select code issued by the host does not match with its own device code. If 7-bit device select code does not match but if parity is still valid, the device does not check for parity error in subsequent bytes; ignores the entire packet and waits until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 3 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

Table 78 — DEVCTRL CCC - Broadcast with PEC¹

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|---------|---|-------|-------|------------------|-------|-------------|-------|--------|----------------------|------|
| S or Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ² | |
| | 0x62 (Broadcast) | | | | | | | | T | |
| | AddrMask[2:0] | | | StartOffset[1:0] | | PEC BL[1:0] | | RegMod | T | |
| | DevID[6:0] | | | | | | | 0 | T ³ | |
| | Byte 0 Data Payload | | | | | | | | T | |
| | Byte 1 Data Payload | | | | | | | | T | |
| | Byte 2 Data Payload | | | | | | | | T | |
| | Byte 3 Data Payload | | | | | | | | T | |
| PEC | | | | | | | | T | Sr ⁴ or P | |
| NOTE 1 | DEVCTRL CCC with PEC check is only supported in I3C Basic mode. | | | | | | | | | |
| NOTE 2 | The SPD5 Hub NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start. | | | | | | | | | |
| NOTE 3 | An exception is made for DEVCTRL CCC where device does report a parity error when it determines 7-bit device select code issued by the host does not match with its own device code. The device does not check for PEC as all subsequent bytes are discarded due to parity error. If 7-bit device select code does not match but if parity is still valid, the device does not check for parity error in subsequent bytes; ignores the entire packet and waits until STOP or next Repeat Start operation. | | | | | | | | | |
| NOTE 4 | Repeat Start or Repeat Start with 7'h7E. | | | | | | | | | |

2.6.18.8 DEVCTRL CCC (cont'd)**Table 79 — DEVCTRL CCC Command Definition**

| Parameter | Definition |
|------------------|--|
| AddrMask[2:0] | <p>Broadcast, Unicast or Multicast Command Selection</p> <p>000 = Unicast Command; SPD5 Hub device responds if DevID[6:0] field matches with SPD5 Hub device's own 7-bit address (4-bit LID + 3-bit HID)</p> <p>011 = Multicast Command; SPD5 Hub device and possible other device responds if DevID[6:3] field matches with SPD5 Hub device's own 4-bit LID address</p> <p>111 = Broadcast Command; All devices responds to this command</p> <p>All other encodings are reserved</p> |
| StartOffset[1:0] | <p>Only applicable if RegMod = '0'</p> <p>Identifies the starting Byte (Byte 0 or Byte 1 or Byte 2 or Byte 3) for DEVCTRL CCC. Host can start at any Byte (from Byte 0 to Byte 3) and has continuous access to next byte until STOP operation. If Byte 3 is reached, the host is responsible for applying STOP operation.</p> <p>00 = Byte 0 01 = Byte 1 10 = Byte 2 11 = Byte 3</p> |
| PEC BL[1:0] | <p>Only applicable if RegMod = '0' and PEC function is enabled.</p> <p>Identifies the burst length just for this DEVCTRL CCC. The device uses the setting in this field to know when the PEC byte is expected after the data bytes.</p> <p>00 = 1 Byte 01 = 2 Byte 10 = 3 Byte 11 = 4 Byte</p> |
| RegMod | <p>Identifies if DEVCTRL is going to be used for General Registers as identified in Byte 0 to Byte 3 or device specific address offset register.</p> <p>0 = Access to General Registers in Byte 0 to Byte 3 (i.e., StartOffset[1:0] = Valid) 1 = Device Specific Offset Address (i.e., StartOffset[1:0] and PECBL[1:0] is a don't care and does not apply). The Host shall NOT use RegMod = '1' with Broadcast Command if there are different types of devices on the I3C Basic bus.</p> |
| DevID[6:0] | <p>Identifies 7-bit device address. Device responds to DEVCTRL CCC data packet depending on AddrMask[2:0].</p> <p>If AddrMask[2:0] = '111', DevID[6:0] is a don't care and device always responds. If AddrMask[2:0] = '000', DevID[6:0] must match for device to respond If AddrMask[2:0] = '011', DevID[6:3] must match for device to respond. DevID[2:0] is don't care. For any other codes for AddrMask[2:0], the device always NACKs.</p> |

2.6.18.8 DEVCTRL CCC (cont'd)**Table 80 — DEVCTRL CCC Data Payload Definition**

| Byte # | Bit # | Function | Definition | Comment |
|---|-------|----------------------|---|--|
| Byte 0 | [7] | PEC Enable | 0 = Disable 1 = Enable | Table 117, “MR18” [7] is updated |
| | [6] | Parity Disable | 0 = Enable 1 = Disable | Table 117, “MR18” [6] is updated |
| | [5:2] | RFU | RFU | |
| | [1] | RSVD | 0 = RSVD 1 = RSVD | SPD5 Hub device always ignores this bit. |
| | [0] | RFU | RFU | |
| Byte 1 | [7:4] | RFU | RFU | |
| | [3] | Global and IBI Clear | 0 = No Action 1 = Clear All Event and pending IBI ¹ | Table 121, “MR27” [7] is updated. |
| | [2:0] | RFU | RFU | |
| Byte 2 | [7:0] | RFU | RFU | |
| Byte 3 | [7:0] | RFU | RFU | |
| NOTE 1 After Target device clears the event, the device can still have certain registers set to ‘1’ if the event is still present in which case, the device will generate an IBI again at the next opportunity. | | | | |

2.6.18.8.1 DEVCTRL CCC Examples - RegMod = ‘0’

Table 81 shows an example of DEVCTRL CCC data packet. It assumes that all devices on the bus are already in I3C Basic mode with PEC function disabled and parity function enabled. In this example, the Host uses DEVCTRL CCC as Multicast command. Host sends Multicast command to all devices with 4-bit LID code of ‘1001’ on I3C bus to do VR Enable followed by all devices with 4-bit LID code of ‘0110’ to disable parity function. The host sends AddrMask = ‘011’ to indicate Multicast command with DevID[6:3] match; StartOffset = ‘00’ to indicate starting Byte 0 and RegMod = ‘0’ to indicates general register. Upon receiving this command, all devices with DevID[6:3] that matches to ‘1001’ will do the VR Enable command and DevID[6:3] that matches to ‘0110’ with disable the parity function.

Table 81 — DEVCTRL CCC Example - Multicast Command to ‘1001’ and ‘0110’ Devices

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--|------------------|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x62 (Broadcast) | | | | | | | | T | |
| | 011 | | | 00 | | 00 | | 0 | T | |
| | 1001 000 | | | | | | | 0 | T | |
| | 0000 0010 | | | | | | | | T | |
| Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x62 (Broadcast) | | | | | | | | T | |
| | 011 | | | 00 | | 00 | | 0 | T | |
| | 0110 000 | | | | | | | 0 | T | |
| | 0100 0000 | | | | | | | | T | P |
| NOTE 1 See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation. | | | | | | | | | | |

2.6.18.8.1 DEVCTRL CCC Examples - RegMod = '0' (cont'd)

Table 82 shows an example of DEVCTRL CCC data packet. It assumes that all devices on the bus are already in I3C Basic mode with PEC function disabled and parity function enabled. In this example, the Host uses DEVCTRL CCC as Broadcast command to enable PEC function. The host sends AddrMask = '111' to indicate Broadcast command; StartOffset = '00' to indicate starting Byte 0 and RegMod = '0' to indicate general register. Upon receiving this command, all devices will enable PEC function.

Table 82 — DEVCTRL CCC Example - Broadcast Command to All Devices

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--|------------------|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x62 (Broadcast) | | | | | | | | T | |
| | 111 | | | 00 | | 00 | | 0 | T | |
| | 0000 000 | | | | | | | 0 | T | |
| | 1000 0000 | | | | | | | | T | |
| NOTE 1 See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation. | | | | | | | | | | |

Table 83 shows an example of DEVCTRL CCC data packet. It assumes that all devices on the bus are already in I3C Basic mode with PEC function disabled and parity function enabled. In this example, the Host uses DEVCTRL CCC as Unicast command to enable VR on DIMM5. The host sends AddrMask = '000' to indicate Unicast command; StartOffset = '00' to indicate starting Byte 0 and RegMod = '0' to indicate general register. Upon receiving this command, PMIC on DIMM5 will enable its regulator.

Table 83 — DEVCTRL CCC Example - Unicast Command to PMIC on DIMM5

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--|------------------|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x62 (Broadcast) | | | | | | | | T | |
| | 000 | | | 00 | | 00 | | 0 | T | |
| | 1001 101 | | | | | | | 0 | T | |
| | 0000 0010 | | | | | | | | T | |
| NOTE 1 See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation. | | | | | | | | | | |

2.6.18.8.2 DEVCTRL CCC Examples - RegMod = '1'

Table 84 shows an example of DEVCTRL CCC data packet for the purpose of configuring device specific address offset register. It assumes that all devices on the bus are already in I3C mode with PEC function enabled and parity function enabled. In this example, the Host sends Multicast command to all devices with 4-bit LID code of '0010' on the I3C Basic bus to write to address offset of 0x1C and 0x1D with data 0xFF and 0x55 respectively followed by all devices with 4-bit LID of '1001' on the I3C bus to write to address offset of 0x15 with data 0x78.

The PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

2.6.18.8.2 DEVCTRL CCC Examples - RegMod = '1' (cont'd)**Table 84 — DEVCTRL CCC Example - Multicast Command to '0010' and '1001' Devices**

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--|---|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x62 (Broadcast) | | | | | | | | T | |
| | 011 | | | 00 | | 00 | | 1 | T | |
| | 0010 000 | | | | | | | 0 | T | |
| | 0001 1100 (address offset 0x1C) | | | | | | | | T | |
| | 0010 0000 (CMD field = 2 bytes of data) | | | | | | | | T | |
| | 1111 1111 (data) | | | | | | | | T | |
| | 0101 0101 (data) | | | | | | | | T | |
| | PEC | | | | | | | | T | |
| Sr | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x62 (Broadcast) | | | | | | | | T | |
| | 011 | | | 00 | | 00 | | 1 | T | |
| | 1001 000 | | | | | | | 0 | T | |
| | 0001 0101 (address offset 0x15) | | | | | | | | T | |
| | 0000 0000 (CMD field = 1 byte of data) | | | | | | | | T | |
| | 0111 1000 (data) | | | | | | | | T | |
| | PEC | | | | | | | | T | |
| NOTE 1 See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation. | | | | | | | | | | P |

Table 85 shows an example of DEVCTRL CCC data packet for the purpose of configuring device specific address offset register. It assumes that all devices on the bus are already in I3C Basic mode with PEC function disabled and parity function enabled. In this example, the Host sends Multicast command to all devices with 4-bit LID code of '1001' on the I3C Basic bus to write to address offset of 0x13 with data 0xFF and it continues to write data 0x01 to the next address.

Table 85 — DEVCTRL CCC Example - Multicast Command to '1001' Devices

| Start | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | A/N/T | Stop |
|--------|---|-------|-------|-------|-------|-------|-------|-------|----------------|------|
| S | 1 | 1 | 1 | 1 | 1 | 1 | 0 | W=0 | A ¹ | |
| | 0x62 (Broadcast) | | | | | | | | T | |
| | 011 | | | 00 | | 00 | | 1 | T | |
| | 1001 000 | | | | | | | 0 | T | |
| | 0001 0011 (address offset 0x13) | | | | | | | | T | |
| | 1111 1111 (data) | | | | | | | | T | |
| | 0000 0001 (data) | | | | | | | | T | P |
| NOTE 1 | See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation. | | | | | | | | | |

2.6.19 IO Operation

The SPD5 Hub device supports configurable IO operation scheme of either Open Drain or Push-Pull on its Host side of the interface (HSCL and HSDA) and Local side of the interface (LSCL and LSDA). Further, the SPD5 Hub device supports independent IO configuration on Host side and Local side of the interface. The voltage levels for either Open Drain mode or Push Pull can also be independently configurable.

At power on, by default, the SPD5 Hub device comes up in legacy I²C mode of operation with Open Drain IO for both its Host side and Local side of the interface. The maximum speed is limited to 1 MHz and supported IO voltage levels are from 1.0 V to 3.3 V.

After power on, the host may put the SPD5 Hub device in I3C mode of operation.

In I3C mode, the host may drive the HSCL clock input of the SPD5 device using either Push-Pull output driver or using the open-drain output driver. It is expected that for all DDR5 DIMM family environment, the host may always drive the HSCL clock input using a Push-Pull output driver.

To support in band interrupt, the SPD5 device supports dynamic switching between Open Drain mode and Push Pull mode on its HSDA and LSDA bus for various event. Table 86 describes the different mode of operation by the SPD5 device for each cycle.

Table 86 — SPD5 Hub Device Dynamic IO Operation Mode Switching

| | Open Drain Mode | Push Pull Mode |
|--|-----------------|----------------|
| START + Device Select Code | Yes | No |
| START + 7'h7E IBI Header Byte | Yes | No |
| REPEAT START + Device Select Code | No | Yes |
| REPEAT START + 7'h7E Header Byte | No | Yes |
| CCC Bytes (i.e., after 7'h7E+W=0+ACK) | No | Yes |
| STOP | No | Yes |
| ACK/NACK Responses | Yes | No |
| Command, Block Address, Address Operation | No | Yes |
| Interrupt Request by Target + Device Select Code | Yes | No |
| IBI Payload | No | Yes |
| Write Data, T-bit sequence | No | Yes |
| Read Data, T-bit sequence | No | Yes |
| PEC, T-bit sequence | No | Yes |

2.6.20 Bus Clear

The SPD Hub device supports the following described Bus Clear feature in I²C mode only. Any attempt by host to perform I²C Bus clear on a Target device in I3C mode may result in an active drive bus contention on the SDA data line.

There may be abnormal circumstances when the host abruptly stops clocking SCL while the Target device is in the middle of outputting data for read operation. For these type of events, the SDA data line may appear as stuck low as the device is expecting to receive more clock pulses from the host. Eventually when the host has control of the SCL clock, the host may optionally clear the device that is stuck low on the SDA data line by sending continuous 18 clock pulses without driving the SDA data line followed by STOP operation. The device floats the SDA line within 18 clock pulses and returns to the Idle state. The device is ready for normal new transaction with Start condition.

2.6.21 Bus Reset

To prevent a malfunctioning device from locking up the I²C bus or I3C Basic bus, a bus reset mechanism is defined. It uses a timeout mechanism on HSCL as shown in Figure 16 to force a device bus reset. All devices (All SPD5 Hub and all local Target devices behind the hub) on a I²C or I3C Basic bus reset simultaneously. Bus reset operation works same way regardless of whether device is operating in I²C or I3C mode.

To guarantee the device resets I²C bus or I3C Basic bus, the HSCL clock input Low time has to be greater than or equal to $t_{\text{TIMEOUT(Max)}}$.

The SPD5 Hub device does not reset I²C bus or I3C Basic bus if the HSCL clock input Low time is less than $t_{\text{TIMEOUT(Min)}}$.

If the HSCL clock input Low time is between $t_{\text{TIMEOUT(Min)}}$ and $t_{\text{TIMEOUT(Max)}}$, the SPD5 Hub device does not guarantee and it may or may not reset the I²C bus or I3C Basic bus.

When RESET, the SPD5 Hub device takes the following actions.

1. Interface and any pending command or transactions are cleared
2. All internal register values are preserved unless noted otherwise in item # 3 below.
3. Device returns to I²C mode of operation; Table 117, “MR18” [7:5] resets to ‘000’; Table 121, “MR27” [4] resets to ‘0’; and Table 134, “MR52” [1:0] resets to ‘00’.
4. Device does not re-sample HSA pin.
5. Device floats the HSDA such that it gets pulled High by external/other pullup. The device pulls LSDA pin High.
6. Device treats bus resets as STOP operation.

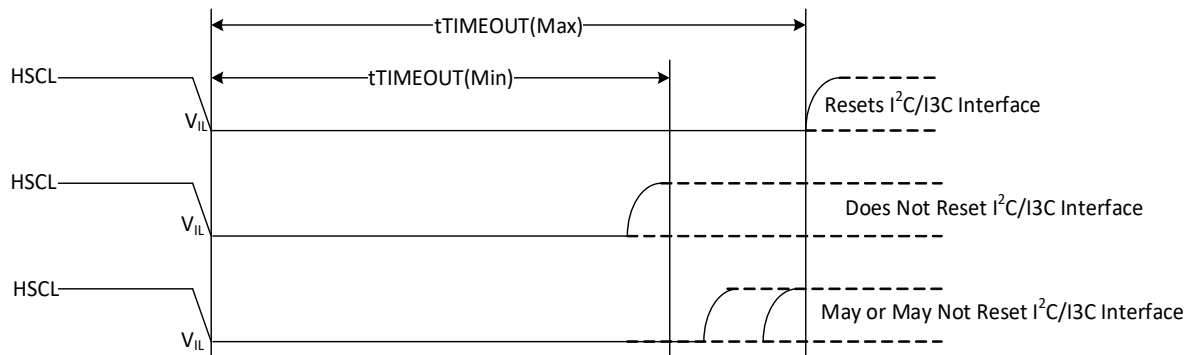


Figure 16 — I²C or I3C Basic Bus Reset - SPD5 Hub Device

2.6.22 Command Truth Table

The command truth table as shown in Table 87 only applies in I3C Basic mode with PEC enabled. In I²C mode and I3C Basic mode with PEC disabled, the command truth table does not apply.

2.6.22 Command Truth Table (cont'd)**Table 87 — For I3C Basic Mode Only with PEC Enabled - Command Truth Table**

| SPD5 Command | Command Name | Command Code | RW | MemReg | Block Address | Address |
|----------------------------|--------------|---------------------|------------------|------------------|--|---------------------|
| | | 2nd Byte Bits [7:5] | 2nd Byte Bit [4] | 1st Byte Bit [7] | 2nd Byte Bits [3:0] 1st Byte Bits [6] | 1st Byte Bits [5:0] |
| Write 1 Byte to Register | W1R | 000 | 0 | 0 | V | V |
| Read 1 Byte from Register | R1R | | 1 | 0 | V | V |
| Write 1 Byte to NVM | W1M | | 0 | 1 | V | V |
| Read 1 Byte from NVM | R1M | | 1 | 1 | V | V |
| Write 2 Byte to Register | W2R | 001 | 0 | 0 | V | V |
| Read 2 Byte from Register | R2R | | 1 | 0 | V | V |
| Write 2 Byte to NVM | W2M | | 0 | 1 | V | V |
| Read 2 Byte from NVM | R2M | | 1 | 1 | V | V |
| Write 4 Byte to Register | W4R | 010 | 0 | 0 | V | V |
| Read 4 Byte from Register | R4R | | 1 | 0 | V | V |
| Write 4 Byte to NVM | W4M | | 0 | 1 | V | V |
| Read 4 Byte from NVM | R4M | | 1 | 1 | V | V |
| Write 16 Byte to Register | W16R | 011 | 0 | 0 | V | V |
| Read 16 Byte from Register | R16R | | 1 | 0 | V | V |
| Write 16 Byte to NVM | W16M | | 0 | 1 | V | V |
| Read 16 Byte from NVM | R16M | | 1 | 1 | V | V |
| Reserved | RSVD | 011 to 111 | RSVD | RSVD | RSVD | RSVD |

2.7 SPD5 Hub Device Pin Definition

Table 88 — Pin Description - SPD5118

| Pin # | Pin Name | Pin Type | Description |
|-------|------------------|----------|---|
| 5 | VDDSPD | Power | 1.8 V Input Power Supply. Connect minimum of 1.0 μ F capacitor to GND |
| 6 | VSS | GND | GND |
| 3 | HSCL | I | Host Bus - I ² C/I ³ C Basic Input Clock |
| 2 | HSDA | IO | Host Bus - I ² C/I ³ C Basic Data |
| 8 | HSA | I | Host Bus - I ² C/I ³ C Basic Address Pin. See Table 89 for ID definition. |
| 4 | LSCL | O | Local Bus - I ² C/I ³ C Basic Output Clock |
| 1 | LSDA | IO | LSDA - Local Bus - I ² C/I ³ C Data |
| 7 | VDDIO | Power | 1.0 V Input Power Supply. Connect minimum of 1.0 μ F capacitor to GND. |
| 9 | Thermal Pad, GND | GND | Connected to GND Plane |

Table 89 shows the HSA pin resistor values and corresponding 3-bit HID for SPD5 Hub device.

Table 89 — HSA Pin Resistor Value and ID - SPD5 Family

| HSA Pin Connection | 3-bit HID | Comment |
|------------------------|-----------|--|
| 10.0 K Ω to GND | HID = 000 | 1% Resistor |
| 15.4 K Ω to GND | HID = 001 | |
| 23.2 K Ω to GND | HID = 010 | |
| 35.7 K Ω to GND | HID = 011 | |
| 54.9 K Ω to GND | HID = 100 | |
| 84.5 K Ω to GND | HID = 101 | |
| 127 K Ω to GND | HID = 110 | |
| 196 K Ω to GND | HID = 111 | |
| Tied directly to GND | HID = 000 | Offline Mode; Write protect override enabled |

2.8 SPD5 Hub Device - Write and Read Access

In both I2C and I3C mode, any read or write access to SPD5 Hub NVM memory must be followed by STOP operation (i.e., not Repeat Start) before launching new access to SPD5 Hub volatile memory registers. Conversely, any read or write access to SPD5 Hub volatile memory registers must be followed by STOP operation (i.e., not Repeat Start) before launching new access to SPD5 Hub NVM memory.

2.8.1 Write and Read Access - NVM Memory

For I²C mode and I3C mode with PEC disabled, the write access to NVM memory is done within 16 byte boundary in a block. Once the last address within 16 byte boundary in a block is reached, the device stops the operation. The write operation to remaining addresses are not executed by the SPD5 Hub device. The SPD5 Hub device does not loop to first address within the 16 byte boundary in that block. The SPD5 Hub device does not set any register to inform this to the host and does not generate any interrupt to the host.

For I3C mode with PEC enabled, if for a given Write access (either W2M, W4M, W16M) to NVM memory reaches the last address within the 16 byte boundary in a block, (i.e., Byte 15, Byte 31, Byte 47, Byte 63), the device stops the operation. The write operation to remaining addresses are not executed by the SPD5 Hub device. The SPD5 Hub device does not loop to first address within the 16 byte boundary in that block. The SPD5 Hub device does not set any register to inform this to the host and does not generate any interrupt to the host.

Unlike Write access to NVM memory, any read access to NVM memory does not impose 16 byte boundary or block boundary. The Read access to NVM memory is treated as continuous address space even if it crosses 16 byte boundary or block boundary in I²C mode or I3C Basic mode (with or without PEC enabled). The last byte for NVM memory is 1024th byte and when the address pointer reaches to the last byte, the SPD5 Hub device does not return any data. In I²C mode and I3C Basic mode with PEC disabled, the host must do the STOP operation.

2.8.2 Write and Read Access - Register Memory

There is no concept of “Block Address”. The “Block Address” is treated as simply as upper address bit when MemReg = ‘0’ by the SPD5 Hub device.

For I²C mode and I3C Basic mode with PEC disabled or PEC enabled, any access to read or write to Register memory is continuous address space even if it appears crossing 16 byte boundary or “Block Address” boundary. The last byte is MR255 and when the address pointer reaches to MR255, the SPD5 Hub device does not return any data. In I²C mode and I3C Basic mode with PEC disabled, the host must do the STOP operation.

2.9 Write Protection of Non-Volatile Memory

2.9.1 Normal Run Time Operation (HSA Pin is Tied to GND via a Resistor Value)

In this mode, the SPD5 Hub device offers a write protection for its NVM memory. Table 113, “MR12” [7:0] and Table 114, “MR13” [7:0] registers contain a bit map for write protection status of each 64 byte block of NVM memory. Table 113, “MR12” [7:0] and Table 114, “MR13” [7:0] registers can be written to ‘1’ at any time. When any bit in Table 113, “MR12” or Table 114, “MR13” is set to ‘1’, further writes to that corresponding block of NVM are ignored and Table 134, “MR52” [6] bit is set to ‘1’.

Once any bit is written to ‘1’ in Table 113, “MR12” and Table 114, “MR13”, clearing that bit in normal run time mode is not allowed. Any attempt to clear the bit in Table 113, “MR12” and Table 114, “MR13” is ignored and Table 134, “MR52” [5] bit is set to ‘1’.

2.9.2 Offline Tester Operation (HSA Pin is Tied Directly to GND, No Resistor Value)

In this mode, the SPD5 Hub device allows to clear any bit in Table 113, “MR12” and Table 114, “MR13” registers. Once cleared, the SPD5 Hub device allows to modify the corresponding block of NVM memory.

2.9.3 Suggested Steps to Program SPD5 Hub Devices

The recommended steps to program the SPD Hub devices are:

1. Connect HSA Pin directly to GND (without a resistor).
2. Power up the device. The device senses HSA pin. It sets Table 130, “MR48” [2] = ‘1’ and enables write protection override.
3. Program Table 113, “MR12” and Table 114, “MR13” to enable desired NVM blocks to be written.
4. Program desired NVM blocks.
5. Program Table 113, “MR12” and Table 114, “MR13” to set the write protection as desired.

2.10 AC Timing Definition

2.10.1 I²C or I3C Basic Bus Timing

The SPD5 Hub device follows the I²C or I3C Basic bus timing requirements. Figure 17 and Figure 18 show the timing diagram for Data bus Input and Data Output parameters.

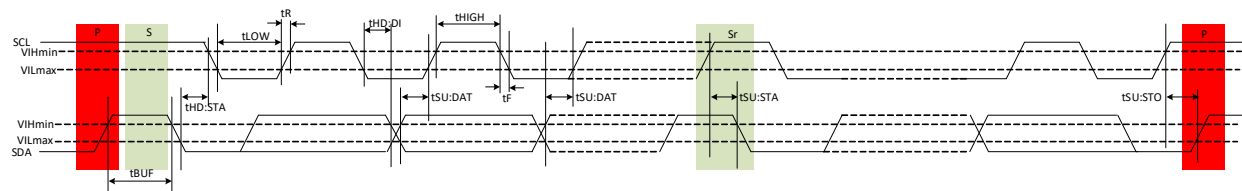


Figure 17 — I²C or I3C Basic Bus AC Input Timing Parameter Definition

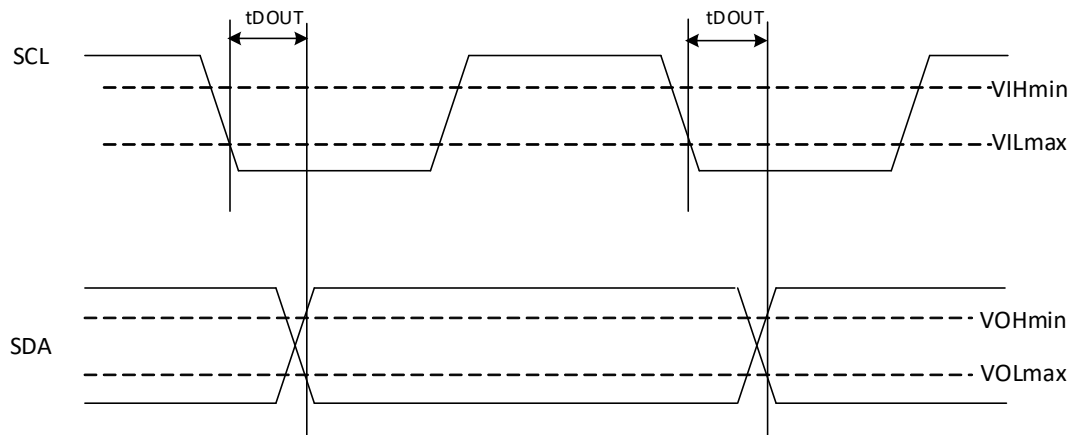


Figure 18 — I3C Basic Bus AC Data Output Timing Parameter Definition

2.10.1 I²C or I³C Basic Bus Timing (cont'd)

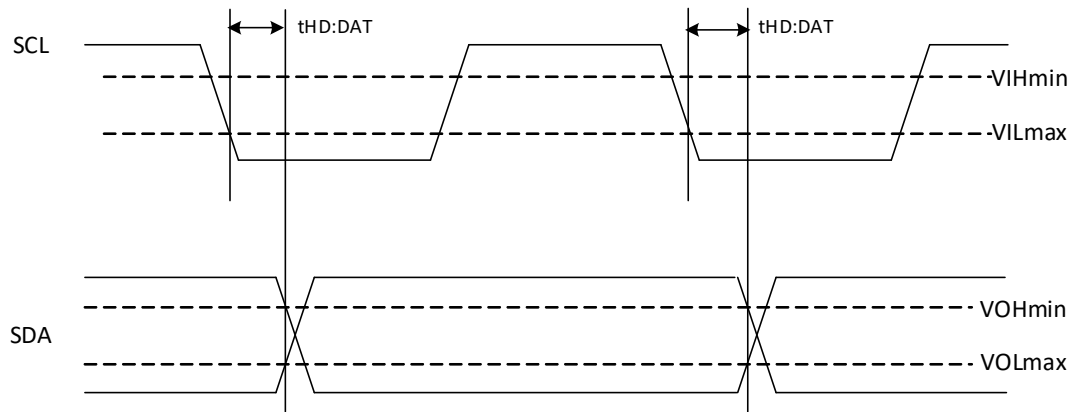


Figure 19 — I²C Bus AC Data Output Timing Parameter Definition

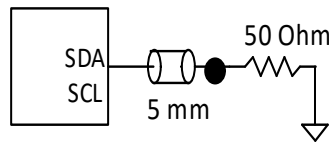


Figure 20 — Output Slew Rate and Output Timing Reference Load

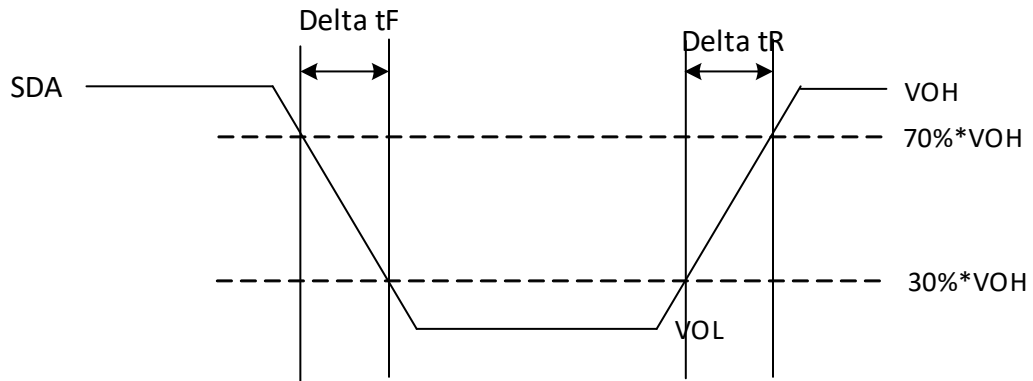
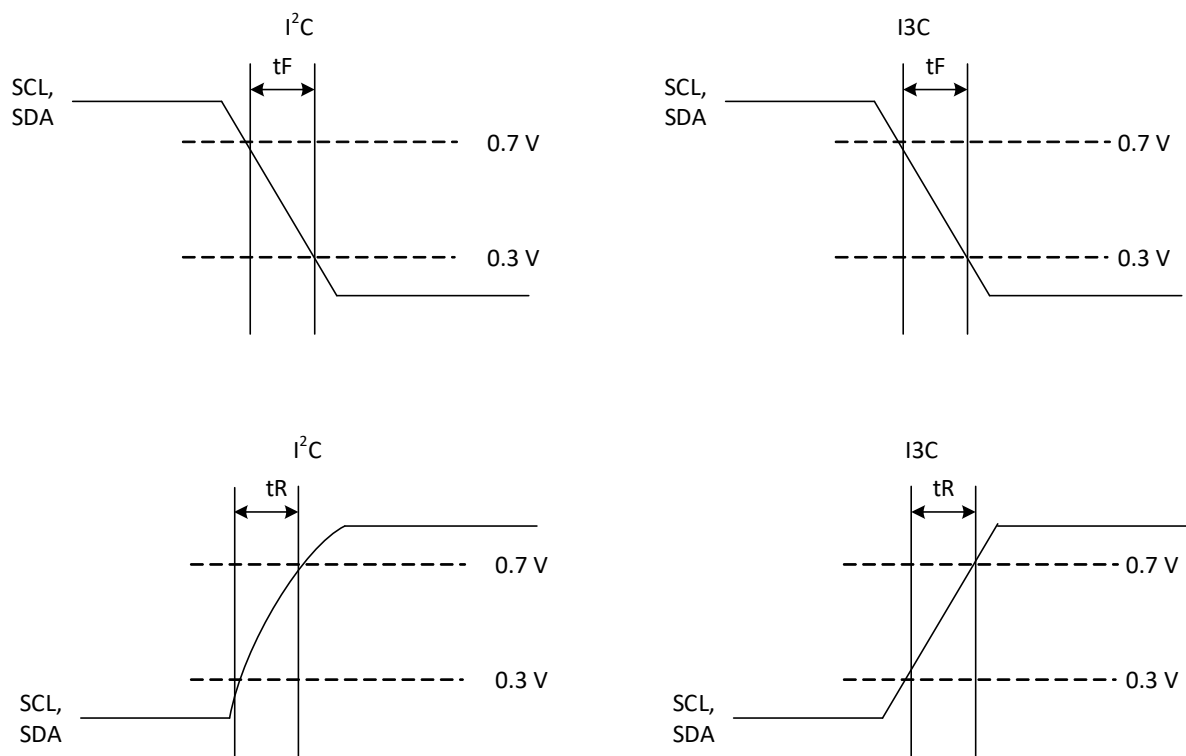


Figure 21 — Output Slew Rate Measurement Points

2.10.1 I²C or I3C Basic Bus Timing (cont'd)**Figure 22 — Rise and Fall Timing Parameter Definition**

2.10.1 I²C or I³C Basic Bus Timing (cont'd)

HSCL Non Monotonicity

Due to non-deterministic loading (number of DIMMs populated) on an unterminated bus, there can be reflections on the bus, causing slope reversal on the HSCL signal on the input receiver of the SPD5 HUB.

The SPD5 Hub device must tolerate t_{SLPR} and t_{SLPR_PK2PK} slope reversal on HSCL in I³C mode as shown in Figures 23 through 25.

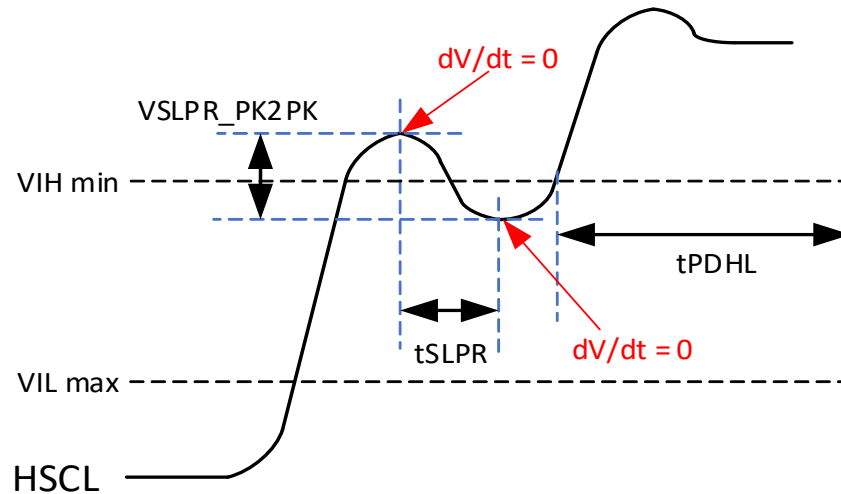


Figure 23 — Slope Reversal on HSCL at VIH

Note that tPDHL is for reference only in this diagram, refer to measurement methodology for details on this parameter.

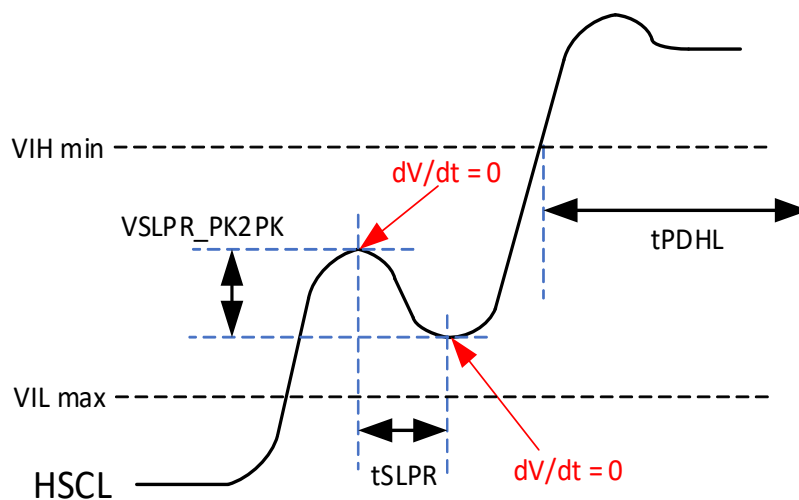


Figure 24 — Slope Reversal on HSCL Between VIL and VIH

Note that tPDHL is for reference only in this diagram, refer to measurement methodology for details on this parameter.

2.10.1 I²C or I3C Basic Bus Timing (cont'd)

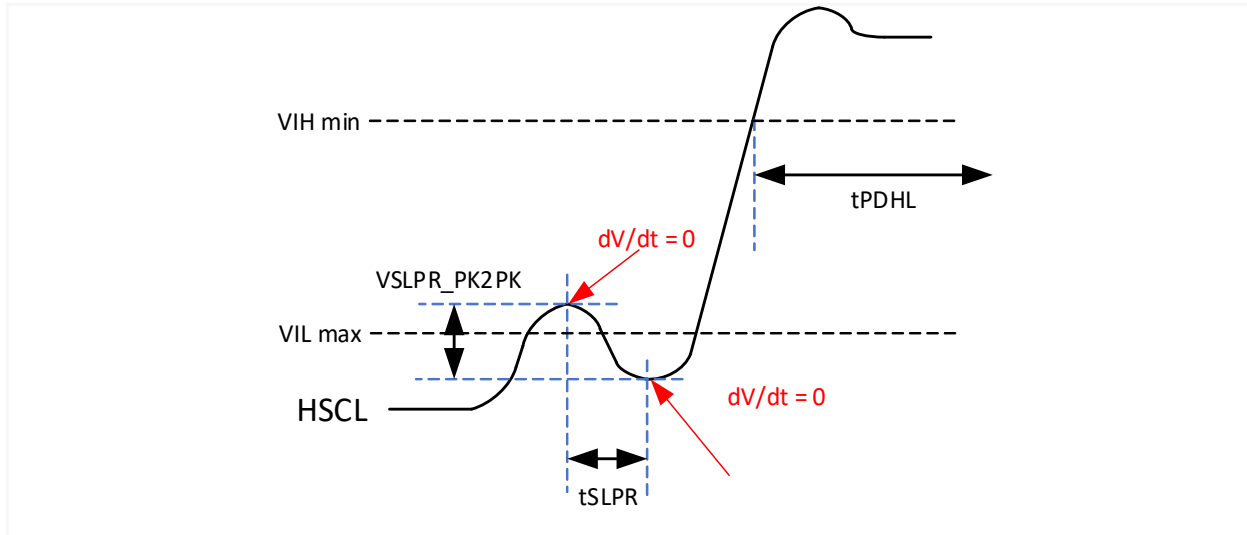


Figure 25 — Slope Reversal on HSCL at VIL

Note that tPDHL is for reference only in this diagram, refer to measurement methodology for details on this parameter.

Table 90 — HSCL Slope Reversal Parameters

| Symbol | Parameter | Min | Max | Unit | Notes |
|--|--|-----|-----|------|-------|
| t_{SLPR} | Pulse width of slope reversal which must be suppressed ¹ | 0 | 2.6 | ns | 1,2 |
| SLPR_PK2PK | The Peak to Peak voltage of slope reversal which must be suppressed ¹ | 0 | 150 | mV | 1,2 |
| NOTE 1 Verified by design and characterization, not necessarily tested on all devices. | | | | | |
| NOTE 2 These parameters apply in I3C mode. | | | | | |

2.10.2 Hub Propagation Delay

The SPD5 Hub device has a propagation delay of tPDHL for its host interface HSCL/HSDA input signals to the local interface LSCL/LSDA signals respectively.

Similarly, the SPD5 Hub device has a propagation delay of tPDLH for its local interface LSDA input signal to the host interface HSDA signal.

2.10.2 Hub Propagation Delay (cont'd)

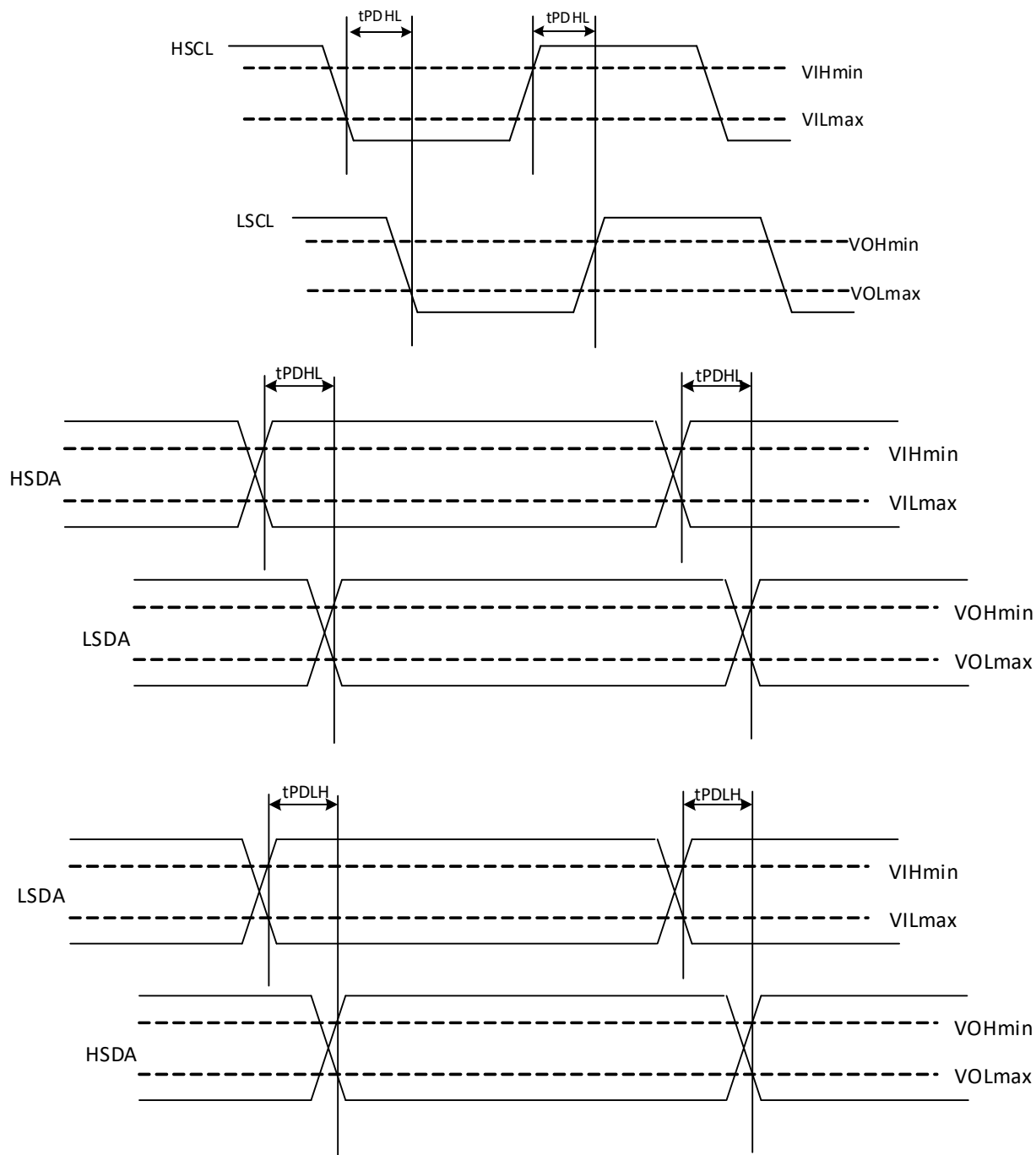


Figure 26 — Propagation Delay Through the SPD5 Hub Device

2.11 Parametric Characteristics

2.11.1 Absolute Maximum Ratings

Stressing the device above the rating listed in the Absolute Maximum Ratings table may cause permanent damage to the device. These are stress ratings only and operation of the device at these or any other conditions above those indicated in the Operating clauses of this standard is not implied. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability.

Table 91 — Absolute Maximum Ratings

| Symbol | Parameter | Min | Max | Units |
|---------------------------|-----------------------------|------|-----|-------|
| T _{STG} | Storage temperature | -65 | 150 | °C |
| V _{DDIO} | Supply Voltage | -0.5 | 2.1 | V |
| V _{DDSPD} | Supply voltage | -0.5 | 2.1 | V |
| HSA | HSA Pin | -0.5 | 2.1 | V |
| HSCL, HSDA, LSCL, LSDA | HSCL, HSDA, LSCL, LSDA Pins | -0.5 | 3.6 | V |

2.11.2 Operating Condition, Measurement Condition, and DC and AC Characteristics

This clause summarizes the operating and measurement conditions, and the DC and AC characteristics of the device. The parameters in the DC and AC Characteristic tables that follow are derived from tests performed under the Measurement Conditions summarized in the relevant tables.

Table 92 — Operating Conditions

| Symbol | Parameter | Min | Typ | Max | Units |
|---|---|------|-----|------|-------|
| V _{DDSPD} | Input Supply Voltage ¹ | 1.7 | 1.8 | 1.98 | V |
| V _{DDIO} | Input Supply Voltage ² | 0.95 | 1.0 | 1.05 | V |
| T _{CASE} | Case operating temperature | -40 | | 125 | °C |
| T _{WRITEOK} | Case temperature range for NVM Write operation Data writes outside this range may not meet retention requirements. | -40 | | 95 | °C |
| NOTE 1 For DDR5 DIMM application, the DDR5 PMIC VOUT_1.8 V setting should be selected such that absolute Min and Max values for SPD Hub standard are not violated. | | | | | |
| NOTE 2 For DDR5 DIMM application, the DDR5 PMIC VOUT_1.0 V setting should be selected such that absolute Min and Max values for SPD Hub specification are not violated. | | | | | |

2.11.2 Operating Condition, Measurement Condition, DC and AC Characteristics (cont'd)

Table 93 — Write Endurance and Data Retention Characteristics

| Parameter | Min | Typ | Max | Units |
|--------------------------------|---------------|-----|-----|--------|
| Data Retention 25 °C | TBD | | | Years |
| Write Endurance Cycles @ 25 °C | ≥ 100000 | | | Cycles |
| Write Endurance Cycles @ 95 °C | 100000 | | | Cycles |

Table 94 — AC Measurement Conditions¹

| Symbol | Parameter | Min | Max | Units |
|----------------|---|------------|-----|-------|
| C _L | Load capacitance | 40 | | pF |
| | Input rise and Fall times - Open Drain | - | TBD | ns |
| | Input rise and fall times - Push Pull | - | TBD | ns |
| | Input signal swing levels | 0.2 to 0.8 | | V |
| | Input levels for timing reference | 0.3 to 0.7 | | V |
| NOTE 1 | This AC measurement condition (Table 94 and Figure 27) is only for lab test purposes. | | | |

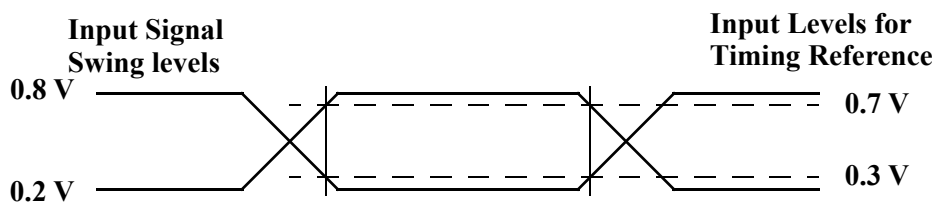


Figure 27 — AC Measurement Waveform

Table 95 — Input Parameters

| Symbol | Parameter ^{1,2} | Test Condition | Min | Max | Units |
|--|---|----------------------------|-----|-----|-------|
| C _{IN} | Input capacitance (HSDA, HSCL, LSDA) | -- | -- | 5 | pF |
| t _{SP} | Pulse width of spikes which must be suppressed by the input filter in I ² C mode | Single glitch, f ≤ 100 KHz | -- | -- | ns |
| | | Single glitch, f > 100 KHz | 0 | 50 | |
| NOTE 1 T _A = 25 °C, f = 400 kHz. | | | | | |
| NOTE 2 Verified by design and characterization, not necessarily tested on all devices. | | | | | |

2.11.2 Operating Condition, Measurement Condition, DC and AC Characteristics (cont'd)**Table 96 — Output Ron Specification**

| Symbol | Parameter | Min | Max | Units | Notes |
|--|--|-----|-----|-------|-------|
| R _{on_PUH} | HSDA Output Pullup Driver Impedance | 10 | 45 | Ohm | 1 |
| R _{on_PDH} | HSDA Output Pulldown Driver Impedance | 10 | 40 | Ohm | |
| R _{on_Local} | LSCL, LSDA Output Pullup and Pulldown Driver Impedance | 20 | 100 | Ohm | |
| NOTE 1 Pulldown Ron = Vout/Iout; Pullup Ron = (VIO - Vout)/Iout. | | | | | |

Table 97 — DC Characteristics

| Symbol | Parameter | Test Condition | Min | Max | Unit |
|---|---|---|-------|-----|------|
| I _{LI} | Input leakage current | | -- | ±5 | μA |
| I _{LO} | Output leakage current | | -- | ±5 | μA |
| I _{DDR} | Supply current, read operation | VDDSPD = 1.8 V, f _C = 12.5 MHz ¹ | -- | 2 | mA |
| I _{DDW} | Supply current, write operation | VDDSPD = 1.8V, f _C = 12.5 MHz ¹ | -- | 3 | mA |
| I _{DD1} | Standby Supply current | V _{IN} = VDDSPD = 1.8 V | -- | TBD | μA |
| V _{IL} | Input low voltage | -- | -0.35 | 0.3 | V |
| V _{IH} | Input high voltage | -- | 0.7 | 3.6 | V |
| V _{OL} | Output low voltage ² | 3 mA sink current | -- | 0.3 | V |
| V _{OH} | Output high voltage | 3 mA source current | 0.75 | - | V |
| I _{OL} | Output low current - HSDA, LSDA, LSCL | V _{OL} = 0.3 V | -3 | - | mA |
| I _{OH} | Output High current - HSDA, LSDA, LSCL | V _{OH} = V _{DDIO} - 0.3 V | - | -3 | mA |
| Slew_Rate | Rising Output Slew Rate (HSDA, LSDA, LSCL) | See Note 3 | 0.1 | 1.0 | V/ns |
| | Falling Output Slew Rate (HSDA, LSDA, LSCL) | | 0.1 | 3.0 | V/ns |
| V _{PON} | Power On Reset threshold | Monotonic rise between V _{PON} and V _{DDSPD} (min) without ringback | 1.6 | -- | V |
| V _{POFF} | Power Off threshold for warm power on cycle | No ringback above V _{POFF} | -- | 0.3 | V |
| NOTE 1 Thermal sensor is active. | | | | | |
| NOTE 2 Example calculation for V _{OL} = R _{on} * (I _{leakage} + I _{out}); where I _{out} = V _{DDIO} / (R _{pu} + R _{on}); R _{pu} = Pullup resistor with typical value of 1 KΩ; Min and Max value of 750 Ω and 1.5 KΩ, respectively. | | | | | |
| NOTE 3 Output slew rate is guaranteed by design and/or characterization. The output slew rate reference load as shown in Figure 20 and Figure 21 shows the timing measurement points. For slew rate measurements, the V _{OH} level shown in Figure 21 is a function of R _{on} value; V _{OH} = {1.0/(R _{on} + 50)} * 50. | | | | | |

2.11.2 Operating Condition, Measurement Condition, DC and AC Characteristics (cont'd)**Table 98 — AC Characteristics**

| | | I²C Mode - Open Drain | | I³C Basic Push-Pull¹ | | | |
|--------------------------------|---|---|------------|---|------------|--------------|--------------|
| Symbol | Parameter | Min | Max | Min | Max | Units | Notes |
| f _{SCL} | Clock frequency | 0.01 | 1 | 0 | 12.5 | MHz | |
| t _{HIGH} | Clock pulse width high time | 260 | -- | 35 | -- | ns | |
| t _{LOW} | Clock pulse width low time | 500 | -- | 35 | -- | ns | |
| t _{TIMEOUT} | Detect clock low timeout | 10 | 50 | 10 | 50 | ms | |
| t _R | SDA rise time | -- | 120 | -- | 5 | ns | 2,3 |
| t _F | SDA fall time | -- | 120 | -- | 5 | ns | 2,3 |
| t _{SU:DAT} | Data in setup time | 50 | -- | 8 | -- | ns | 2 |
| t _{HD:DI} | Data in hold time | 0 | | 3 | -- | ns | 2 |
| t _{SU:STA} | Start condition setup time | 260 | -- | 12 | -- | ns | 2 |
| t _{HD:STA} | Start condition hold time | 260 | -- | 30 | -- | ns | 2 |
| t _{SU:STO} | Stop condition setup time | 260 | -- | 12 | -- | ns | 2 |
| t _{BUF} | Time between Stop Condition and next Start Condition | 500 | -- | 500 | -- | ns | 2,4 |
| t _W | Write time | -- | 5 | | 5 | ms | |
| t _{POFF} | Warm power cycle off time | 1 | -- | 1 | -- | ms | |
| t _{Sense_HSA} | Time from valid 1.8V supply to Sense HSA pin for HID code assignment | - | 5 | - | 5 | ms | |
| t _{INIT} | Time from power on to first command | 10 | -- | 10 | -- | ms | |
| t _{RST} | Device reinitialization time | | | TBD | TBD | ms | |
| t _{AVAIL} | Bus Available time (no edges seen on HSDA and HSCL) | | | 1 | -- | μs | |
| t _{IBI_ISSUE} | Time to issue IBI after an event is detected when Bus is available | - | - | - | 15 | μs | |
| t _{CLR_I3C_CMD_Delay} | Time from Clear Register Status to any I3C operation with Start condition to avoid IBI generation; PEC Disabled | - | - | 4 | - | μs | |
| | Time from Clear Register Status to any I3C operation with Start condition to avoid IBI generation; PEC Enabled | - | - | 15 | - | μs | |
| t _{PDHL} | Propagation Delay, HSDA to LSDA and HSCL to LSCL | | 6 | | 6 | ns | 5 |
| t _{PDLH} | Propagation Delay, LSDA to HSDA | | 6 | | 6 | ns | 5 |
| t _{HD:DAT} | HSCL Falling Clock In to HSDA Data Out Hold Time | 0.5 | 350 | N/A | N/A | ns | 6 |
| t _{DOUT} | HSCL Falling Clock In to HSDA Valid Data Out Time | N/A | N/A | 0.5 | 12 | ns | 7 |
| t _{DOFFT} | HSCL Rising Clock In to SDA Output Off | N/A | N/A | 0.5 | 12 | ns | 8 |

2.11.2 Operating Condition, Measurement Condition, DC and AC Characteristics (cont'd)**Table 98 — AC Characteristics (cont'd)**

| | | I²C Mode - Open Drain | | I3C Basic Push-Pull¹ | | | |
|--|--|---|------------|--|-------------------|--------------|--------------|
| Symbol | Parameter | Min | Max | Min | Max | Units | Notes |
| t _{DOFFC} | HSCL Rising Clock In to Controller SDA Output Off | N/A | N/A | 0.5 | t _{HIGH} | ns | 9 |
| t _{CL_r_DAT_f} | HSCL Rising Clock In to Controller Driving HSDA Signal Low | N/A | N/A | 40 | - | ns | 10 |
| t _{DEVCTRLCCC_DELAY_PEC_DIS} | DEVCTRL CCC Followed by DEVCTRL CCC or Register Read/Write Command Delay | 3 | - | 3 | - | μs | 11,12,13 |
| t _{WR_RD_DELAY_PEC_EN} | Register Write Command Followed by Register Read Command Delay in PEC Enabled Mode | N/A | N/A | 8 | - | μs | 14,15,16 |
| t _{I2C_CCC_Update_Delay} | SETHID CCC or SETAASA CCC to any other CCC or Read/Write Command delay | 2.5 | - | - | - | μs | |
| t _{I3C_CCC_Update_Delay} | RSTDAA CCC or ENEC CCC or DISEC CCC to any other CCC or Read/Write Command Delay | - | - | 2.5 | - | μs | |
| t _{CCC_Delay} | Any CCC to RSTDAA CCC delay | N/A | N/A | 2.5 | - | μs | |
| <p>NOTE 1 I3C mode with Open Drain operation follows timing values as shown in I²C Mode - Open Drain column.</p> <p>NOTE 2 See Figure 17 for input timing parameter definition.</p> <p>NOTE 3 See Figure 22 for voltage threshold definition for rise and fall times.</p> <p>NOTE 4 If PEC is enabled, t_{WR_RD_DELAY_PEC_EN} timing parameter also applies.</p> <p>NOTE 5 See Figure 26 for timing definition. See Figure 20 for output timing parameter measurement reference load.</p> <p>NOTE 6 See Figure 19 for output timing parameter definition.</p> <p>NOTE 7 The SPD5 Hub device must be configured in I3C Basic mode to guarantee t_{DOUT} value. See Figure 18 for output timing parameter definition. See Figure 20 for output timing parameter measurement reference load.</p> <p>NOTE 8 The SPD5 Hub device must be configured in I3C Basic mode to guarantee t_{DOFFT} value. See Figure 7. See Figure 20 for output timing parameter measurement reference load.</p> <p>NOTE 9 The SPD5 Hub device must be configured in I3C Basic mode. The Host guarantees t_{DOFFC} value. See Figure 8. See Figure 20 for output timing parameter measurement reference load. Also refer to MIPI Alliance Specification for I3C Basic Version 1.0-19 July 2018, clause 5.1.2.3.2, Transition from Address ACK to Mandatory Byte during IBI.</p> <p>NOTE 10 See Figure 10.</p> <p>NOTE 11 From STOP condition of DEVCTRL CCC to START condition for Register Read or Register Write Command Data Packet delay.</p> <p>NOTE 12 The device sends NACK if Host does not satisfy t_{DEVCTRLCCC_DELAY_PEC_DIS} timing parameter.</p> <p>NOTE 13 This timing parameter restriction is only applicable when PEC function is disabled in SPD5 Hub. If PEC is enabled, this timing parameter does not apply.</p> <p>NOTE 14 From STOP condition for Register Write Command Data Packet to START condition for Register Read Command Data Packet delay.</p> <p>NOTE 15 This timing parameter restriction is only applicable when PEC function is enabled in SPD5 Hub. If PEC is disabled, this timing parameter does not apply.</p> <p>NOTE 16 The SPD5 Hub sends NACK if Host does not satisfy t_{WR_RD_DELAY_PEC_EN} timing parameter.</p> | | | | | | | |

2.11.3 Temperature Sensor Performance

Table 99 — Temperature Sensor Performance

| Parameter | Min | Typ | Max | Unit | Test Conditions |
|--|-----|------|------|------|---|
| Temperature Sensor Accuracy (Active Range) | -- | ±0.5 | ±1.0 | °C | $75\text{ °C} \leq T_A \leq 95\text{ °C}$ |
| Temperature Sensor Accuracy (Monitor Range) | -- | ±1.0 | ±2.0 | °C | $40\text{ °C} \leq T_A \leq 125\text{ °C}$ |
| Temperature Sensor Accuracy (Industrial Temperature Range) | -- | ±2.0 | ±3.0 | °C | $-40\text{ °C} \leq T_A \leq 125\text{ °C}$ |
| Resolution | | 0.25 | | °C | |
| Conversion Time | | | 68 | ms | Assumes 0.25 °C accuracy |
| Hysteresis between temperature events | 1 | - | - | °C | |

2.12 Device Package and Pinout

The SPD5 Hub device is packaged in a 9-contact thermally enhanced PSON-8, MO-229 Variation V/W/UCED-3. The package size is 2.0 mm x 3.0 mm.

2.12.1 Package Pinout

The pinout is shown in Figure 28. The pinout is a BOTTOM view.

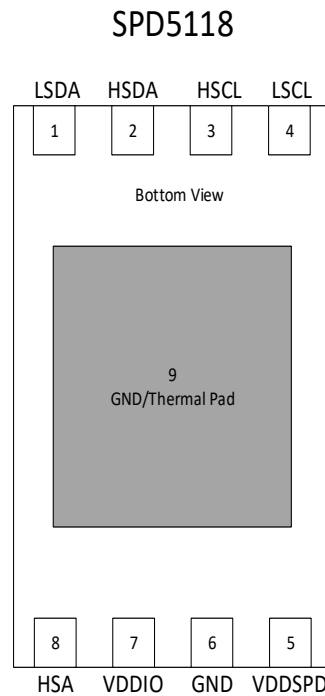
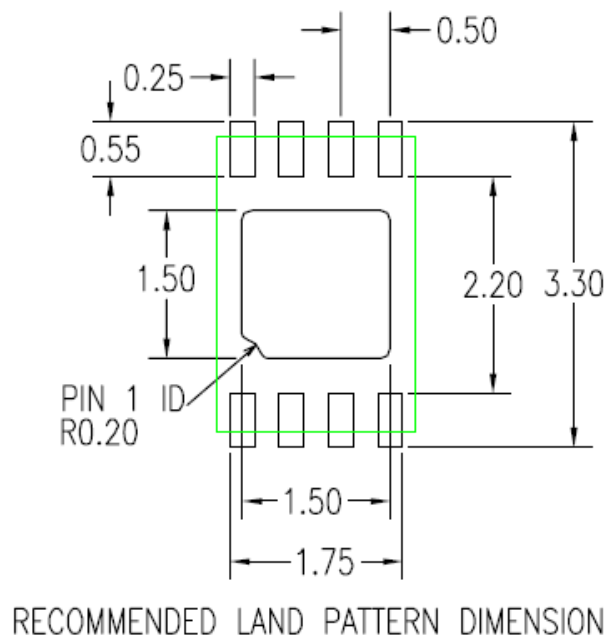


Figure 28 — Pinout for SPD5 Hub Devices

2.12.2 Recommended Landing Pattern



NOTES:

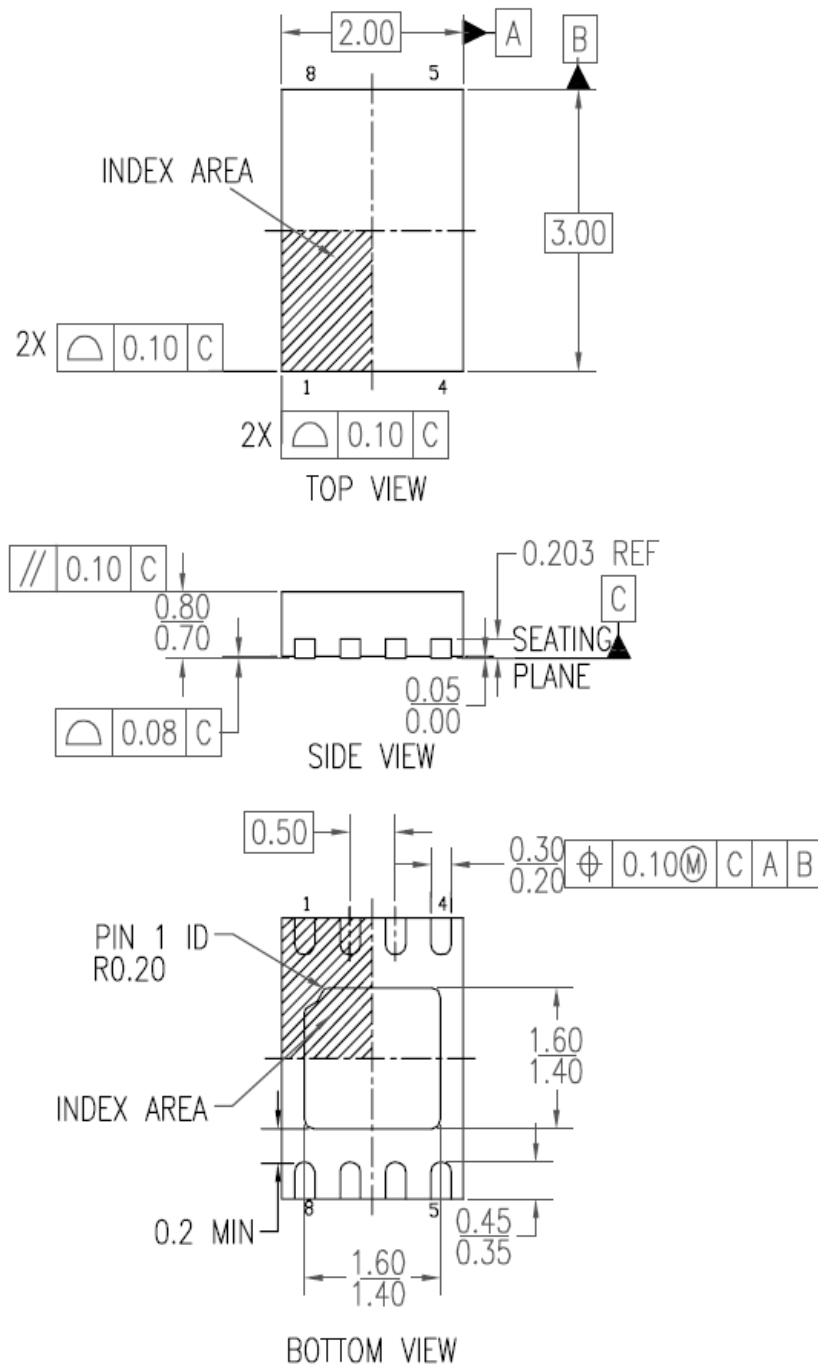
1. ALL DIMENSIONS ARE IN MM. ANGLES IN DEGREES.
2. TOP DOWN VIEW. AS VIEWED ON PCB.
3. LAND PATTERN RECOMMENDATION PER IPC-7351B GENERIC REQUIREMENT FOR SURFACE MOUNT DESIGN AND LAND PATTERN.
4. COMPONENT OUTLINE SHOW FOR REFERENCE IN GREEN.

Figure 29 — Common Landing Array

2.12.3 Mechanical Outline Drawing

The side view in Figure 30 shows only MO-229 variation W2030D-3 (formerly identified as WCED-3). However, other variations U/V are also allowed.

2.12.3 Mechanical Outline Drawing (cont'd)



NOTE:

1. DIMENSIONING AND TOLERANCING CONFORM TO ASME Y14.5M-1994
2. ALL DIMENSIONS ARE IN MILLIMETERS ANGLES ARE IN DEGREES
3. PIN 1 LOCATION IS EITHER IDENTIFIED BY NOTCH OR CHAMFER

Figure 30 — Mechanical Outline Drawing

3 Volatile Registers Space

3.1 Access Mechanism

3.1.1 Register Attribute Definition

All volatile registers have Base Attributes as defined in Table 100. Some register attributes are further modified with Attribute Modifiers, as defined in Table 101.

The volatile register space has a continuous address. Unlike Non-Volatile memory in SPD5 Hub device, there is no concept of “Block” memory in volatile register space. When writing to and reading from volatile register space (i.e., MemReg = 0), the “Block Address bits” are treated simply as Upper address bits.

Table 100 — Register Base Attributes

| Attribute | Abbreviation | Description |
|------------|--------------|---|
| Read Only | R | This bit can be read by software. Writes have no effect. |
| Read/Write | RW | This bit can be read or written by software. |
| Write Only | W | This bit can only be written by software. |
| Reserved | RV | This bit is reserved for future expansion and its value must not be modified by software. The bit will return ‘0’ when read. When writing this bit, software must preserve the value read unless otherwise indicated. |

Table 101 — Register Attribute Modifier

| Attribute | Abbreviation | Description |
|--------------|--------------|---|
| Write 1 Only | 1O | This bit can only be set (i.e., write ‘1’) but not reset (i.e., write ‘0’) |
| Protected | P | This bit is protected by the password registers TBD. This bit cannot be written to unless the password code has been written into the password registers. |
| Persistent | E | Persistent. |

3.2 Registers

3.2.1 Register Map

Table 102 — Register Map

| Register Name | Register Address (hex) | Attribute | Description |
|-------------------|------------------------|------------|--|
| Table 105, “MR0” | 0x00 | ROE | Device Type; Most Significant Byte |
| Table 106, “MR1” | 0x01 | ROE | Device Type; Least Significant Byte |
| Table 107, “MR2” | 0x02 | ROE | Device Revision |
| Table 108, “MR3” | 0x03 | ROE | Vendor ID Byte 0 |
| Table 109, “MR4” | 0x04 | ROE | Vendor ID Byte 1 |
| Table 110, “MR5” | 0x05 | ROE | Device Capability |
| Table 111, “MR6” | 0x06 | ROE | Device Write Recovery Time Capability |
| MR7 to MR10 | 0x07 to 0x0A | RV | Reserved |
| Table 112, “MR11” | 0x0B | RW | I ² C Legacy Mode Device Configuration |
| Table 113, “MR12” | 0x0C | RWE | Write Protection For NVM Blocks [7:0] |
| Table 114, “MR13” | 0x0D | RWE | Write Protection for NVM Blocks [15:8] |
| Table 115, “MR14” | 0x0E | RWE | Device Configuration - Host and Local Interface IO; DO NOT USE [4:0] |
| Table 116, “MR15” | 0x0F | RV | DO NOT USE |
| MR16 to MR17 | 0x10 to 0x11 | RV | Reserved |
| Table 118, “MR19” | 0x12 | RO, RW | Device Configuration |
| Table 118, “MR19” | 0x13 | IO | Clear Register MR51 Temperature Status Command |
| Table 119, “MR20” | 0x14 | IO | Clear Register MR52 Error Status Command |
| MR21 to MR25 | 0x15 to 0x19 | RV | Reserved |
| Table 120, “MR26” | 0x1A | RW | TS Configuration |
| Table 122, “MR28” | 0x1B | IO, RO, RW | Interrupt Configurations |
| Table 122, “MR28” | 0x1C | RW | TS Temperature High Limit Configuration - Low Byte |
| Table 123, “MR29” | 0x1D | RW | TS Temperature High Limit Configuration - High Byte |
| Table 124, “MR30” | 0x1E | RW | TS Temperature Low Limit Configuration - Low Byte |
| Table 125, “MR31” | 0x1F | RW | TS Temperature Low Limit Configuration - High Byte |
| Table 126, “MR32” | 0x20 | RW | TS Critical Temperature High Limit Configuration - Low Byte |
| Table 127, “MR33” | 0x21 | RW | TS Critical Temperature High Limit Configuration - High Byte |
| Table 128, “MR34” | 0x22 | RW | TS Critical Temperature Low Limit Configuration - Low Byte |
| Table 129, “MR35” | 0x23 | RW | TS Critical Temperature Low Limit Configuration - High Byte |
| MR36 to MR47 | 0x24 to 0x2F | RV | Reserved for Device Configuration Type of Registers |
| Table 130, “MR48” | 0x30 | RO | Device Status |
| Table 131, “MR49” | 0x31 | RO | TS Current Sensed Temperature - Low Byte |
| Table 132, “MR50” | 0x32 | RO | TS Current Sensed Temperature - High Byte |
| Table 133, “MR51” | 0x33 | RO | TS Temperature Status |
| Table 134, “MR52” | 0x34 | RO | Hub, Thermal and NVM Error Status |
| MR53 to MR127 | 0x35 to 0x7F | RV | Reserved |

3.2.2 Thermal Sensor Registers Read Out Mechanism

All thermal registers are sixteen bit quantities stored in two consecutive registers; low byte first and then high byte. Five bits are reserved for future use. Reserved bits are Read only bits and must be set to '0' when host writes to low and high byte. The device returns '0' in reserved bits when host reads from the low and high byte. Remaining eleven bits in these paired register form a signed value of multiples of 0.25 ranging from -256.00 to + 255.75. Unit for all thermal registers is °C.

The format of each pair of thermal registers is shown in Table 103.

Table 103 — Thermal Register - Low Byte and High Byte

| Register | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|-----------|------|------|------|------|-----|------|------|------|
| MRX | Low Byte | 8 | 4 | 2 | 1 | 0.5 | 0.25 | RSVD | RSVD |
| MRX + 1 | High Byte | RSVD | RSVD | RSVD | Sign | 128 | 64 | 32 | 16 |

The examples (reserved bits in red, sign bit highlighted in cyan) are shown in Table 104.

Table 104 — Thermal Register Examples

| High Byte | Low Byte | Value | Unit |
|------------|------------|---------|------|
| 000 0 0101 | 1111 00 00 | +95.00 | °C |
| 000 0 0101 | 0101 00 00 | + 85.00 | °C |
| 000 0 0100 | 1011 00 00 | + 75.00 | °C |
| 000 0 0000 | 0001 00 00 | + 1.00 | °C |
| 000 0 0000 | 0000 11 00 | + 0.75 | °C |
| 000 0 0000 | 0000 10 00 | + 0.50 | °C |
| 000 0 0000 | 0000 01 00 | + 0.25 | °C |
| 000 0 0000 | 0000 00 00 | 0.00 | °C |
| 000 1 1111 | 1111 11 00 | -0.25 | °C |
| 000 1 1111 | 1111 10 00 | -0.50 | °C |
| 000 1 1111 | 1111 01 00 | -0.75 | °C |
| 000 1 1111 | 1111 00 00 | -1.00 | °C |
| 000 1 1101 | 1000 00 00 | -40.00 | °C |

3.2.3 Register Description

Table 105 — MR0

| Addr | MR0 | | Device Type; Most Significant Byte ¹ |
|--|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | ROE | 0x51 | MR0[7:0]: MSB_DEV_TYPE Device Type - SPD5 Hub Device |
| NOTE 1 The code in this register is used in conjunction with any device type in Table 106, “MR1” register. | | | |

Table 106 — MR1

| Addr | MR1 | | Device Type; Least Significant Byte ¹ |
|--|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | ROE | - | MR1[7:0]: LSB_DEV_TYPE Device Type 0x18: w/ Temperature Sensor; SPD5 Hub Device |
| NOTE 1 The code in this register is used in conjunction with any device type in Table 105, “MR0” register. | | | |

Table 107 — MR2

| Addr | MR2 | | Device Revision |
|------|------|---------|---|
| Bits | Attr | Default | Description |
| 7:6 | RV | 0 | MR2[7:6]: Reserved |
| 5:4 | ROE | - | MR2[5:4]: DEV_REV_MAJOR Major Revision 00 = Revision 1 01 = Revision 2 10 = Revision 3 11 = Revision 4 |
| 3:1 | ROE | - | MR2[3:1]: DEV_REV_MINOR Minor Revision 000 = Revision 0 001 = Revision 1 010 = Revision 2 .. 111 = Revision 8 |
| 0 | RV | 0 | MR2[0]: Reserved |

Table 108 — MR3

| Addr | MR3 | | Device Revision |
|------|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | ROE | - | MR3[7:0]: VENDOR_ID_BYTE0 Vendor ID Byte 0 |

Table 109 — MR4

| Addr | MR4 | | Device Revision |
|------|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | ROE | - | MR4[7:0]: VENDOR_ID_BYTE1 Vendor ID Byte 1 |

3.2.3 Register Description (cont'd)**Table 110 — MR5**

| Addr | MR5 | | Device Capability |
|------|------|---------|---|
| Bits | Attr | Default | Description |
| 7:2 | RV | 0 | MR5[7:2]:Reserved |
| 1 | ROE | - | MR6[1]: TS_SUPPORT Internal Temperature Sensor Support 0 = Does not support Temperature Sensor 1 = Supports Temperature Sensor |
| 0 | ROE | 1 | MR6[0]: HUB_SUPPORT Hub Function Support 0 = Does not support Hub function 1 = Supports Hub function |

Table 111 — MR6

| Addr | MR6 | | Write Recovery Time |
|------|------|---------|--|
| Bits | Attr | Default | Description |
| 7:4 | ROE | - | MR6[7:4]:WR_REC_UNIT Write Recovery Unit 0000 = 0 0001 = 1 0010 = 2 0011 = 3 0100 = 4 0101 = 5 0110 = 6 0111 = 7 1000 = 8 1001 = 9 1010 = 10 1011 = 50 1100 = 100 1101 = 200 1110 = 500 1111 = Reserved |
| 3:2 | RV | 0 | MR6[3:2]:Reserved |
| 1:0 | ROE | - | MR6[1:0]: WR_REC_UNIT_TIME Write Recovery Time Unit 00 = ns 01 = μ s 10 = ms 11 = Reserved |

3.2.3 Register Description (cont'd)

Table 112 — MR11

| Addr | MR11 | | I ² C Legacy Mode Device Configuration |
|---|------|---------|---|
| Bits | Attr | Default | Description |
| 7:4 | RV | 0 | MR11[7:4]:Reserved |
| 3 | RW | 0 | MR11[3]:I ² C_LEGACY_MODE_ADDR SPD5 Hub Device - I ² C Legacy Mode Addressing 0 = 1 Byte Addressing for SPD5 Hub Device Memory 1 = 2 Bytes Addressing for SPD5 Hub Device Memory |
| 2:0 | RW | 000 | MR11[2:0]: I ² C_LEGACY_MODE_ADDR_POINTER SPD5 Device - Non Volatile Memory Address Page Pointer in I ² C Legacy Mode ^{1,2,3} 000 = Page 0 (0x00 to 0x7F) 001 = Page 1 (0x80 to 0xFF) 010 = Page 2 (0x100 to 0x17F) 011 = Page 3 (0x180 to 0x1FF) 100 = Page 4 (0x200 to 0x27F) 101 = Page 5 (0x280 to 0x2FF) 110 = Page 6 (0x300 to 0x37F) 111 = Page 7 (0x380 to 0x3FF) |
| NOTE 1 This register is only applicable if bit [3] = '0' and Table 117, "MR18" [5] = '0'. The SPD5 Hub device does not incur any delay to switch from one page to another page. | | | |
| NOTE 2 This register only applies to non-volatile memory (1024 Bytes) access of SPD5 Hub device. For volatile memory access, this register must be programmed to '000'. | | | |
| NOTE 3 See clause 2.8.1 for the NVM Write and Read operation when device reaches the last byte of the 16 byte block boundary. | | | |

3.2.3 Register Description (cont'd)**Table 113 — MR12**

| Addr | MR12 | | NVM Protection Configuration For Blocks [7:0] ^{1,2} |
|--|------|---------|--|
| Bits | Attr | Default | Description |
| 7 | RWE | - | MR12[7]: WP_BLK_7 Write Protect - Block 7 0 = Not Protected 1 = Protected |
| 6 | RWE | - | MR12[6]: WP_BLK_6 Write Protect - Block 6 0 = Not Protected 1 = Protected |
| 5 | RWE | - | MR12[5]: WP_BLK_5 Write Protect - Block 5 0 = Not Protected 1 = Protected |
| 4 | RWE | - | MR12[4]: WP_BLK_4 Write Protect - Block 4 0 = Not Protected 1 = Protected |
| 3 | RWE | - | MR12[3]: WP_BLK_3 Write Protect - Block 3 0 = Not Protected 1 = Protected |
| 2 | RWE | - | MR12[2]: WP_BLK_2 Write Protect - Block 2 0 = Not Protected 1 = Protected |
| 1 | RWE | - | MR12[1]: WP_BLK_1 Write Protect - Block 1 0 = Not Protected 1 = Protected |
| 0 | RWE | - | MR12[0]: WP_BLK_0 Write Protect - Block 0 0 = Not Protected 1 = Protected |
| NOTE 1 Once any register bit is set to '1', it can only be cleared when the SPD5 Hub device is in offline tester mode of operation. | | | |
| NOTE 2 The write (or update) transaction to this register must be followed by STOP operation to allow SPD5 Hub device to update the setting. | | | |

3.2.3 Register Description (cont'd)

Table 114 — MR13

| Addr | MR13 | | NVM Protection Configuration For Blocks [15:8] ^{1,2} |
|--|------|---------|--|
| Bits | Attr | Default | Description |
| 7 | RWE | - | MR13[7]: WP_BLK_15 Write Protect - Block 15 0 = Not Protected 1 = Protected |
| 6 | RWE | - | MR13[6]: WP_BLK_14 Write Protect - Block 14 0 = Not Protected 1 = Protected |
| 5 | RWE | - | MR13[5]: WP_BLK_13 Write Protect - Block 13 0 = Not Protected 1 = Protected |
| 4 | RWE | - | MR13[4]: WP_BLK_12 Write Protect - Block 12 0 = Not Protected 1 = Protected |
| 3 | RWE | - | MR13[3]: WP_BLK_11 Write Protect - Block 11 0 = Not Protected 1 = Protected |
| 2 | RWE | - | MR13[2]: WP_BLK_10 Write Protect - Block 10 0 = Not Protected 1 = Protected |
| 1 | RWE | - | MR13[1]: WP_BLK_9 Write Protect - Block 9 0 = Not Protected 1 = Protected |
| 0 | RWE | - | MR13[0]: WP_BLK_8 Write Protect - Block 8 0 = Not Protected 1 = Protected |
| NOTE 1 Once any register bit is set to '1', it can only be cleared when the SPD5 Hub device is in offline tester mode of operation. | | | |
| NOTE 2 The write (or update) transaction to this register must be followed by STOP operation to allow SPD5 Hub device to update the setting. | | | |

3.2.3 Register Description (cont'd)**Table 115 — MR14**

| Addr | MR14 | | Device Configuration- Local Interface ^{1,2} |
|---|------|---------|--|
| Bits | Attr | Default | Description |
| 7:6 | RV | 0 | MR14[7:6]:Reserved |
| 5 | RWE | 0 | MR14[5]: LOCAL_INF_PULLUP_CONF Local Interface - Pull Up Resistor Configuration 0 = Internal (on die) Pullup Resistor ³ 1 = External (board) Pullup Resistor |
| 4:0 | RV | 0 | MR14[4:0]: DO NOT USE |
| NOTE 1 DIMM Vendor configures this register during assembly based on the DIMM design. After SPD Hub device is powered up, the Host can alter the setting through this register. | | | |
| NOTE 2 The write (or update) transaction to this register must be followed by STOP operation to allow SPD5 Hub device to update the setting. | | | |
| NOTE 3 Typical value of on die pullup resistor is 1 KΩ. The minimum and maximum on die pullup resistor value are 750 Ω and 1.5 KΩ, respectively. | | | |

Table 116 — MR15

| Addr | MR15 | | DO NOT USE |
|------|------|---------|-----------------------|
| Bits | Attr | Default | Description |
| 7:0 | RV | 0 | MR15[7:0]: DO NOT USE |

3.2.3 Register Description (cont'd)

Table 117 — MR18

| Addr | MR18 | | Device Configuration ¹ |
|--|------|---------|--|
| Bits | Attr | Default | Description |
| 7 | RW | 0 | MR18[7]: PEC_EN PEC Enable ^{2,3} 0 = Disable 1 = Enable |
| 6 | RW | 0 | MR18[6]: PAR_DIS Parity (T bit) Disable ^{3,4} 0 = Enable 1 = Disable |
| 5 | RO | 0 | MR18[5]: INF_SEL Interface Selection 0 = I ² C Protocol (Max speed of 1 MHz) 1 = I3C Basic Protocol ⁵ |
| 4 | RW | 0 | MR18[4]: DEF_RD_ADDR_POINT_EN Default Read Address Pointer Enable 0 = Disable Default Read Address Pointer (Address pointer is set by the Host) ⁶ 1 = Enable Default Read Address Pointer; Address selected by register bits [3:2] |
| 3:2 | RW | 0 | MR18[3:2]: DEF_RD_ADDR_POINT_START Default Read Pointer Starting Address ⁷ 00 = Table 131, "MR49" 01 = Reserved 10 = Reserved 11 = Reserved |
| 1 | RW | 0 | MR18[1]: DEF_RD_ADDR_POINT_BL Burst Length for Read Pointer Address for PEC Calculation ⁸ 0 = 2 Bytes 1 = 4 Bytes |
| 0 | RV | 0 | MR18[0]: Reserved |
| <p>NOTE 1 The write (or update) transaction to this register must be followed by STOP operation to allow the SPD5 Hub device to update the setting.</p> <p>NOTE 2 This register is only applicable if Table 117, "MR18" [5] = '1'.</p> <p>NOTE 3 This register is updated when RSTDAA CCC is registered by SPD5 Hub device or when SPD5 Hub device goes through bus reset as described in clause 2.6.21.</p> <p>NOTE 4 This register is only applicable if Table 117, "MR18" [5] = '1'. When Parity function is disabled, the SPD5 Hub device simply ignores the "T" bit information from the Host. The host may actually choose to compute the parity and send that information in "T" bit or simply drive static low or high in "T" bit.</p> <p>NOTE 5 This register is automatically updated when SETAASA CCC or RSTDAA CCC is registered by the SPD5 Hub device or when SPD5 Hub device goes through bus reset as described in clause 2.6.21. This register can be read by the Host through normal Read operation but it cannot be written with normal write operation either in I²C mode or I3C Basic mode of operation. When this register is updated, it takes in effect when there is a next START operation (i.e., after STOP operation).</p> <p>NOTE 6 The setting in register Table 117, "MR18" [3:1] is a don't care.</p> <p>NOTE 7 This register is only applicable if Table 117, "MR18" [4] = '1'.</p> <p>NOTE 8 This register is only applicable if Table 117, "MR18" [7, 4] = '11'.</p> | | | |

3.2.3 Register Description (cont'd)**Table 118 — MR19**

| Addr | MR19 | | Clear Register Command ¹ |
|--|------|---------|---|
| Bits | Attr | Default | Description |
| 7:4 | RV | 0 | MR19[7:4]: Reserved |
| 3 | 1O | 0 | MR19 [3]: CLR_TS_CRIT_LOW Clear Temperature Sensor Critical Low Status 1 = Clear Table 133, “MR51” [3] Register |
| 2 | 1O | 0 | MR19 [2]: CLR_TS_CRIT_HIGH Clear Temperature Sensor Critical High Status 1 = Clear Table 133, “MR51” [2] Register |
| 1 | 1O | 0 | MR19 [1]: CLR_TS_LOW Clear Temperature Sensor Low Status 1 = Clear Table 133, “MR51” [1] Register |
| 0 | 1O | 0 | MR19 [0]: CLR_TS_HIGH Clear Temperature Sensor High Status 1 = Clear Table 133, “MR51” [0] Register |
| NOTE 1 This entire register is self clearing register after corresponding register is cleared. | | | |

Table 119 — MR20

| Addr | MR20 | | Clear Register Command ¹ |
|--|------|---------|--|
| Bits | Attr | Default | Description |
| 7 | 1O | 0 | MR20[7]: CLR_SPD_BUSY_ERROR Clear Write or Read Attempt while SPD Device Busy Error Status 1 = Clear + [7] Register |
| 6 | 1O | 0 | MR20[6]: CLR_WR_NVM_BLK_ERROR Clear Write Attempt to Protected NVM Block Error Status 1 = Clear Table 134, “MR52” [6] Register |
| 5 | 1O | 0 | MR20[5]: CLR_WR_NVM_PRO_REG_ERROR Clear Write Attempt to NVM Protection Register Error Status 1 = Clear Table 134, “MR52” [5] Register |
| 4:2 | RV | 0 | MR20[4:2]: Reserved |
| 1 | 1O | 0 | MR20[1]: CLR_PEC_ERROR Clear Packet Error Status 1 = Clear Table 134, “MR52” [1] Register |
| 0 | 1O | 0 | MR20[0]: CLR_PAR_ERROR Clear Parity Error Status 1 = Clear Table 134, “MR52” [0] Register |
| NOTE 1 This entire register is self clearing register after corresponding register is cleared. | | | |

3.2.3 Register Description (cont'd)

Table 120 — MR26

| Addr | MR26 | | Thermal Sensor Configuration |
|---|------|---------|---|
| Bits | Attr | Default | Description |
| 7:1 | RV | 0 | MR26[7:1] Reserved |
| 0 | RW | 0 | MR26[0]: DIS_TS Disable Temperature Sensor ^{1,2} 0 = Enable thermal sensor 1 = Disable thermal sensor |
| NOTE 1 This register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18 | | | |
| NOTE 2 If this bit is set to ‘1’ and then reset to ‘0’, the host must wait minimum of tINIT before accessing samples on the thermal sensor. | | | |

3.2.3 Register Description (cont'd)

Table 121 — MR27

| Addr | MR27 | | Interrupt Configuration |
|--|------|---------|---|
| Bits | Attr | Default | Description |
| 7 | IO | 0 | MR27[7]: CLR_GLOBAL Global Clear Event Status and In Band Interrupt Status ^{1,2} 1 = Clear Table 130, “MR48” [7], Table 133, “MR51” [3:0], and Table 134, “MR52” [7:5,3,1:0] Register |
| 6:5 | RV | 0 | MR27[6:5]: Reserved |
| 4 | RO | 0 | MR27[4]: IBI_ERROR_EN In Band Error Interrupt Enable for Table 134, “MR52” Error Log ³ 0 = Disable; Errors logged in Table 134, “MR52” [7:5, 1:0] registers do not generate an IBI to Host 1 = Enable; Errors logged in Table 134, “MR52” [7:5, 1:0] registers generates an IBI to Host |
| 3 | RW | 0 | MR27[3]: IBI_TS_CRIT_LOW_EN In Band Error Interrupt Enable for Temperature Sensor Critical Low 0 = Disable; Table 133, “MR51” [3] = ‘1’ does not generate an IBI to Host 1 = Enable; Table 133, “MR51” [3] = ‘1’ and Table 121, “MR27” [4] = ‘1’ generate an IBI to Host |
| 2 | RW | 0 | MR27[2]: IBI_TS_CRIT_HIGH_EN In Band Error Interrupt Enable for Temperature Sensor Critical High 0 = Disable; Table 133, “MR51” [2] = ‘1’ does not generate an IBI to Host 1 = Enable; Table 133, “MR51” [2] = ‘1’ and Table 121, “MR27” [4] = ‘1’ generate an IBI to Host |
| 1 | RW | 0 | MR27[1]: IBI_TS_LOW_EN In Band Error Interrupt Enable for Temperature Sensor Low 0 = Disable; Table 133, “MR51” [1] = ‘1’ does not generate an IBI to Host 1 = Enable; Table 133, “MR51” [1] = ‘1’ and Table 121, “MR27” [4] = ‘1’ generate an IBI to Host |
| 0 | RW | 0 | MR27[0]: IBI_TS_HIGH_EN In Band Error Interrupt Enable for Temperature Sensor High 0 = Disable; Table 133, “MR51” [0] = ‘1’ does not generate an IBI to Host 1 = Enable; Table 133, “MR51” [0] = ‘1’ and Table 121, “MR27” [4] = ‘1’ generate an IBI to Host |
| NOTE 1 This register is a self clearing register after corresponding registers are cleared. Writing ‘0’ in this register has no effect. | | | |
| NOTE 2 After this command is issued, the device does not generate an IBI for any pending event. But if new event occurs, the device does generate an IBI. | | | |
| NOTE 3 This register is automatically updated when ENEC CCC or DISEC CCC or RSTDAA CCC is registered by the SPD5 Hub device or when SPD5 Hub device goes through bus reset as described in clause 2.6.21. This register can be read by the Host through normal read operation but cannot be written with normal write operation either in I ² C mode or I3C Basic mode. When this register is updated, it takes effect when there is a next START operation (i.e., after STOP operation). | | | |

3.2.3 Register Description (cont'd)

Table 122 — MR28

| Addr | MR28 | | Thermal Sensor High Limit Configuration - Low Byte ^{1,2,3} |
|---|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | RW | 0x70 | MR28[7:0]: TS_HIGH_LIMIT_LOW Table 122, “MR28” and Table 123, “MR29” - 16 bit thermal registers define the high limit for thermal sensor. See Table 103, “Thermal Register - Low Byte and High Byte”. |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 Critical temperature High Limit value must have a higher value than temperature High Limit (Table 122, “MR28” [7:0] and Table 123, “MR29” [7:0]). | | | |
| NOTE 3 The Reserved bits are Read Only bits. The host must write ‘0’ in reserved bits when writing and device always returns ‘0’ from reserved bits when reads this byte. | | | |

Table 123 — MR29

| Addr | MR29 | | Thermal Sensor High Limit Configuration - High Byte ^{1,2,3} |
|---|------|---------|--|
| Bits | Attr | Default | Description |
| 7:0 | RW | 0x03 | MR29[7:0]: TS_HIGH_LIMIT_HIGH Table 122, “MR28” and Table 123, “MR29” - 16 bit thermal registers define the high limit for thermal sensor. See Table 103, “Thermal Register - Low Byte and High Byte”. |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 Critical temperature High Limit value must have a higher value than temperature High Limit (Table 122, “MR28” [7:0] and Table 123, “MR29” [7:0]). | | | |
| NOTE 3 The Reserved bits are Read Only bits. The host must write ‘0’ in reserved bits when writing and device always returns ‘0’ from reserved bits when reads this byte. | | | |

Table 124 — MR30

| Addr | MR30 | | Thermal Sensor Low Limit Configuration - Low Byte ^{1,2,3} |
|---|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | RW | 0 | MR30[7:0]: TS_LOW_LIMIT_LOW Table 124, “MR30” and Table 125, “MR31” - 16 bit thermal registers define the low limit for thermal sensor. See Table 103, “Thermal Register - Low Byte and High Byte”. |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 Critical temperature Low Limit value must have a lower value than temperature Low Limit (Table 124, “MR30” [7:0] and Table 125, “MR31” [7:0]). | | | |
| NOTE 3 The Reserved bits are Read Only bits. The host must write ‘0’ in reserved bits when writing and device always returns ‘0’ from reserved bits when reads this byte. | | | |

3.2.3 Register Description (cont'd)**Table 125 — MR31**

| Addr | MR31 | | Thermal Sensor Low Limit Configuration - High Byte ^{1,2,3} |
|---|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | RW | 0 | MR31[7:0]: TS_LOW_LIMIT_HIGH Table 124, “MR30” and Table 125, “MR31” - 16 bit thermal registers define the low limit for thermal sensor. See Table 103, “Thermal Register - Low Byte and High Byte” |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 Critical temperature Low Limit value must have a lower value than temperature Low Limit (Table 124, “MR30” [7:0] and Table 125, “MR31” [7:0]). | | | |
| NOTE 3 The Reserved bits are Read Only bits. The host must write ‘0’ in reserved bits when writing and device always returns ‘0’ from reserved bits when reads this byte. | | | |

Table 126 — MR32

| Addr | MR32 | | Thermal Sensor Critical Temperature High Limit Configuration - Low Byte ^{1,2,3} |
|---|------|---------|--|
| Bits | Attr | Default | Description |
| 7:0 | RW | 0x50 | MR32[7:0]: TS_CRIT_HIGH_LIMIT_LOW Table 126, “MR32” and Table 127, “MR33” - 16 bit thermal registers define the critical temperature high limit for thermal sensor. See Table 103, “Thermal Register - Low Byte and High Byte” |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 Critical temperature High Limit value must have a higher value than temperature High Limit (Table 122, “MR28” [7:0] and Table 123, “MR29” [7:0]). | | | |
| NOTE 3 The Reserved bits are Read Only bits. The host must write ‘0’ in reserved bits when writing and device always returns ‘0’ from reserved bits when reads this byte. | | | |

Table 127 — MR33

| Addr | MR33 | | Thermal Sensor Critical Temperature High Limit Configuration- High Byte ^{1,2,3} |
|---|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | RW | 0x05 | MR33[7:0]: TS_CRIT_HIGH_LIMIT_HIGH Table 126, “MR32” and Table 127, “MR33” - 16 bit thermal registers define the critical temperature high limit for thermal sensor. See Table 103, “Thermal Register - Low Byte and High Byte” |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 Critical temperature High Limit value must have a higher value than temperature High Limit (Table 122, “MR28” [7:0] and Table 123, “MR29” [7:0]). | | | |
| NOTE 3 The Reserved bits are Read Only bits. The host must write ‘0’ in reserved bits when writing and device always returns ‘0’ from reserved bits when reads this byte. | | | |

3.2.3 Register Description (cont'd)

Table 128 — MR34

| Addr | MR34 | | Thermal Sensor Critical Temperature Low Limit Configuration- Low Byte ^{1,2,3} |
|---|------|---------|--|
| Bits | Attr | Default | Description |
| 7:0 | RW | 0 | MR34[7:0]: TS_CRIT_LOW_LIMIT_LOW Table 128, “MR34” and Table 129, “MR35” - 16 bit thermal registers define the critical temperature low limit for thermal sensor. See Table 103, “Thermal Register - Low Byte and High Byte” |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 Critical temperature Low Limit value must have a lower value than temperature Low Limit (Table 124, “MR30” [7:0] and Table 125, “MR31” [7:0]). | | | |
| NOTE 3 The Reserved bits are Read Only bits. The host must write ‘0’ in reserved bits when writing and device always returns ‘0’ from reserved bits when reads this byte. | | | |

Table 129 — MR35

| Addr | MR35 | | Thermal Sensor Critical Temperature Low Limit Configuration- High Byte ^{1,2,3} |
|---|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | RW | 0 | MR35[7:0]: TS_CRIT_LOW_LIMIT_HIGH Table 128, “MR34” and Table 129, “MR35” - 16 bit thermal registers define the critical temperature low limit for thermal sensor. See Table 103, “Thermal Register - Low Byte and High Byte” |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 Critical temperature High Limit value must have a higher value than temperature High Limit (Table 124, “MR30” [7:0] and Table 125, “MR31” [7:0]). | | | |
| NOTE 3 The Reserved bits are Read Only bits. The host must write ‘0’ in reserved bits when writing and device always returns ‘0’ from reserved bits when reads this byte. | | | |

3.2.3 Register Description (cont'd)**Table 130 — MR48**

| Addr | MR48 | | Device Status |
|------|------|---------|--|
| Bits | Attr | Default | Description |
| 7 | RO | 0 | MR48[7]: IBI_STATUS Device Event In Band Interrupt Status 0 = No pending IBI 1 = Pending IBI |
| 6:4 | RV | 0 | MR48[6:4]: Reserved |
| 3 | RO | 0 | MR48[3]: WR_OP_STATUS Write Operation Status 0 = No internal write operation is on going; 1 = Internal write operation is on-going; Device ignores Host Write command if Host attempts to write when this bit is '1'. The device self clears this bit to '0' when its completes internal write operation |
| 2 | RO | - | MR48[2]: WP_OVERRIDE_STATUS Write Protect Override Status 0 = Override of write protect bits in Table 113, "MR12" and Table 114, "MR13" are blocked 1 = Override of write protect bits in Table 113, "MR12" and Table 114, "MR13" are allowed The default state of this register reflects the sensing of HSA pin during power on. This bit is set to '1' if HSA pin is directly tied to GND. This bit is set to '0' if HSA pin is connected to GND through a resistor. |
| 1:0 | RV | 0 | MR48[1:0]: Reserved |

3.2.3 Register Description (cont'd)

Table 131 — MR49

| Addr | MR49 | | Current Sensed Temperature - Low Byte ^{1,2} |
|--|------|---------|---|
| Bits | Attr | Default | Description |
| 7:0 | RO | 0 | MR49[7:0]: TS_SENSE_LOW Table 131, “MR49” and Table 132, “MR50” - 16 bit thermal registers return the most recent conversion of the thermal sensor See Table 103, “Thermal Register - Low Byte and High Byte” |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 The device always returns ‘0’ from reserved bits. | | | |

Table 132 — MR50

| Addr | MR50 | | Current Sensed Temperature - High Byte ^{1,2} |
|--|------|---------|--|
| Bits | Attr | Default | Description |
| 7:0 | RO | 0 | MR50[7:0]: TS_SENSE_HIGH Table 131, “MR49” and Table 132, “MR50” - 16 bit thermal registers return the most recent conversion of the thermal sensor See Table 103, “Thermal Register - Low Byte and High Byte” |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is 0x18. | | | |
| NOTE 2 The device always returns ‘0’ from reserved bits. | | | |

Table 133 — MR51

| Addr | MR51 | | Thermal Sensor Temperature Status ¹ |
|---|------|---------|--|
| Bits | Attr | Default | Description |
| 7:4 | RV | 0 | MR51[7:4]: Reserved |
| 3 | RO | 0 | MR51[3:]: TS_CRIT_LOW_STATUS Temperature Sensor Critical Low 0 = Temperature is above the limit set in Table 128, “MR34” and Table 129, “MR35” 1 = Temperature is below the limit set in Table 128, “MR34” and Table 129, “MR35” |
| 2 | RO | 0 | MR51[2]: TS_CRIT_HIGH_STATUS Temperature Sensor Critical High 0 = Temperature is below the limit set in Table 126, “MR32” and Table 127, “MR33” 1 = Temperature is above the limit set in Table 126, “MR32” and Table 127, “MR33” |
| 1 | RO | 0 | MR51[1]: TS_LOW_STATUS Temperature Sensor Low 0 = Temperature above limit set in registers Table 124, “MR30” and Table 125, “MR31” 1 = Temperature below limit set in registers Table 124, “MR30” and Table 125, “MR31” |
| 0 | RO | 0 | MR51[0]: TS_HIGH_STATUS Temperature Sensor High 0 = Temperature is below the limit set in registers Table 122, “MR28” and Table 123, “MR29” 1 = Temperature is above the limit set in registers Table 122, “MR28” and Table 123, “MR29” |
| NOTE 1 This entire register is only applicable if Table 106, “MR1” [7:0] programmed value is either 0x18. | | | |

3.2.3 Register Description (cont'd)

Table 134 — MR52

| Addr | MR52 | | Hub and Thermal Sensor Error Status |
|--|------|---------|---|
| Bits | Attr | Default | Description |
| 7 | RO | 0 | MR52[7]: BUSY_ERROR_STATUS Write or Read Attempt while SPD5 Hub Device was Busy (Write Recovery Time Violation) ^{1,2} 0 = No write or read attempt while SPD Hub device was busy 1 = Write or Read attempt while SPD5 Hub device was busy |
| 6 | RO | 0 | MR52[6]: WR_NVM_BLK_ERROR_STATUS Write Attempt to Protected NVM Block 0 = No write attempt 1 = Write attempt to protected NVM Block |
| 5 | RO | 0 | MR52[5]: WR_NVM_PRO_REG_ERROR_STATUS Write Attempt to NVM Protection Registers 0 = No attempt to modify write protect registers 1 = Write attempt to modify write protect registers |
| 4:2 | RV | 0 | MR52[4:2]: Reserved |
| 1 | RO | 0 | MR52[1]: PEC_ERROR_STATUS Packet Error ^{3,4} 0 = No PEC Error 1 = PEC Error in one or more packets |
| 0 | RO | 0 | MR52[0]: PAR_ERROR_STATUS Parity Check Error ^{4,5} 0 = No Parity Error 1 = Parity Error in one or more bytes |
| NOTE 1 SPD5 Hub device busy status is only for accessing EEPROM memory. For any access to volatile register space, this bit definition does not apply. | | | |
| NOTE 2 When SPD5 Hub device is busy with EEPROM write/read, it sends NACK to the host requests within write recovery time. | | | |
| NOTE 3 Only applicable Table 117, “MR18” [5] = ‘1’ and if PEC function is enabled. | | | |
| NOTE 4 This register is updated when SPD5 Hub device goes through bus reset as described in clause 2.6.21. | | | |
| NOTE 5 Only applicable in Table 117, “MR18” [5] = ‘1’ and if Parity function is not disabled or for supported CCC in I ² C mode. | | | |

4 Annex A — (Informative) Differences Between Revisions

4.1 Differences Between Version 1.1 and 1.0

1. Table 3 (7-bit Address of the Local Devices (e.g. PMIC Device): slight wording update for MR19 and MR20 description
2. Clause 2.6.21 (Bus Reset): Added Bus Clear definition.
3. Clauses 2.6.12.2 and 2.6.12.3: Added clarification.
4. Table 97 (DC Characteristics): Added test condition; Fixed typo in note 2 and test condition; added footnote 3 for example calculation; and updated VILmin value.
5. Table 92 (Operating Conditions): Added notes 1 and 2 for clarification; Updated VDDIO max specification.
6. Table 98 (AC Characteristics): Updated AC timing values for some parameters to align with other specification.
7. Section 2.6.18.8 (DEVCTRL CCC): Fixed typographical error.
8. Figure 9 (Controller Push Pull to Target Open Drain Hand Off Operation): Added Figure for Controller Push Pull to Target Open Drain Hand Off Operation
9. Table 55 (Read Command Data Packet; PEC Enabled): Updated footnote references and added clarification.
10. Table 96 (Output Ron Specification): Fixed typographical errors; updated min value.
11. Clause 2.2 (Device Power Up): Added clarification.
12. Clause 2.8 SPD5 Hub Device - Write and Read Access): Added clarification.
13. Clauses 2.6.8 (I2C Target Protocol - Host to SPD5 Hub Device) and clause 2.6.10.1 (Write Operation Data Packet): Added additional clarification.
14. Table 16 (Read Command Data Packet (e.g., PMIC): Added note for on die pullup resistor specification.
15. Table 25 (Read Command Data Packet; PEC Enabled), Table 26 (Read Command Data Packet with IBI Header; PEC Disabled), Table 29 (Read Command Data Packet with Read Address Pointer Mode; PEC Enabled), and Table 30 (Read CMD Data Packet with Read Address Pointer Mode and IBI Header; PEC Disabled): Corrected register cross reference.
16. Clause 2.6.18.8 (DEVCTRL CCC): Added clarification.

4.2 Differences Between Version 1.2 and 1.1

1. Table 96 (Output Ron Specification): Corrected Ron min value.
2. Table 97 (DC Characteristics): Fixed typographical error.
3. Table 56 (SPD5 Hub CCC Support Requirement): Deleted direct RSTDAA CCC.
4. Table 69 (SETAASA CCC - Broadcast) and Table 70 (GETSTATUS CCC - Direct): Deleted to remove direct RSTDAA CCC.

4.3 Differences Between Version 1.3 and 1.2

1. Table 96: Separated output slew rate value.
2. Table 92: Added write endurance value for 25C.
3. Changed all references from Master to Controller and from Slave to Target
4. Changed all references from tDOFFM/tDOFFS to tDOFFC/tDOFFT
5. Changed error states Mx/Sx to CEx/TE_x.
6. Eliminated SPD5108 without thermal sensor; adjusted Ron per SidebandBus change 21-033
7. Removed SPD5108 from package diagram. Updated MO-229 to reflect new variation name U/V/W2030D-3
8. Editorial changes: IBI, ACK clarifications, corrected local device interrupt diagrams with Slave references
9. Editorial change: IBI note moved to correct place

4.4 Differences Between Version 1.4 and 1.3

1. Added HSCL slope reversal item 600.02 ballot 21-228.
2. Remove empty column in slope reversal table.
3. Editorial: Changed title from SPD5118 Hub and Serial Presence Detect Device Specification to SPD5118 Hub And Serial Presence Detect Device *Standard*
4. Editorial: Reformatted to JEDEC standard
5. Editorial: Adjusted table and clause numbering
6. Editorial: Replaced every instance of “section” with “clause” and JEDEC “specification” with “standard”

4.5 Differences Between Version 1.5 and 1.4

1. Changed slope reversal diagrams (Figures 23 through 25)
2. Corrected -40C example in Table 104 “Thermal Register Examples”
3. Clarification on Table 90 “HSCL Slope Reversal Parameters”
4. Changed MIN/MAX values of I_{OL} and I_{OH} in Table 97 “DC Characteristics”



Standard Improvement Form**JEDEC Standard JESD300-5B**

The purpose of this form is to provide the Technical Committees of JEDEC with input from the industry regarding usage of the subject standard. Individuals or companies are invited to submit comments to JEDEC. All comments will be collected and dispersed to the appropriate committee(s).

If you can provide input, please complete this form and return to:

JEDEC
Attn: Publications Department
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

Fax: 703.907.7583

1. I recommend changes to the following:

☐ Requirement, clause number _____

☐ Test method number _____ Clause number _____

The referenced clause number has proven to be:

☐ Unclear ☐ Too Rigid ☐ In Error

☐ Other _____

2. Recommendations for correction:

3. Other suggestions for document improvement:

Submitted by

Name: _____

Phone: _____

Company: _____

E-mail: _____

Address: _____

City/State/Zip: _____

Date: _____

JEDEC[®]

The JEDEC logo is centered on the page. It features the word "JEDEC" in a bold, italicized, dark brown sans-serif font. A registered trademark symbol (®) is located at the end of the word. Below the text is a thick, dark red horizontal line that starts under the 'J' and extends to the right, ending under the 'C'.