UPD350

Highly Integrated Small Form Factor USB Type-C® Power Delivery 3.0 Port Controller

Highlights

- Companion PD Controller for Microchip USB Hub / SoC
- Small Form Factor QFN Package
- Integrated Analog Discrete Components Reduce Bill of Materials and Design Footprint
- USB Power Delivery 3.0 Compliant MAC
- USB Type-C (1) Connector Support with Connection Detection and Control
- I^2C/SPI (2) Interface for CPU/SoC Communication
- USB Type-C® Alternate Mode Support
- Dual Role Port (DRP) and Fast Role Swap (FRS) Support with DRP offload mode

Target Applications

- Notebook Computers
- All-in-One/Desktop PCs
- Smartphones
- Tablets
- Monitors
- Docking Stations
- HDTVs
- Printers
- Automotive Breakout Boxes
- Multi-port Chargers

Key Benefits

- Integrated Analog Discrete Components
  - VCONN FETs with Rp/Rd Switching
  - Dead Battery Rd termination
  - Programmable Current Sense for Overcurrent Conditions
  - Voltage Sense for Overvoltage Conditions
- Integrated 3.3V Power Switch
  - Provides Dead Battery Support (2)
  - Automatically Switch between VBUS and Main +3.3V
- USB Power Delivery MAC
  - Compliant with USB Power Delivery Specification Revision 3.0
  - Power Delivery Packet Framing
  - CRC Checking/Generation
  - 4B/5B Encoding/Decoding
  - BMC Encoding/Decoding
  - EOP/SOP Generation for PD Frames
  - SOP Detection and SOP Header Processing
  - Separate RX/TX FIFOs
  - Automatic GoodCRC Message Generation
  - Automatic Retry Generation
  - Error Handling
  - Low Standby Power Support
- USB Type-C Cable Detect Logic
  - Auto Cable Attach & Orientation Detection
  - Routes Baseband Communication to Respective CC Pin per Detected Orientation
  - VCONN Supply Control for Active Cable
  - Configurable Downstream Facing Port (DFP) and Upstream Facing Port (UFP) Modes
  - Charging Current Capability Detection
  - Detection of Debug Accessory Mode, Audio Adapter Accessory Mode
- I^2C/SPI Interface Supports Communication/Configuration via Microchip USB Power Delivery hub or supported embedded controller (2)
- Alternate Mode Support
  - DisplayPort™ and other Major Protocols
- **CFG_SEL** Pin for Selection of Device Mode and I^2C addresses (2)
- Power and I/Os
  - Integrated 1.8V Voltage Regulator
  - 10 Configurable General Purpose I/O Pins
- Package
  - 28-QFN (4.0mm x 4.0mm)
- Environmental Product Options
  - Commercial (0°C to +70°C)
  - Industrial (-40°C to +85°C)
  - Automotive AEC-Q100 Grade 3 (-40°C to +85°C)

1. *USB Type-C® and USB-C® are trademarks of USB Implementers Forum.*
2. *Available only in select UPD350 configurations.*

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com**. We welcome your feedback.

Most Current Documentation

To obtain the most up-to-date version of this documentation, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; http://www.microchip.com
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include -literature number) you are using.

Customer Notification System

Register on our web site at **www.microchip.com** to receive the most current information on all of our products.

Table of Contents

1.0 PREFACE

1.1 Glossary of Terms

TABLE 1-1: GLOSSARY OF TERMS

Term	Definition
ADC	Analog to Digital Converter
AFE	Analog Front End
BCI	Baseband CC Interface
Billboard	USB Billboard Device. A required USB device class for UFPs which support Alternate Modes in order to provide product information to the USB Host.
BIST	Built-In Self Test
BMC	Bi-phase Mark Coding
Byte	8-bits
CC	Generic reference to USB Type-C® Cable / Connector CC1/CC2 pins
CSR	Control and Status Register
DB	Dead Battery
DFP	Downstream Facing Port (USB Type-C® Specification definition)
DP	DisplayPort (a VESA standard interface)
DPM	Device Policy Manager (PD Specification definition)
DRP	Dual Role Power (USB Type-C® Specification definition)
DWORD	32-bits
EC	Embedded Controller
EP	USB Endpoint
FIFO	First In First Out buffer
FW	Firmware
FS	Full-Speed
Host	External system (Includes processor, application software, etc.)
HPD	Hot-Plug Detect functionality as defined by DisplayPort and DisplayPort Alternate Mode specifications
HS	High-Speed
HW	Hardware (Refers to function implemented by the device)
IC	Integrated Circuit
IFC	InterFrame Gap
LDO	Linear Drop-Out regulator
MAC	Media Access Controller
Microchip	Microchip Technology Incorporated
N/A	Not Applicable
OCS	Over-Current Sense
PCS	Physical Coding Sublayer
PD / UPD	USB Power Delivery
PIO	General Purpose I/O
PMIC	Power Management Integrated Circuit
POR	Power-On Reset
PRBS	Pseudo Random Binary Sequence
QWORD	64-bits
SA	Source Address

TABLE 1-1: GLOSSARY OF TERMS (CONTINUED)

Term	Definition
SBU	SideBand Use
SCSR	System Control and Status Register
SPM	System Policy Manager (PD Specification definition)
SS	SuperSpeed
SVDM	Standard/Vendor Defined Message (PD Specification definition)
SVID	Standard/Vendor IDentity (PD Specification definition)
TCPC	USB Type-C Port Controller
UFP	Upstream Facing Port (USB Type-C® Specification definition)
USB	Universal Serial Bus
USB Type-C	USB Type-C Cable / Connector
VDO	Vendor-defined Object (PD Specification definition)
VSM	Vendor Specific Messaging
WORD	16-bits
ZLP	Zero Length USB Packet

1.2 Buffer Types

TABLE 1-2: BUFFER TYPES

Buffer Type	Description
IS	Schmitt-triggered input
I2C	I^2C interface
O8	Output with 8 mA sink and 8 mA source
OD8	Open-drain output with 8 mA sink
PU	70k (typical) internal pull-up. Unless otherwise noted in the pin description, internal pull-ups are always enabled. **Note:** Internal pull-up resistors prevent unconnected inputs from floating. Do not rely on internal resistors to drive signals external to the device. When connected to a load that must be pulled high, an external resistor must be added.
AIO	Analog bidirectional
P	Power pin

Note: Digital signals are not 5V tolerant unless specified.

Note: Refer to Section 15.5, "DC Characteristics," on page 200 for the electrical characteristics of the various buffers.

1.3 Register Nomenclature

TABLE 1-3: REGISTER NOMENCLATURE

Register Bit Type Notation	Register Bit Description
R	**Read:** A register or bit with this attribute can be read.
W	**Write:** A register or bit with this attribute can be written.
RO	**Read only:** Read only. Writes have no effect.
RS	**Read to Set:** This bit is set on read.
WO	**Write only:** If a register or bit is write-only, reads will return unspecified data.
W1S	**Write One to Set:** Writing a one sets the value. Writing a zero has no effect.
W1C	**Write One to Clear:** Writing a one clears the value. Writing a zero has no effect.
WC	**Write Anything to Clear:** Writing anything clears the value.
LL	**Latch Low:** Clear on read of register.
LH	**Latch High:** Clear on read of register.
SC	**Self-Clearing:** Contents are self-cleared after the being set. Writes of zero have no effect. Contents can be read.
RO/LH	**Read Only, Latch High:** Bits with this attribute will stay high until the bit is read. After it is read, the bit will remain high, but will change to low if the condition that caused the bit to go high is removed. If the bit has not been read, the bit will remain high regardless of a change to the high condition.
NASR	**Not Affected by Software Reset.** The state of NASR bits do not change on assertion of a software reset.
RESERVED	**Reserved Field:** Reserved fields must be written with zeros, unless otherwise indicated, to ensure future compatibility. The value of reserved bits is not guaranteed on a read.

1.4 References

- NXP I^2C-Bus Specification (UM10204, April 4, 2014): www.nxp.com/documents/user_manual/UM10204.pdf
- USB Power Delivery and USB Type-C® Specifications: http://www.usb.org/developers/docs/usb_31_102015.zip
- VESA DisplayPort Alternate Mode Specification 1.0: http://www.vesa.org

2.0 INTRODUCTION

2.1 General Description

The UPD350 is a highly integrated, small form factor USB Type-C® Power Delivery (PD) Port Controller designed to adhere to the *USB Type-C® Cable and Connector Specification* and *USB Power Delivery 3.0 Specification*. The UPD350 provides cable plug orientation and detection for a USB Type-C receptacle and implements baseband communication with a partner USB Type-C device via the integrated USB Power Delivery 3.0 MAC. The UPD350 is designed to function as a Companion Power Delivery Controller to an external Microchip MCU/SoC or USB hub using the integrated I^2C/SPI interface. The device is capable of controlling up to 100W of Power Delivery current and voltage using an external power device. Alternatively, the UPD350 can operate as a standalone UFP basic Type-C (non-PD) controller.The device can function in Standalone UFP modes, or utilize the integrated I^2C/SPI interface to connect to a companion CPU/SoC (dependent on device version, see Section 2.2, "UPD350 Family Differences Summary").

The UPD350 integrates many of the analog discrete components required for USB Type-C PD applications, including two VCONN FETs with Rp/Rd switching, a power switch, and current and voltage sense circuitry for over-voltage/current detection. By integrating many of the analog discrete components required for USB Type-C PD applications, the UPD350 provides a low cost, low power, small footprint solution for consumer (notebooks, desktop PCs, smartphones, tablets, monitors, docking stations), industrial, and automotive applications.

To enable the UPD350 to efficiently support dead battery use cases, an integrated power switch is provided to select between two external 3.3V supplies (VBUS and main). This effectively allows connection detection and system wakeup without external processor intervention (external processor in sleep mode).

The UPD350 is capable of negotiating alternate modes over USB Type-C connectors using the Power Delivery 3.0 protocol. DisplayPort operation over USB Type-C connectors is supported in addition to other major protocols.

A system diagram utilizing the UPD350 is shown in Figure 2-1. An internal block diagram of the UPD350 is shown in Figure 2-2.

FIGURE 2-1: SYSTEM BLOCK DIAGRAM

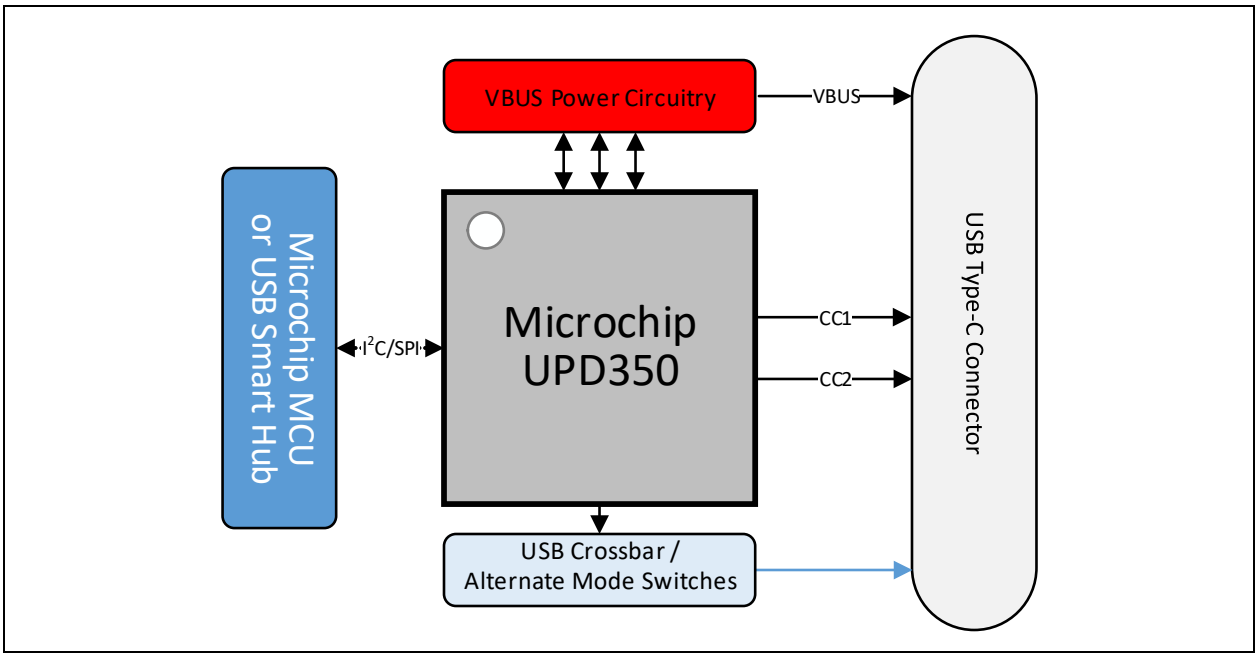
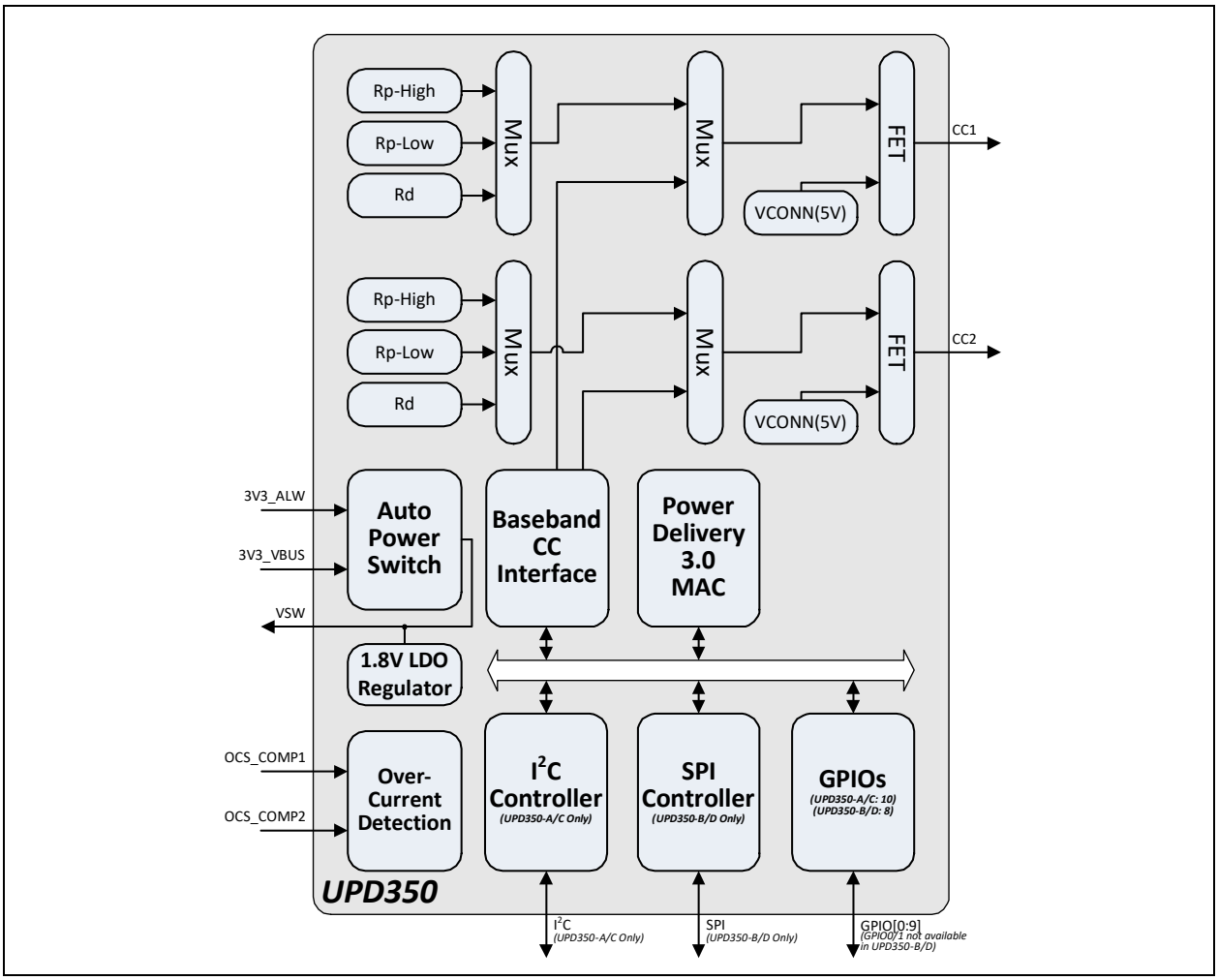


FIGURE 2-2: INTERNAL BLOCK DIAGRAM



2.2 UPD350 Family Differences Summary

The UPD350 is available in four versions:

- UPD350-A
- UPD350-B
- UPD350-C
- UPD350-D

A summary of the differences between these versions is provided in Table 2-1. Device specific features that do no pertain to the entire UPD350 family are called out independently throughout this document. For ordering information, refer to the Product Identification System on page 210.

TABLE 2-1: UPD350 FAMILY DIFFERENCES

Device	+1.8V-3.3V I²C Interface	SPI Interface	Standalone UFP/DFP Mode	Dead Battery Support
UPD350-A	X		X	X
UPD350-B		X		X
UPD350-C	X		X	
UPD350-D		X		

3.0 PIN DESCRIPTIONS AND CONFIGURATION

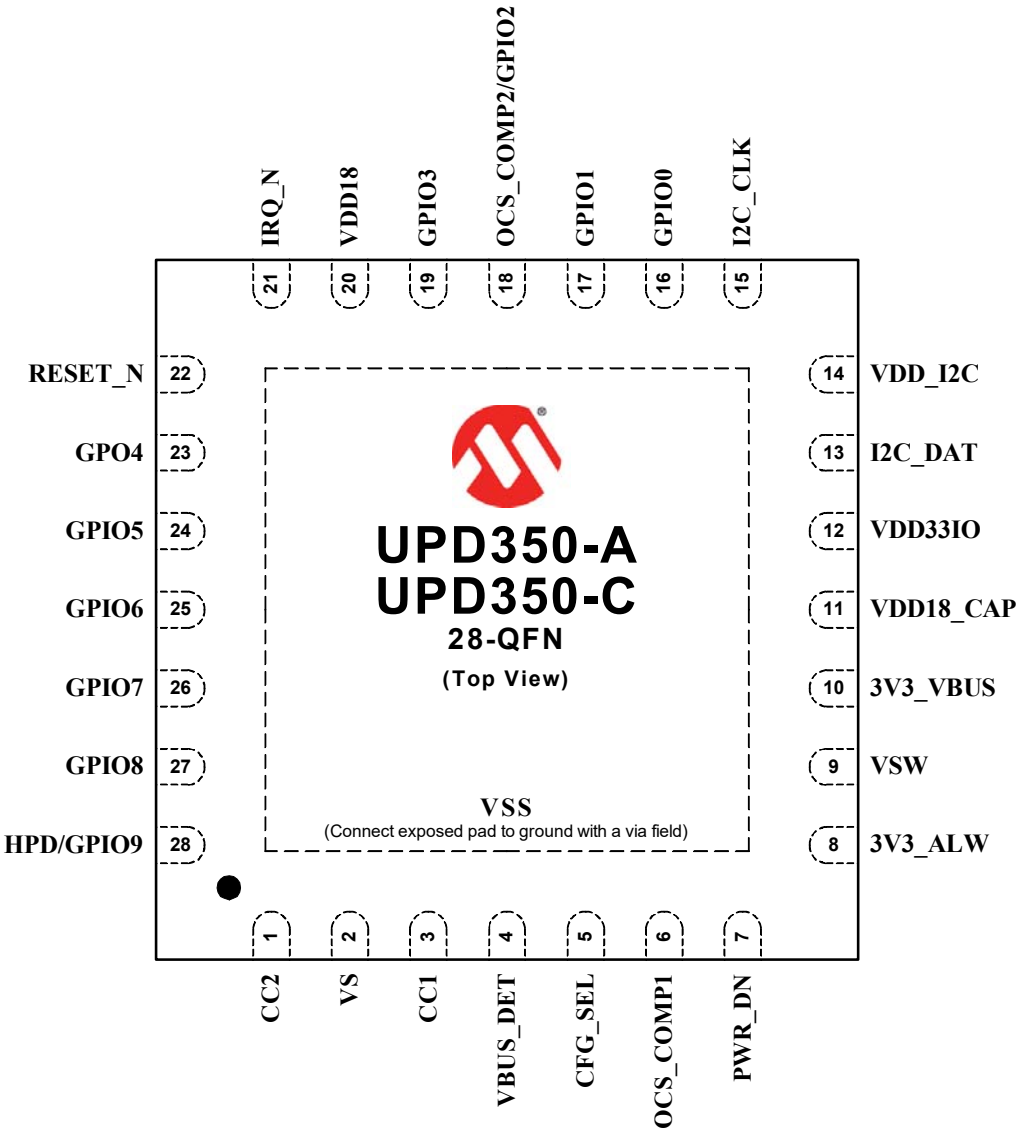
3.1 Pin Assignments

The pin assignments for the UPD350-A / UPD350-C are detailed in Section 3.1.1, "UPD350-A / UPD350-C Pin Assignments," on page 9. The pin assignments for the UPD350-B / UPD350-D are detailed in Section 3.1.2, "UPD350-B / UPD350-D Pin Assignments," on page 11. For information on the differences between the UPD350 family of devices, refer to Section 2.2, "UPD350 Family Differences Summary," on page 8.

3.1.1 UPD350-A / UPD350-C PIN ASSIGNMENTS

The pin assignments of the UPD350-A and UPD350-C devices are identical. The device pin diagram for the UPD350-A / UPD350-C can be seen in Figure 3-1. Table 3-1 provides a UPD350-A / UPD350-C pin assignment table. Pin descriptions are provided in Section 3.2, "Pin Descriptions".

FIGURE 3-1: UPD350-A / UPD350-C PIN ASSIGNMENTS (TOP VIEW)



Note: Exposed pad (VSS) on bottom of package must be connected to ground with a via field.

TABLE 3-1: UPD350-A / UPD350-C PIN ASSIGNMENTS

Pin	Pin Name	Pin	Pin Name
1	CC2	15	I2C_CLK
2	VS	16	GPIO0(Note 3-1)
3	CC1	17	GPIO1(Note 3-1)
4	VBUS_DET	18	OCS_COMP2/GPIO2(Note 3-1)
5	CFG_SEL	19	GPIO3(Note 3-1)
6	OCS_COMP1	20	VDD18
7	PWR_DN	21	IRQ_N
8	3V3_ALW	22	RESET_N
9	VSW	23	GPO4
10	3V3_VBUS	24	GPIO5(Note 3-1)
11	VDD18_CAP	25	GPIO6(Note 3-1)
12	VDD33IO	26	GPIO7(Note 3-1)
13	I2C_DAT	27	GPIO8(Note 3-1)
14	VDD_I2C	28	HPD/GPIO9(Note 3-1)

Note 3-1 This pin provides alternate functions when in Standalone DFP and/or Standalone UFP Mode. Refer to Section 3.1.1.1, "UPD350-A / UPD350-C GPIO Functions in Standalone DFP/UFP Modes" for additional information.

3.1.1.1 UPD350-A / UPD350-C GPIO Functions in Standalone DFP/UFP Modes

When the UPD350-A / UPD350-C is configured in Standalone DFP or Standalone UFP mode, the following GPIO pins are assigned specific alternate functions, as detailed in Table 3-2.

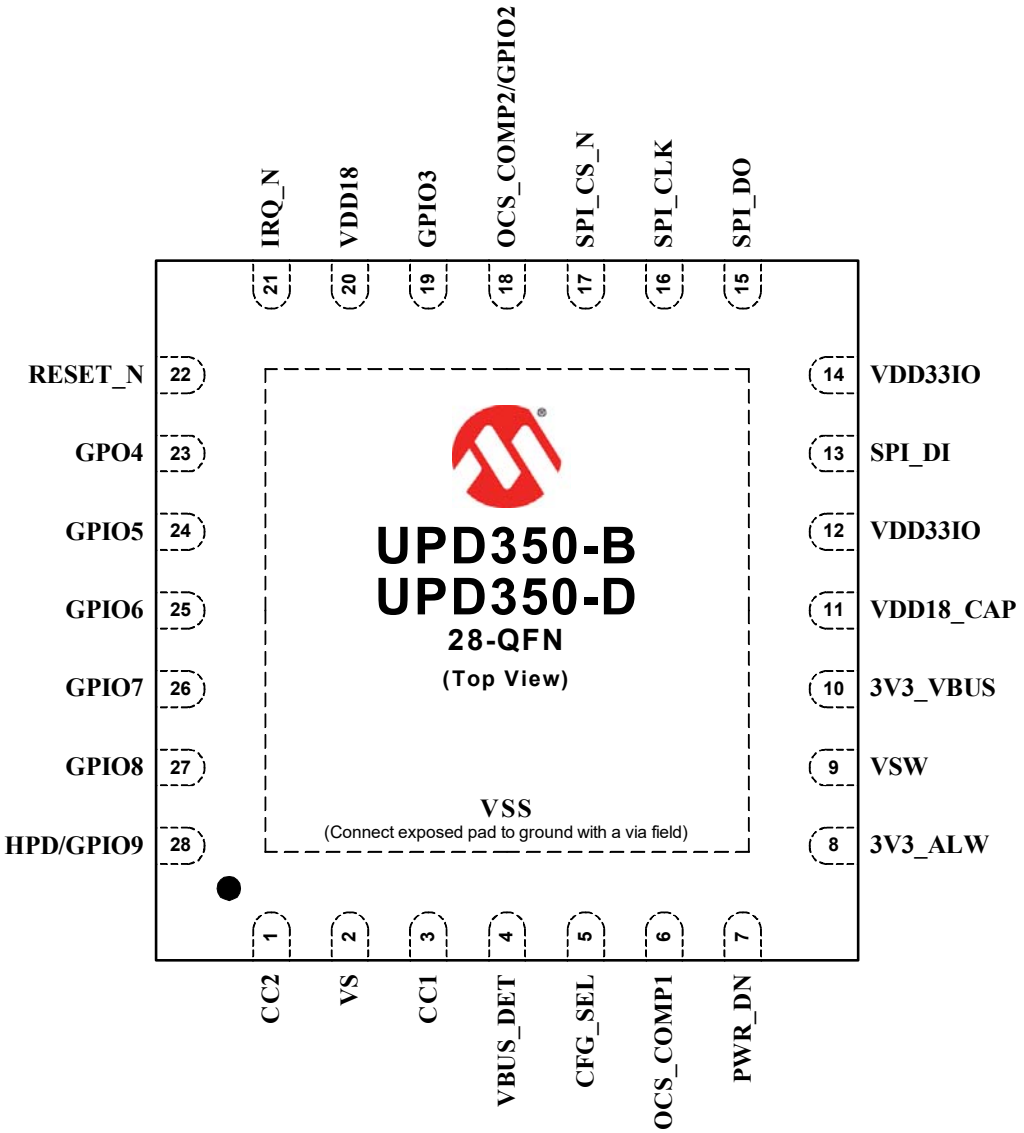
TABLE 3-2: UPD350-A / UPD350-C ALTERNATE GPIO FUNCTIONS IN STANDALONE DFP/UFP MODE

Pin	I^2C Companion Mode	Standalone DFP Mode	Standalone UFP Mode
16	GPIO0	ERR_RECOVER	GPIO0
17	GPIO1	PWR_EN	GPIO1
18	GPIO2	ORIENTATION	ORIENTATION
19	GPIO3	ATTACH	ATTACH
23	GPO4	DISCHARGE	GPO4
24	GPIO5	PORT_POWER	GPIO5
25	GPIO6	OCS_IN	SINK_5V_LEGACY_N
26	GPIO7	OCS_OUT	SINK_5V_1A5_N
27	GPIO8	PWR_CAP0	SINK_5V_3A0_N
28	GPIO9	PWR_CAP1	GPIO9

3.1.2 UPD350-B / UPD350-D PIN ASSIGNMENTS

The pin assignments of the UPD350-B and UPD350-D devices are identical. The device pin diagram for the UPD350-B / UPD350-D can be seen in Figure 3-2. Table 3-3 provides a UPD350-B / UPD350-D pin assignment table. Pin descriptions are provided in Section 3.2, "Pin Descriptions".

FIGURE 3-2: UPD350-B / UPD350-D PIN ASSIGNMENTS (TOP VIEW)



Note: Exposed pad **(VSS)** on bottom of package must be connected to ground with a via field.

TABLE 3-3: UPD350-B / UPD350-D PIN ASSIGNMENTS

Pin	Pin Name	Pin	Pin Name
1	CC2	15	SPI_DO
2	VS	16	SPI_CLK
3	CC1	17	SPI_CS_N
4	VBUS_DET	18	OCS_COMP2/GPIO2
5	CFG_SEL	19	GPIO3
6	OCS_COMP1	20	VDD18
7	PWR_DN	21	IRQ_N
8	3V3_ALW	22	RESET_N
9	VSW	23	GPO4
10	3V3_VBUS	24	GPIO5
11	VDD18_CAP	25	GPIO6
12	VDD33IO	26	GPIO7
13	SPI_DI	27	GPIO8
14	VDD33IO	28	HPD/GPIO9

3.2 Pin Descriptions

This sections details the functions of the various device signals.

TABLE 3-4: PIN DESCRIPTIONS

Name	Symbol	Buffer Type	Description
USB Type-C®			
Configuration Channel 1	**CC1**	AIO	Configuration Channel (CC) used in the discovery, configuration and management of connections across a USB Type-C cable. Refer to Section “CC Comparator” for additional information.
Configuration Channel 2	**CC2**	AIO	Configuration Channel (CC) used in the discovery, configuration and management of connections across a USB Type-C cable. Refer to Section “CC Comparator” for additional information.
I^2C Interface *(UPD350-A / UPD350-C Only)*			
I^2C Clock	**I2C_CLK**	I2C	+1.8/3.3V I^2C clock signal
I^2C Data	**I2C_DAT**	I2C	+1.8/3.3V I^2C data signal
SPI Interface *(UPD350-B / UPD350-D Only)*			
SPI Clock	**SPI_CLK**	IS	SPI clock. The maximum supported SPI clock frequency is 25 MHz.
SPI Data Out	**SPI_DO**	O8	SPI output data.
SPI Data In	**SPI_DI**	IS	SPI input data.
SPI Chip Enable	**SPI_CS_N**	IS	Active low SPI chip enable input.
Power Delivery Control			
Hot Plug Detect	**HPD**	IS/O8	DisplayPort Hot Plug Detection. Refer to Section “DisplayPort Hot Plug Detect (HPD) for additional information.
VBUS Discharge	**DISCHARGE**	O8	VBUS discharge. **Note:** This signal is not available in the UPD350-B / UPD350-D.
Type-C Attach	**ATTACH**	O8	In the Standalone UFP and Standalone DFP mode *(UPD350-A / UPD350-C only)*, this signal indicates that the USB Type-C receptacles at the near and far end of the cable both have a plug-in. This pin is autonomously driven by the device in DFP standalone mode. 0b: Nothing attached 1b: USB Type-C port has an end-end attached **Note:** Float this signal when unused. **Note:** This signal is not available in the UPD350-B / UPD350-D.

TABLE 3-4: PIN DESCRIPTIONS (CONTINUED)

Name	Symbol	Buffer Type	Description
Type-C Orientation	**ORIENTATION**	O8	In the Standalone UFP and Standalone DFP mode *(UPD350-A / UPD350-C only)*, this signal is used to indicate which CC pin is terminated by the attached device. This pin is autonomously driven by the device in DFP standalone mode. DFP: 0b: **CC1** pin is terminated by Rd. 1b: **CC2** pin is terminated by Rd. UFP: 0b: **CC1** pin is pulled to a higher voltage than **CC2**. 1b: **CC2** pin is pulled to a higher voltage than **CC1**. **Note:** Float this signal when unused. **Note:** This signal is not available in the UPD350-B / UPD350-D.
VBUS Power Enable	**PWR_EN**	IS	In the Standalone DFP mode *(UPD350-A / UPD350-C only)*, this signal is used as a port power switch enable for USB hubs. **Note:** In UFP mode, this signal should be pulled-up when unused. **Note:** This signal is not available in the UPD350-B / UPD350-D.
Power Capability 0	**PWR_CAP0**	IS	In the Standalone DFP mode *(UPD350-A / UPD350-C only)*, these signals define the charging current supported by the device. 00b: USB 2.0 Default Current 01b: USB 3.0 Default Current 10b: 1.5 A 11b: 3.0 A **Note:** It is not valid to change the state of **PWR_CAP0** and **PWR_CAP1** pins after reset is de-asserted. **Note:** These signals are not available in the UPD350-B / UPD350-D.
Power Capability 1	**PWR_CAP1**	IS	
Over-current Sense Out	**OCS_OUT**	OD8	In Standalone DFP mode *(UPD350-A / UPD350-C only)*, this active-low output pin indicates an OCS event occurred to the partner hub. **Note:** It is recommended this signal be augmented with a weak pull-up at all times. **Note:** This signal is not available in the UPD350-B / UPD350-D.
Over-current Sense In	**OCS_IN**	IS	In Standalone DFP mode *(UPD350-A / UPD350-C only)*, this active-low input pin indicates an OCS event occurred at the partner hub. This signal maps to the PPC_INT interrupt. **Note:** It is recommended this signal be augmented with a weak pull-up at all times. **Note:** This signal is not available in the UPD350-B / UPD350-D.

TABLE 3-4: PIN DESCRIPTIONS (CONTINUED)

Name	Symbol	Buffer Type	Description
Port Power	**PORT_POWER**	O8	In Standalone DFP mode *(UPD350-A / UPD350-C only)*, this pin is used to enable the external Port Power Controller (PPC). **Note:** This signal is not available in the UPD350-B / UPD350-D.
Error Recovery	**ERR_RECOVER**	IS	In Standalone DFP mode *(UPD350-A / UPD350-C only)*, this pin determines whether or not the USB Type-C logic shall attempt to auto-recover from an OCS or other error. 0b: When operating as a standalone DFP, do not attempt to auto-recover from any OCS conditions. 1b: When operating as a standalone DFP, the Type-C FSM shall attempt to auto-recover from an OCS condition sources from either the external Port Power Switch or VCONN FET. **Note:** This signal is not available in the UPD350-B / UPD350-D.
Sink Legacy Current	**SINK_5V_LEGACY_N**	OD8	In the Standalone UFP mode *(UPD350-A / UPD350-C only)*, this pin asserts autonomously when a source has been detected that provides legacy USB current. **Note:** Float this signal when unused. **Note:** This signal is not available in the UPD350-B / UPD350-D.
Sink 1.5A Current	**SINK_5V_1A5_N**	OD8	In the Standalone UFP mode *(UPD350-A / UPD350-C only)*, this pin asserts autonomously when a source has been detected that provides 1.5A USB current. **Note:** Float this signal when unused. **Note:** This signal is not available in the UPD350-B / UPD350-D.
Sink 3A Current	**SINK_5V_3A0_N**	OD8	In the Standalone UFP mode *(UPD350-A / UPD350-C only)*, this pin asserts autonomously when a source has been detected that provides 3.0A USB current. **Note:** Float this signal when unused. **Note:** This signal is not available in the UPD350-B / UPD350-D.
Miscellaneous			
Interrupt	**IRQ_N**	OD8	Active low interrupt signal. **Note:** Float this signal when unused.
VBUS Detection	**VBUS_DET**	AIO	Scaled down version of VBUS. Tie this signal to VBUS via a resistor divider.
Configuration Select	**CFG_SEL**	AIO	This multi-level configuration signal is sampled after a system reset to select the device's default mode of operation based on the connected 1% precision resistor value. **Note:** This pin is used to determine the default I^2C slave address and operating mode in the UPD350-A / UPD350-C. For the UPD350-B / UPD350-D, this pin can be used for customer specific purposes to provide a discrete value (0-15) based upon the attached resistor value.

TABLE 3-4: PIN DESCRIPTIONS (CONTINUED)

Name	Symbol	Buffer Type	Description
General Purpose I/O 0-9	**GPIO0**, **GPIO1**, **GPIO2**, **GPIO3**, **GPO4**, **GPIO5**, **GPIO6**, **GPIO7**, **GPIO8**, **GPIO9**	IS/O8/ OD8 (PU)	The general purpose I/O signals are fully programmable as either a push-pull output, an open-drain output, or a Schmitt-triggered input (except **GPO4**). A programmable pull-up may optionally be enabled. **Note:** The functionality of these GPIOs is defined and controlled by USB Power Delivery firmware executed external to the UPD350 (in the Microchip USB hub or embedded controller). **Note:** **The GPO4 general purpose signal must be augmented with a weak pull-down in order to prevent the device from entering a test mode on start-up.** **Note:** In companion mode, these signals should be tied to ground when unused. **Note:** External pull-ups and pull-downs shall be placed on GPIO pins to ensure that when in the reset state the inputs to external devices are driven to a valid state. **Note:** **GPIO0** and **GPIO1** are not available in the UPD350-B / UPD350-D. **Note:** In Standalone DFP/UFP mode *(UPD350-A / UPD350-C only)*, select GPIOs have alternate dedicated functions, as defined in Section 3.1.1.1, "UPD350-A / UPD350-C GPIO Functions in Standalone DFP/UFP Modes," on page 10.
System Reset	**RESET_N**	IS	Active low system reset. **Note:** If this signal is unused, it must be pulled up to **VDD33IO**.
Power Down	**PWR_DN**	AI	When asserted, this signal places the device into the power-down state. **Note:** Tie this signal to ground when unused.
Over-Current Sense Comparator 1	**OCS_COMP1**	AI	This pin is used by the integrated OCS comparator to detect for error conditions. **Note:** Tie this signal to ground when unused.
Over-Current Sense Comparator 2	**OCS_COMP2**	AI	This pin is used by the integrated OCS comparator to detect for error conditions. **Note:** Tie this signal to ground when unused.
Power/Ground			
+3.3V Voltage Switch Supply	**VSW**	P	+3.3V power supply output from the integrated power switch. **Note:** This pin also provides capacitance for the integrated power switch and must be connected to a 1 uF (<100 Mohm ESR) capacitor to ground.
+3.3V VBUS Supply	**3V3_VBUS**	P	+3.3V main power supply input derived from VBUS to the integrated power switch. **Note:** The 2.2 uF capacitor is only required for the UPD350-A and UPD350-B.

TABLE 3-4: PIN DESCRIPTIONS (CONTINUED)

Name	Symbol	Buffer Type	Description
+3.3V Always Supply	**3V3_ALW**	P	+3.3V main power supply input to the integrated power switch. **Note:** This pin must be connect to a 2.2 uF capacitor to ground.
+3.3V I/O Power Supply Input	**VDD33IO**	P	+3.3V I/O power supply input.
+3.3/1.8V I^2C Power Supply Input	**VDD_I2C**	P	+3.3/1.8V I^2C power supply input. Tie this pin to **VDD33IO** for +3.3V I^2C interfaces. Tie this pin to **VDD18** for +1.8V I^2C interfaces. **Note:** This pin is not available in the UPD350-B / UPD350-D.
+1.8V Core Voltage Power Supply Input	**VDD18**	P	+1.8V core voltage power supply input.
+1.8V Digital Core Power Supply Capacitor	**VDD18_CAP**	P	+1.8V digital core power supply capacitor. This signal must be connected to a 1uF capacitor to ground for proper operation.
+5V VS Power Supply Input	**VS**	P	+5V VCONN FET power source.
Ground	**VSS**	P	Ground pins.

4.0 REGISTER MAP

This chapter provides the device register map, summarizing the various directly addressable System Control and Status Registers (CSRs). All CSRs are directly accessible via the device's internal I2C Slave Controller (UPD350-A/UPD350-C Only) or SPI Slave Controller (UPD350-B/UPD350-D Only). Detailed descriptions of the System CSRs are provided in the chapters corresponding to their function. Table 4-1 provides a summary of all directly addressable CSRs and their corresponding addresses.

Note: Register bit type definitions are provided in Section 1.3, "Register Nomenclature," on page 6.

TABLE 4-1: DEVICE REGISTER MAP

Address	Registers
3400h-FFFFh	RESERVED
3000h-33FFh	Watchdog Timer Registers
2C00h-2FFFh	RESERVED
2800h-2BFFh	Baseband CC Interface Registers
2000h-27FFh	RESERVED
1C00h-1FFFh	Power Switch Registers
1800h-1BFFh	Power Delivery MAC Registers
1400h-17FFh	RESERVED
1000h-13FFh	Clocks and Power Management Registers
0C00h-0FFFh	DisplayPort HPD Registers
0800h-0BFFh	Cable Orientation and Detection Registers
0400h-07FFh	RESERVED
0000h-03FFh	System Control Registers

Note: RESERVED address space must not be written under any circumstances. Failure to heed this warning may result in untoward operation and unexpected results.

5.0 I²C SLAVE CONTROLLER *(UPD350-A/UPD350-C ONLY)*

This chapter details the integrated I²C slave controller (**I2C_DAT** and **I2C_CLK**) available in the UPD350-A and UPD350-C. The I²C slave controller can be used for Host CPU serial management and data transfer, and allows host access to all device Configuration and Status Registers.

5.1 I²C Overview

I²C is a bi-directional 2-wire data protocol. A device that is currently sending data is defined as the “transmitter” and a device that is currently receiving data is defined as the “receiver”. The bus is controlled by a master which generates the SCL clock, controls bus access, and generates the start and stop conditions. The master and slave will operate as transmitter or receiver, bit-by-bit, as determined by the master. Since the device I²C controller is a slave only, the terms “host” and “master” are synonymous, both referring to the external side of the interface.

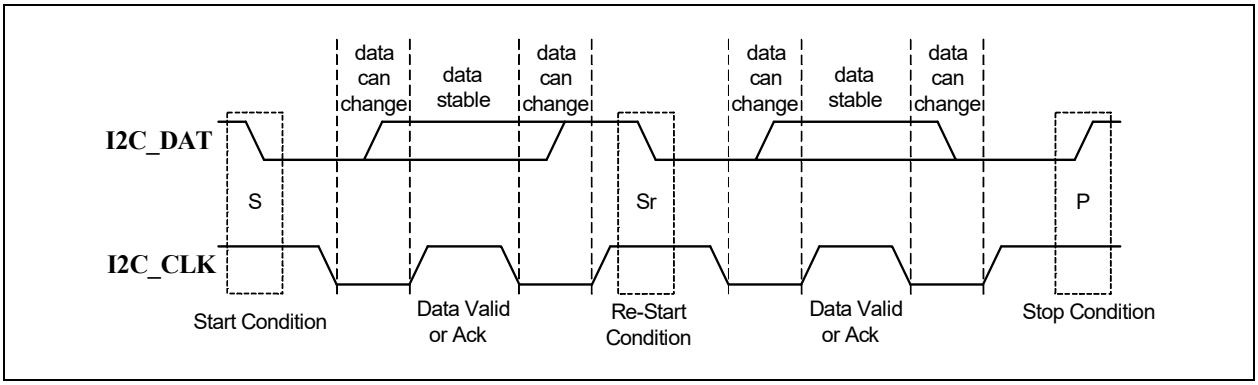
Both the clock (SCL) and data (SDA) signals have analog input filters that reject pulses that are less than 50 ns. The data pin is driven low when either interface sends a low, emulating the wired-AND function of the I²C bus. Since the slave interface never drives the clock pin, the wired-AND is not necessary.

The following bus states exist:

- **Idle:** Both **I2C_DAT** and **I2C_CLK** are high when the bus is idle.
- **Start & Stop Conditions:** A start condition (S) is defined as a high to low transition on the SDA line while SCL is high. A stop condition (P) is defined as a low to high transition on the SDA line while SCL is high. The bus is considered to be busy following a start condition and is considered free 4.7 µs / 1.3 µs / 0.5µs (for 100 kHz / 400 kHz / 1MHz operation, respectively) following a stop condition. The bus stays busy following a repeated start condition (Sr) in the absence of a stop condition. Stop/start sequences and repeated starts are otherwise functionally equivalent.
- **Data Valid:** Data is valid, following the start condition, when SDA is stable while SCL is high. Data can only be changed while the clock is low. There is one valid bit per clock pulse. Every byte must be 8 bits long and is transmitted MSB first.
- **Acknowledge:** Each byte of data is followed by an acknowledge bit. The master generates a ninth clock pulse for this bit, and the transmitter releases SDA (high). To provide a positive “acknowledge” (ACK), the receiver drives SDA low so that it remains valid during the high period of the clock, taking into account the setup and hold times. To provide a negative “no-acknowledge” (NACK or ACK), the receiver will allow the line to remain high during this bit time. The receiver may be the master or the slave depending on the direction of the data. Typically the receiver acknowledges each byte. If the master is the receiver, it does not generate an acknowledge on the last byte of a transfer. This informs the slave to not drive the next byte of data, freeing SDA so that the master may generate a stop or repeated start condition.

Figure 5-1 displays the various bus states of a typical I²C cycle.

FIGURE 5-1: I²C CYCLES



5.2 I²C Slave Operation

The I²C slave serial interface consists of a data wire (**I2C_DAT**) and a serial clock (**I2C_CLK**). The serial clock is driven by the master, while the data wire is bi-directional. Both signals are open-drain and require external pull-up resistors.

The I²C slave controller implements the low level I²C slave serial interface (start and stop condition detection, data bit transmission/reception and acknowledge generation/reception), handles the slave command protocol and performs system register reads and writes. It tolerates and also provides clock stretching, in particular for supporting a transparent Wake on Host Access (see Section 7.3, "Asynchronous I2C Wakeup (UPD350-A/UPD350-C Only)," on page 31).

The I²C slave controller conforms to the NXP *I²C-Bus Specification* (UM10204, April 4, 2014), and supports traffic as defined therein for the following modes:

- Standard-mode (Sm, 100 kbit/s)
- Fast-mode (Fm, 400 kbit/s)
- Fast-mode Plus (Fm+, 1 Mbit/s)

Refer to Section 15.6.2, "I2C Slave Interface (UPD350-A/UPD350-C only)," on page 202 for timing information.

5.2.1 I²C SLAVE COMMAND FORMAT

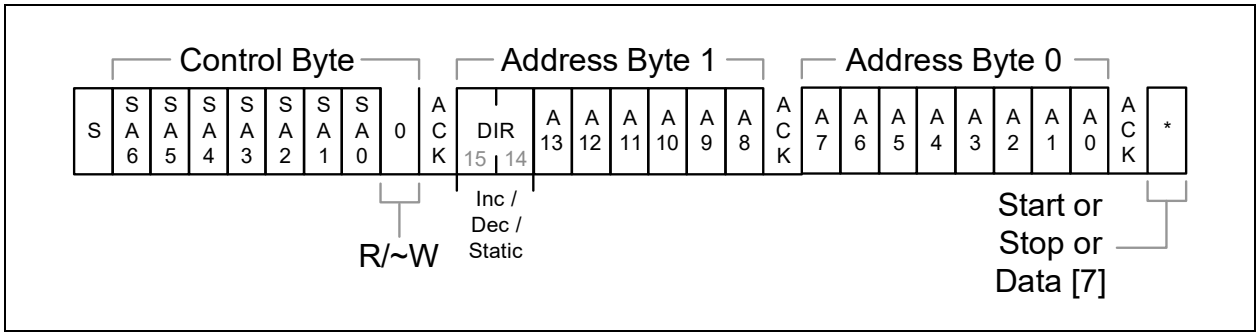
The I²C slave serial interface supports single register and multiple register Read and Write commands. A Read or Write command is started by the master first sending a Start condition, followed by a Control byte. The Control byte consists of a 7-bit slave address and a 1-bit Read/Write indication (R/~W). The default slave address used by the device is selected via the CFG_SEL configuration strap. Assuming the slave address in the Control byte matches this address, the Control byte is acknowledged by the device. Otherwise, the entire sequence is ignored until the next Start condition. The I²C slave controller also supports the General Call Address. The I²C command formats can be seen in Figure 5-2, Figure 5-4, and Figure 5-5.

If the read/write indication (R/~W) in the Control byte is a 0 (Write), the next two bytes sent by the master are a register address, and these two bytes are mandatory. The upper (first) two bits of the address field are a Direction control (DIR), which indicates whether multi-byte accesses will increment, decrement, or fix (as static) the issued address (Section 5.2.2). After the address bytes are acknowledged by the device, the master may send data bytes, which will be written to successive registers starting at this address. It may instead send another Start condition (to start the reading of data) or a Stop condition (only setting the address). The latter two will terminate the current Write before writing any data, but will have the effect of setting the internal register address which will be used for subsequent Reads.

If the read/write indication (R/~W) in the Control byte is a 1 (Read), the device will start sending data following the Control byte acknowledge bit. Read commands cannot designate an address by themselves, but may optionally be prefixed with a Write command to set it (see Figure 5-4, prefixes in gray). If however the Read immediately follows a Multiple Register Write or Read, the address may have been incremented or decremented internally according to its DIR field, so this Read will start its access at the next successive byte address. Also, regardless of the previous access, a multiple-byte Read will continue the Increment/Decrement internally, as determined by the previously-issued DIR field (Section 5.2.2).

The length of the register address field is always two full bytes. Some high-order bits are don't-care. Don't-care register address bits should be sent as '0' always, for upward compatibility.

FIGURE 5-2: I²C SLAVE ADDRESSING



Note:	Within bytes (address and data), the bits are transferred most-significant bit first. Addresses are transferred Most-Significant Byte first. All registers are accessed in units of bytes, and register data is transferred in increasing byte address order. Refer to the device register layout to determine the effect of this on the significance order of any multi-byte value.

5.2.2 MULTIPLE-BYTE REGISTER ADDRESS SEQUENCING

The DIR subfield in Address field bits [15:14] determines how multiple-byte sequences will be interpreted. This field is held internally whenever issued with an address, but is not applied in I^2C except in multiple-byte transfers, Read or Write. The DIR field definitions are as follows:

- **DIR = 00b:** Selects auto-incrementing of the internally-held register address for subsequent byte accesses in a multiple-byte packet.
- **DIR = 10b:** Selects auto-decrementing of the internally-held register address for subsequent byte accesses in a multiple-byte packet.
- **DIR = 11b:** Select a fixed address. No modification of the internal register address will occur, meaning that all subsequent accesses, single- or multiple-byte, are made to the same register.
- **DIR = 01b:** Reserved for future use.

Note that the DIR field is altered only by issuing an address. It remains, affecting any subsequent multiple-byte Read packets, until altered.

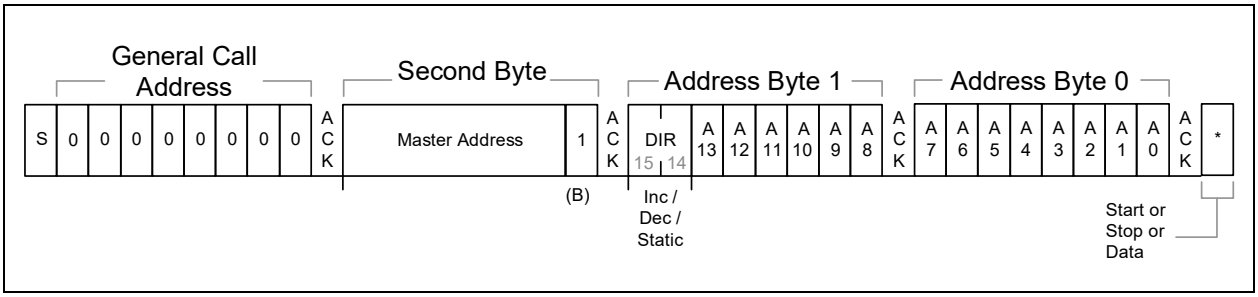
5.2.3 GENERAL CALL ADDRESS

The device supports the I^2C General Call Address. The intent of this feature is to enable global I^2C writes to topologies that have multiple UPD350 slaves. This minimizes the I^2C transactions for device reset, via the Hardware Control Register (HW_CTL),as well as for various common configuration registers. This mode of operation is intended for topologies that consist solely of UPD350 slaves. This mode of operation may not be compatible with non-UPD350 slaves coexisting on the I^2C bus.

Only the case where the least significant bit, “B”, of the General Call address is set to one is supported. The device will ignore the case when the least significant bit, “B”, of the General Call address is set to zero. For the latter case, the device will ACK the first byte, General call address. The device will ignore and silently discard all subsequent bytes and not acknowledge them. The second byte of the General Call address is also ignored and not acknowledged by the device.

Figure 5-3 illustrates the supported General Call Address format.

FIGURE 5-3: I^2C GENERAL CALL ADDRESS



5.2.4 DEVICE INITIALIZATION

Until the device has initialized itself to the point where the various configuration inputs are valid, the I^2C slave interface will not respond to or be affected by any external pin activity. The device should not be accessed by the master in this state. If, however, it is necessary to do so, this state will appear externally as a NACK (high) in the ACK bit time of the Control Byte, and of any further bytes transmitted by the master. The device will continue to act in this manner until the first Start condition is received after it is initialized internally. A Read transaction should not be attempted until an Address Write has been completed successfully (Figure 5-2), since the value(s) read may be unpredictable otherwise. Alternatively, an **IRQ_N** pin assertion can be used to indicate the device is ready.

5.2.5 ACCESS DURING AND FOLLOWING POWER MANAGEMENT

During low-power modes, a Start condition will trigger the device to wake, and the device will also stretch the I2C clock low until its internal clocks are running and locked. It will then release the I2C clock, and process the incoming packet.

It performs these steps before receiving the Slave Address bits, meaning that if there are multiple devices of this type asleep on the same I2C bus segment then they will all stretch the clock, and they will all wake, regardless of whether they were actually addressed. In the event that the slave address of the I2C transaction does not match the value specified in the I2C Slave Address Register (I2C_ADDR) (UPD350-A/UPD350-C Only), the device will power-down automatically.

5.2.6 I2C SLAVE READ SEQUENCE

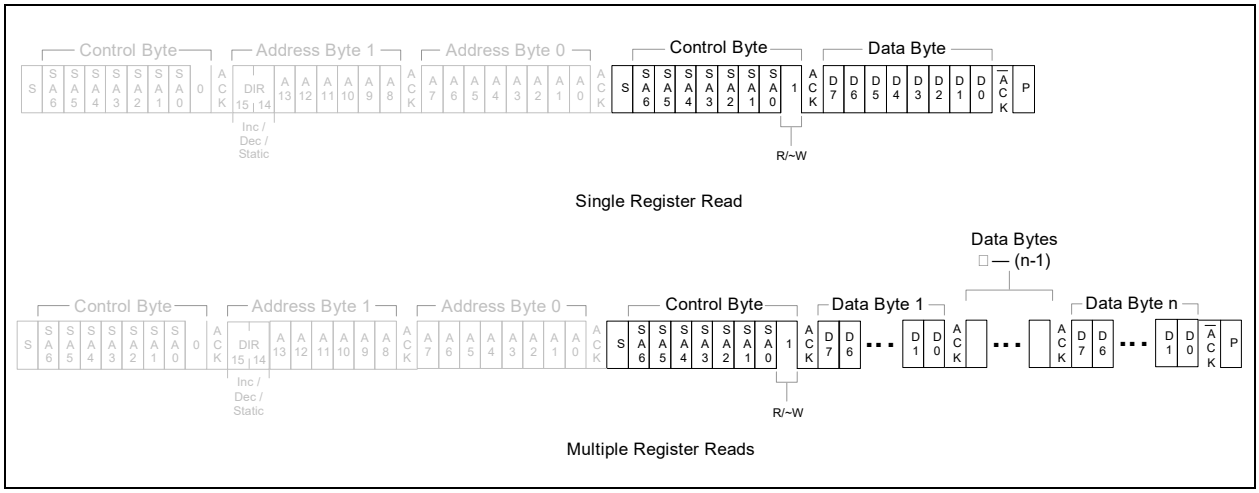
Following the device addressing, as detailed in Section 5.2.1, a register is read from the device when the master sends a Start condition and Control byte with the R/~W bit set to '1'. Assuming the slave address in the Control byte matches the device address, the Control byte is acknowledged by the device. Otherwise, the entire sequence is ignored until the next Start condition. Following the acknowledge, the device sends 1 or more bytes of data, from successive register addresses according to the last-issued DIR address subfield (Section 5.2.2), until the master sends a no-acknowledge followed by the Stop condition. The no-acknowledge informs the device not to send any further bytes.

The internal register address is unchanged if only a single register byte is read, otherwise (a Multiple Register Read) the internal register address may be incremented or decremented (Section 5.2.2) after each byte including the final one. If the internal address reaches its maximum, it rolls over to 0. See the register map for the location of this boundary.

If the master sends an unexpected start or stop condition, the device will stop sending immediately and will respond to the next sequence as needed.

Figure 5-4 illustrates a typical single and multiple register read. An optional Write of an address is allowed to occur first, shown in gray. Note that this example shows an abbreviated case, where the Write does not have a Stop condition before the Read transfer's Starts. in this case, the Stop is still allowed, but not required.

FIGURE 5-4: I2C SLAVE READS



5.2.7 I2C SLAVE WRITE SEQUENCE

Following the device addressing, as detailed in Section 5.2.1, a register value is written to the device when the master continues to send data bytes. Each byte is acknowledged by the device. Following any data byte, after the acknowledge, the master may either send another start condition or halt the sequence with a stop condition. The internal register address is unchanged following a single-byte write.

Multiple writes are performed when the master sends additional data bytes following the first. The internal address is automatically incremented and the next register is written. Once the internal address reaches its maximum value, it rolls over to 0. See the device register map for this boundary. The multiple write is concluded when the master sends another start or stop condition. In performing a multiple write, the internal register address may be incremented or decremented

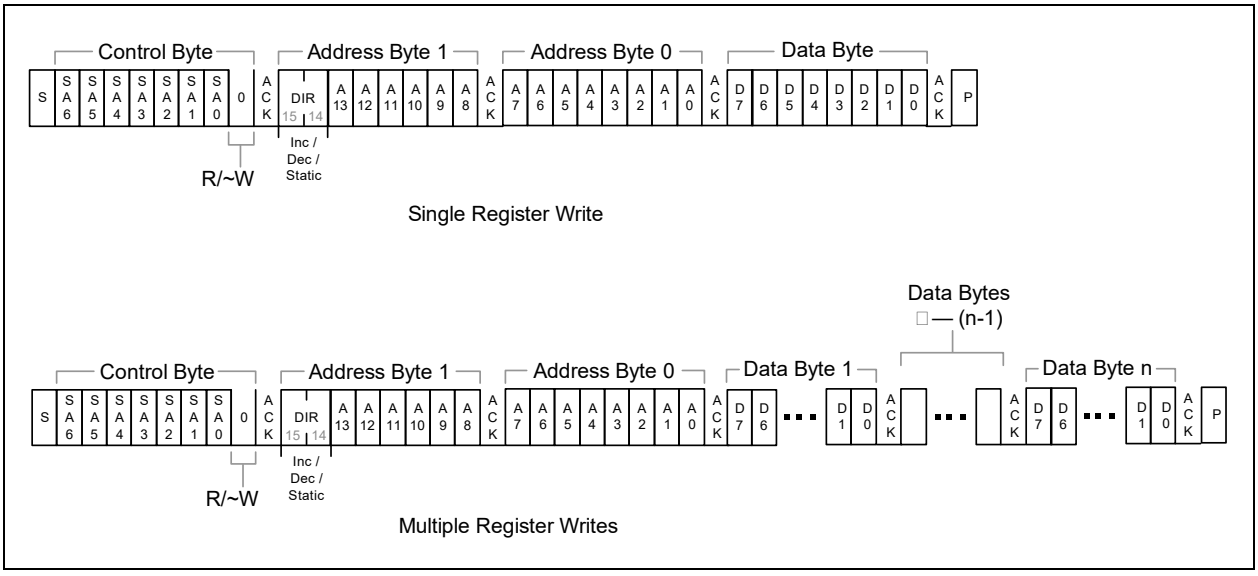
(Section 5.2.2) for each write including the final. This is not relevant for subsequent writes after a new Start condition, since a new register address (with its DIR subfield) must then be included. However, this would affect the address used by any subsequent read without first resetting the register address.

For both single and multiple writes, if the master sends an unexpected start or stop condition, the device will stop immediately and will respond to the next sequence as needed.

The data write to a multi-byte register may be delayed until after all bits are input. In the event that the full register is not written (master sends a start or a stop condition occurs unexpectedly), the write may be considered invalid and the register not affected. Multiple registers may be written in a multiple write cycle, each one being written in sequence. I^2C writes must not be performed to unused register addresses.

Figure 5-5 illustrates a typical single and multiple register write.

FIGURE 5-5: I^2C SLAVE WRITES



5.2.8 SPECIAL CSR HANDLING

5.2.8.1 Live Bits

Register values are latched (registered) at the beginning of each register read to prevent the host from reading a changing value. The latching occurs individually per register in a multiple register read sequence.

5.2.8.2 Change-on-Read Registers and FIFOs

Any single-byte register that triggers a side-effect from a read operation (for example, containing “clear on read” bits, or advancing a FIFO structure) triggers only after the host has begun accessing the value. The value seen by the master will always be the original value and never the updated result of the side-effect.

For a multiple-byte register that is considered a single unit, the change may be delayed until all bytes of the register have been read. In the event that the host sends a no-acknowledge on one of the first bytes of a multi-byte register, or a start or stop condition occurs unexpectedly before the acknowledge of the full register, the read may be considered invalid and the side-effect not triggered.

Registers read in multiple-register read access will trigger multiple side-effects, occurring as they are read. The following registers have read side-effects:

5.2.8.3 Watchdog Count Register (WDT_COUNT)Live Bits that are also Change-on-Read

As described above, the current value from a register with live bits (as is the case of any register) is captured and latched as output data, and Change on Read bits are then changed in the original register. To prevent loss of a hardware event that occurs following the data capture but before the Change on Read, these hardware events are held pending until after the read action and after any change due to the read. This sequence also ensures an edge in the bit due to the hardware event.

6.0 SPI SLAVE CONTROLLER *(UPD350-B/UPD350-D ONLY)*

This chapter details the integrated SPI slave controller (**SPI_DI**, **SPI_DO**, **SPI_CLK**, and **SPI_CS_N**) available in the UPD350-B and UPD350-D. The SPI slave controller can be used for Host CPU serial management and data transfer, and allows host access to all device Configuration and Status Registers.

6.1 SPI Overview

The SPI Slave module provides a low pin count synchronous slave interface that facilitates communication between the device and a host system. The SPI slave allows access to the System CSRs and internal FIFOs and memories. It supports single and multiple register read and write commands with incrementing, decrementing and static addressing. Only a Single bit lane is supported in SPI mode at up to 25 MHz.

The following is an overview of the functions provided by the SPI Slave:

- **Fast Read:** 4-wire (clock, select, data in and data out) reads. Serial command, address and data. This is called “Fast” Read for historical reasons, and is the only Read command supported. There is a single Dummy byte required for first access. Single and multiple register reads with incrementing, decrementing or static addressing.
- **Write:** 4-wire (clock, select, data in and data out) writes at up to 25 MHz. Serial command, address and data. Single and multiple register writes with incrementing, decrementing or static addressing.

6.2 SPI Slave Operation

A SPI frame starts on the falling edge of **SPI_CS_N**, and ends with **SPI_CS_N** rising. At the edges of **SPI_CS_N**, the **SPI_CLK** clock may be at its reset state of either low (Mode 0) or high (Mode 3), at the option of the Master.

Input data on the **SPI_DI** pin (often called “MOSI”) is sampled on the rising edge of the **SPI_CLK** input clock. Output data is launched on the **SPI_DO** pin (often called “MISO”) with the falling edge of the clock. While the **SPI_CS_N** chip select input is high, the **SPI_DI** and **SPI_CLK** inputs are ignored and the **SPI_DO** output is floating.

Each frame starts with an 8-bit instruction byte, transmitted by the Master, and it is accepted on **SPI_DI** starting at the first rising edge of the input clock after **SPI_CS_N** goes active.

For both Write and (Fast) Read instructions, two address bytes follow the instruction byte. The address field expresses a byte address. Fourteen address bits specify the address. The remaining two bits [15:14] constitute the DIR subfield of the address field, which specifies whether the address is Auto-Incremented (00b) or Auto-Decremented (10b) for consecutive data bytes in the frame. A special Static address coding (11b) keeps the address static throughout the frame of data, causing a single byte address to be accessed repeatedly if multiple bytes are transferred in the frame. DIR subfield encoding 01b is reserved and should be decoded in implementation to be the same as 00b, for the sake of minimizing the effect of a software error that increments beyond the address space.

For the Fast Read instruction, one dummy byte follows the address bytes. The dummy byte occupies 8 bits, one per clock.

The device will normally not drive **SPI_DO** during the Instruction, Address or Dummy byte cycles, but see Section 6.2.2, "Access During and Following Power Management," on page 26 for a special case.

For Fast Read instructions, one or more 8-bit data fields follow the dummy byte. For Write instructions, they immediately follow the address bytes.

Individual bytes in instruction, address and data fields are transferred with the most-significant bit (msb) first. The two-byte Address field is transferred with the most-significant byte (MSB) first. Multi-byte data values are transferred in the order specified by the DIR subfield of the Address field (bits [15:14]), and so their order can be effectively selected by using Increment mode (starting from the lowest byte address) or Decrement mode (starting from the highest byte address).

The SPI interface supports a minimum time of 50ns between successive commands (a minimum **SPI_CS_N** inactive time of 50ns).

The instructions supported by the SPI slave controller are listed in Table 6-1. Unsupported instructions are reserved and must not be used.

TABLE 6-1: SPI INSTRUCTIONS

Instruction	Description	Bus Bit Width	Inst. Code	Address Bytes	Dummy Bytes	Data bytes	Max Freq.
Read							
FASTREAD	Read, higher speed format	1	0Bh	2	1	1 to ∞	25 MHz
Write							
WRITE	Write	1	02h	2	0	1 to ∞	25 MHz

6.2.1 DEVICE INITIALIZATION

Until the device has been initialized to the point where the various configuration inputs are valid, the SPI interface will not respond to or be affected by any external pin activity.

Once device initialization completes, the SPI interface will ignore the pins until a rising edge of **SPI_CS_N** is detected.

If the device initialization completes during an active cycle (**SPI_CS_N** low), the trailing end of the frame must be seen (**SPI_CS_N** returning high) before any internal registers are affected or the state of the SPI interface changes.

The first SPI access after device initialization must always be a dummy read to the SPI Test Register (SPI_TEST) (UPD350-B/UPD350-D Only).

6.2.1.1 SPI Slave Read Polling for Initialization Complete

With an external weak pull-up resistor present on **SPI_DO**, a value of FFh will appear to have been read from any internal register while the device is uninitialized. By verifying the SPI Test Register (SPI_TEST) (UPD350-B/UPD350-D Only) has at least one “0” bit in it, it is possible to tell when the device is initialized.

6.2.2 ACCESS DURING AND FOLLOWING POWER MANAGEMENT

During any power management mode other than D0, Reads and Writes are ignored internally except to begin a Wake of the device internally on the falling edge of **SPI_CS_N**. The Wake event on SPI traffic is local to the specific device, and does not affect the states of other devices even on the same SPI bus. Until waking is complete (up to oscillator lock), the SPI interface holds the **SPI_DO** pin low for the duration of the **SPI_CS_N** low time.

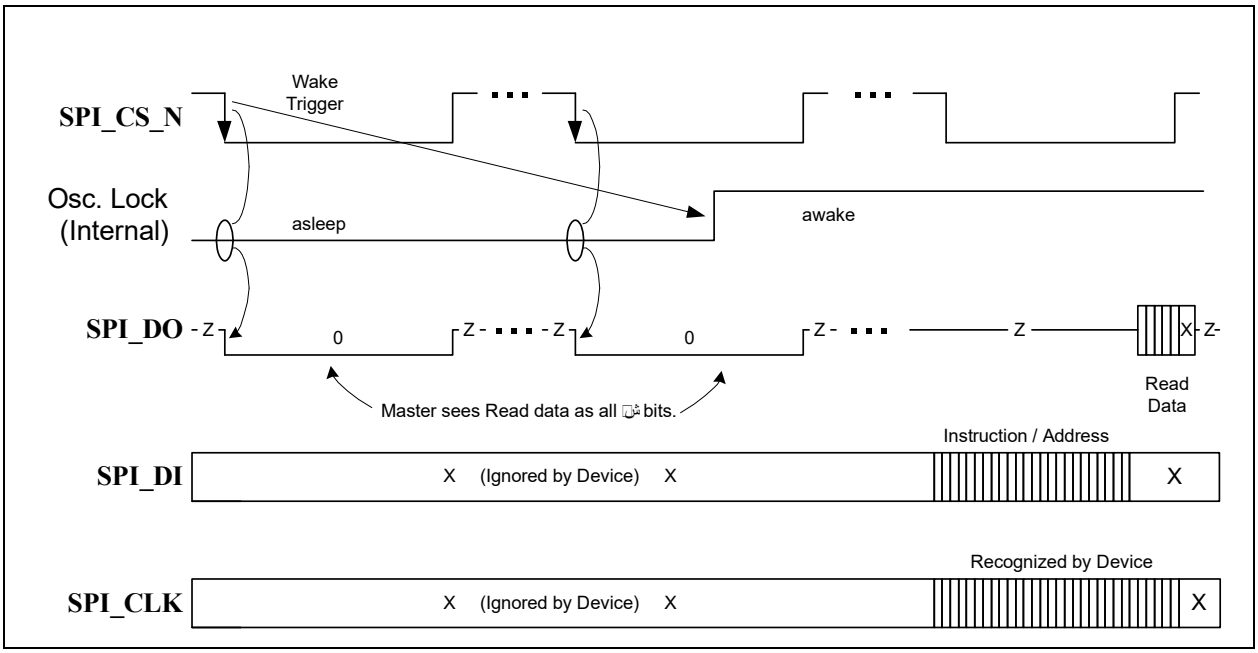
Until the device is awake, then, any Read access performed by the Master will appear to have returned all “1” bits. To determine when the device is awake and the SPI interface functional, the SPI Test Register (SPI_TEST) (UPD350-B/UPD350-D Only) should be repeatedly polled by the Master in separate frames (**SPI_CS_N** low then high). Once a correct, non-zero value is read, the interface can be considered functional. As an alternative to polling, an **IRQ_N** pin assertion can be used to indicate the device is ready.

Once the power management mode changes back to ACTIVE, the SPI interface will still ignore the **SPI_CLK** and **SPI_DI** pins, following **SPI_CS_N** low with **SPI_DO** low, until **SPI_CS_N** is seen high. At the next **SPI_CS_N** falling edge, SPI communication will continue normally.

At any time after performing SPI traffic, the device will not go back to a non-communicating power state until explicitly allowed to do so by a command from the SPI Master.

Figure 6-1 illustrates the sequence of waking from SPI traffic.

FIGURE 6-1: POWER MANAGEMENT WAKE ON SPI TRAFFIC



6.2.3 SPI READ COMMAND (FAST READ)

The Fast Read command is supported by the SPI slave. A single byte, or multiple bytes, may be read in a single frame (**SPI_CS_N** low).

Fast Read is the only form of Read access supported by the device. The instruction inputs the instruction code, the address and a dummy byte on **SPI_DI**, and outputs the data one bit per clock on **SPI_DO**.

The SPI slave interface is selected by first bringing **SPI_CS_N** active. The 8-bit FASTREAD instruction, 0Bh, is input into the **SPI_DI** pin, followed by the two address bytes and 1 dummy byte. The address bytes specify a Byte register address within the device, and also specify how addresses are sequenced for successive bytes in a Multiple Byte Read (below). The contents of the dummy byte are don't-care.

On the falling clock edge following the rising edge of the last dummy bit, the **SPI_DO** pin is driven starting with the most significant bit of the selected register byte. The remaining register bits are shifted out on subsequent falling clock edges.

The **SPI_CS_N** input is brought inactive to conclude the cycle. The **SPI_DO** pin is floated by the device in response.

6.2.3.1 Multiple Byte Reads

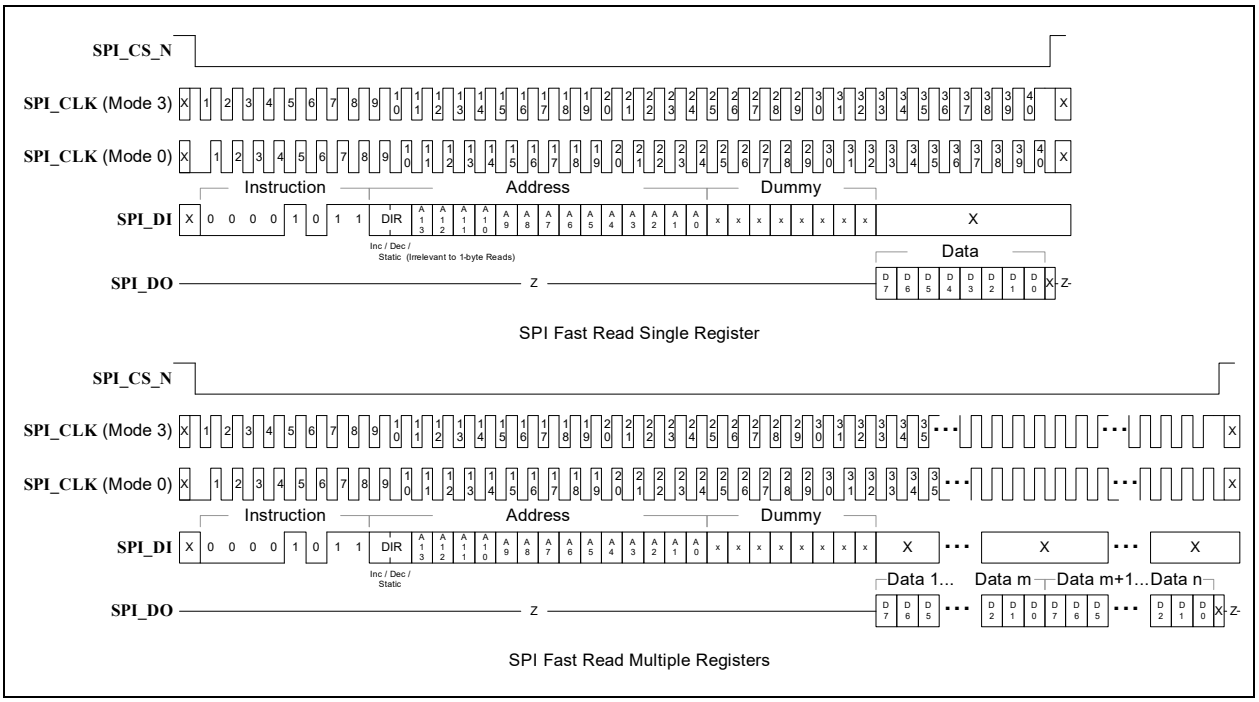
Additional byte reads beyond the first are performed by the Master by continuing the clock pulses while **SPI_CS_N** is active. The upper two bits [15:14] (DIR subfield) of the address specify Auto-Incrementing (DIR=00b) or Auto-Decrementing (DIR=10b) or Static (fixed) (DIR=11b) for successive bytes read. Maintaining a Static internal address is provided for FIFO Read/Write or low-level register polling within a single frame, if the Master supports it.

Towards the end of the current one-byte output shift the address is incremented or decremented, if appropriate, and another synchronized capture sequence is done.

6.2.3.2 Fast Read

Figure 6-2 illustrates a typical single and multiple register fast read for SPI mode.

FIGURE 6-2: SPI FAST READ



6.2.4 SPI WRITE COMMANDS

The following write commands are supported by the SPI slave controller:

- Write
- Multiple Writes

6.2.4.1 Write

The Write instruction provides the instruction code and address and data bytes on the **SPI_DI** pin, one bit per clock.

The SPI transfer is started by the Master first driving **SPI_CS_N** active. The 8-bit WRITE instruction, 02h, is given on the **SPI_DI** pin, followed by the two address bytes. The address bytes specify a byte address within the device, and a Direction control subfield (DIR).

The data immediately follows the address bytes on the **SPI_DI** pin, starting with the most significant bit of the first byte. The data is input from the **SPI_DI** pin by the device, shifted in on each subsequent rising clock edge.

6.2.4.2 Multiple Writes

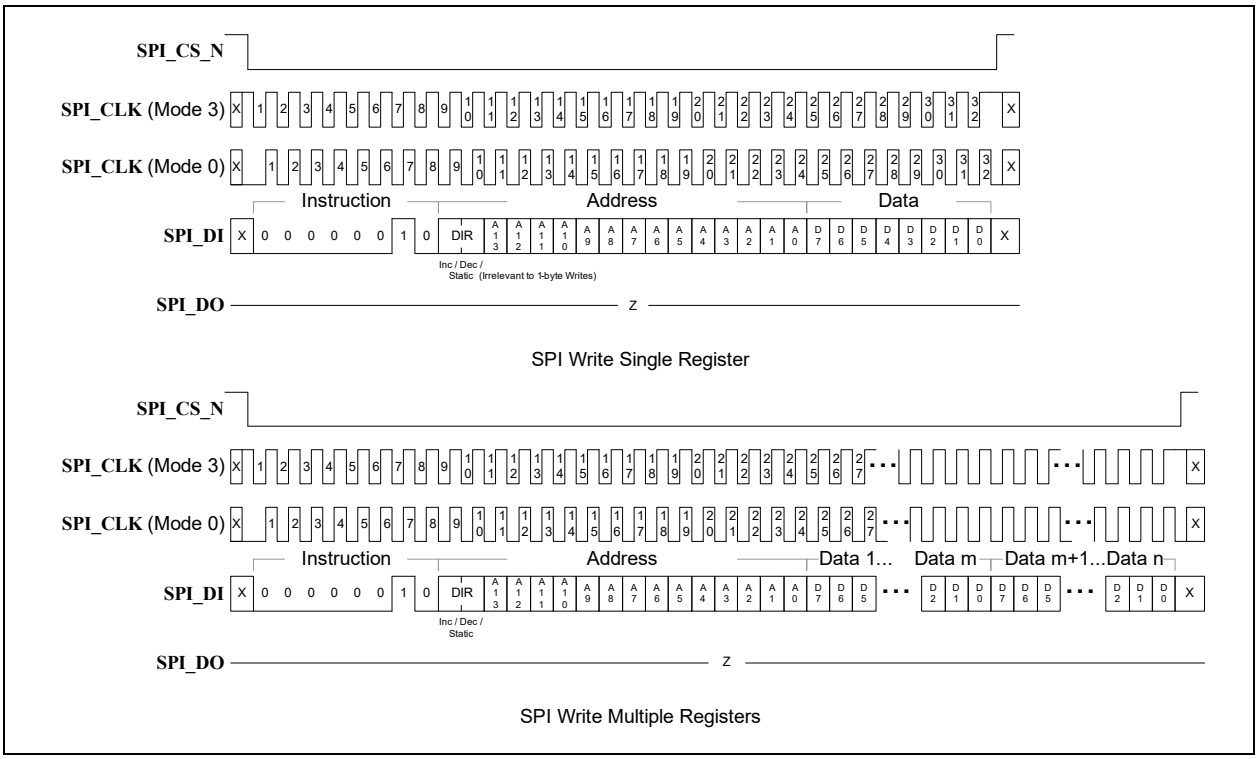
Multiple writes are performed by the Master by continuing the clock pulses and **SPI_DI** data while **SPI_CS_N** remains active. The upper two bits [15:14] of the address constitute the DIR subfield, and specify auto-incrementing (DIR=00b) or auto-decrementing (DIR=10b) or Static addressing. The internal Byte address is incremented, decremented, or kept fixed (Static) based on these bits. Maintaining a fixed internal address may be useful for FIFO access, register “bit-banging” or other repeated activity.

The data write to the register occurs after the full register contents are input: this depends on the defined size of the register. In the event that the full register is not written when **SPI_CS_N** is returned high, the write is considered invalid and the register is not affected.

The **SPI_CS_N** input is then brought inactive to conclude the cycle.

Figure 6-3 illustrates a typical SPI single and multiple register write.

FIGURE 6-3: SPI WRITE



6.2.5 SPECIAL CSR HANDLING

6.2.5.1 Live Bits

Register values are latched (registered) at the beginning of each register read to prevent the host from reading a changing value. The latching occurs individually per register in a multiple register read sequence.

6.2.5.2 Change-on-Read Registers and FIFOs

Any single-byte register that triggers a side-effect from a read operation (for example, containing “clear on read” bits, or advancing a FIFO structure) triggers only after the host has begun accessing the value. The value seen by the master will always be the original value and never the updated result of the side-effect.

For a multiple-byte register that is considered a single unit, the change may be delayed until all bytes of the register have been read. In the event that the host sends a no-acknowledge on one of the first bytes of a multi-byte register, or a start or stop condition occurs unexpectedly before the acknowledge of the full register, the read may be considered invalid and the side-effect not triggered.

Registers read in multiple-register read access will trigger multiple side-effects, occurring as they are read. The following registers have read side-effects:

6.2.5.3 Watchdog Count Register (WDT_COUNT)Live Bits that are also Change-on-Read

As described above, the current value from a register with live bits (as is the case of any register) is captured and latched as output data, and Change on Read bits are then changed in the original register. To prevent loss of a hardware event that occurs following the data capture but before the Change on Read, these hardware events are held pending until after the read action and after any change due to the read. This sequence also ensures an edge in the bit due to the hardware event.

7.0 CLOCKS, RESETS, AND POWER MANAGEMENT

This section details the various clocks, resets, and power managements states of the device:

7.1 Clocks

The following internal clocks are generated by the device:

- **48 MHz Relaxation Oscillator**
- **20 KHz Keep Alive Oscillator**
- **Ring Oscillator**

These oscillators can be manually enabled/disabled via software the Clock Control Register (CLK_CTL).

7.2 Power States

The device supports the following power states, as defined in their respective sub-sections:

- SLEEP
- HIBERNATE
- STANDBY
- ATTACHED IDLE (FRS Enabled)
- ATTACHED IDLE (FRS Disabled)
- ACTIVE

7.2.1 SLEEP

This is the lowest power state of the device. The SLEEP state is entered via assertion of the **PWR_DN** pin. Virtually all of the device is powered off in this mode with minimal circuity in the 3.3V domain to detect deassertion of **PWR_DN**.

This mode is intended to minimize power consumption when the device is not being used in battery powered applications. In these applications, a wake up event such as a button press, can cause the host CPU to deassert **PWR_DN**.

7.2.2 HIBERNATE

In this state, the port is disabled by the USB PD firmware and the **PWR_DN** pin is low. Attach detection is disabled due to CC terminations in the high-impedance state.

7.2.3 STANDBY

STANDBY is the lowest power functional state of the device. The majority of the device is powered off in this state. The internal CC comparator and 20 KHz oscillator are enabled in this state as well as requisite analog components (1.8V LDO, PORs, Biases, etc).

The CC lines are constantly monitored for an attach condition which shall result in an interrupt assertion to the host. If an attachment has been made, this state can detect a change in the partner's advertisement as well as a detach.

STANDBY is the power state that the UPD350 device will be in when in USB Type-C® Unattached.SRC/SNK.

7.2.4 ATTACHED IDLE (FRS ENABLED)

In this state, a USB Type-C® device is connected and the USB PD bus is idle (no USB packets in transit). The CC signals are constantly being monitored for packet transmission and Fast Role Swap (FRS) signal detection is enabled.

7.2.5 ATTACHED IDLE (FRS DISABLED)

In this state, a USB Type-C® device is connected and the USB PD bus is idle (no USB packets in transit). The CC signals are constantly being monitored for packet transmission and Fast Role Swap (FRS) signal detection is disabled.

The following asynchronous wake-ups are supported from this state:

- Asynchronous I2C Wakeup (UPD350-A/UPD350-C Only)
- Asynchronous SPI Wakeup (UPD350-B/UPD350-D Only)
- Power Delivery MAC Wakeup

7.2.6 ACTIVE

This state defines the condition of the device after an attachment occurred. In this state, Power Delivery communication is supported. This state is also used for any condition in which the 48 MHz Relaxation Oscillator must be enabled, such as when it is desired to debounce a GPIO within the micro-second range.

When transmitting a Power Delivery packet, an additional 5 mA may be consumed. Additional power consumption results from enabling the OCS comparator, VBUS comparator and other modules. When VCONN FETs are enabled, there is an additional 70 mW of power consumption.

7.3 Asynchronous I²C Wakeup *(UPD350-A/UPD350-C Only)*

The device supports asynchronous wakes on the I²C slave interface. Via clock stretching, the I²C transaction that caused the wakeup will not be lost and does not have to be repeated by the host. The device will not clock stretch for more than 3 us.

The following steps illustrate the I²C wake function. Initially the Ring Oscillator and 48 MHz Oscillator are disabled.

1. The Host initiates an I²C transaction to the device.
2. The device asynchronously detects reception of the Start Bit and enables clock stretching by pulling-down **I2C_CLK** after the host drives SCL low. The Ring Oscillator is asynchronously enabled and used as a clock source for the power management logic.
3. After a delay of approximately 5 us the oscillator stabilizes and clocks the I²C controller.
4. Clock stretching is disabled and the I²C controller is enabled and begins processing the pending transaction.
5. The I²C transaction completes.
6. The Host checks the device status to see if there are any pending transactions. The I²C transaction may have initiated a PD transmission or conversely a coincident PD transaction may be in the process of being received.
7. After host confirms the device has no pending transactions, it power downs the device by disabling the Ring Oscillator and 48 MHz Relaxation Oscillator. via Clock Control Register (CLK_CTL).
8. The device is ready to accept future asynchronous I²C wake event.

7.4 Asynchronous SPI Wakeup *(UPD350-B/UPD350-D Only)*

UPD350 supports asynchronous wakes on the SPI interface. The SPI protocol for this device is defined such that there is no requirement that the SPI transaction must be repeated.

The following steps illustrate the SPI wake function. Initially the Ring Oscillator and 48 MHz Oscillator are disabled.

1. The device is powered down.
2. The Host initiates an SPI transaction to the SPI Test Register (SPI_TEST) (UPD350-B/UPD350-D Only) which, when the device is operational, returns a non-zero value. The device drives **SPI_DO** to 0b while in power-down.
3. The device detects reception of an SPI message. The Ring Oscillator is asynchronously enabled and used as a clock source for the power management logic.
4. After a delay of approximately 5 us the oscillator stabilizes and clocks the SPI controller.
5. The device processes the next received SPI transaction.
6. The SPI transaction(s) complete(s).
7. The Host checks the device status to see if there are any pending transactions. The SPI transaction may have initiated a PD transmission or conversely a coincident PD transaction may be in the process of being received.
8. After the Host confirms the device has no pending transactions, it powers down the device by disabling the Ring Oscillator, and 48 MHz Relaxation Oscillator. via Clock Control Register (CLK_CTL).
9. The device drives **SPI_DO** to 0b and awaits an asynchronous SPI wake.

7.5 Power Delivery MAC Wakeup

The PD MAC is capable of asynchronous wakeup upon reception of a PD packet. This enables the device to be placed in STANDBY mode and minimize power consumption.

The following steps illustrate the RX PD MAC wake function. Initially the Ring Oscillator and the 48 MHz Relaxation Oscillator are disabled. The 20 KHz Keep Alive Oscillator is enabled, but not used in the wake process.

1. In order to receive a PD message, the RX AFE is enabled via software and the trip point is set. via the CC RX DAC Control Register (CC_RX_DAC_CTL) and CC RX DAC Filter Register (CC_RX_DAC_FILT).
2. The PD MAC is configured and enabled via software.
3. The device is powered down. Software disables the 48 MHz Relaxation Oscillator, and Ring Oscillator if enabled.via the Clock Control Register (CLK_CTL).
4. After some time elapses, the device receives a PD message from the attached partner.
5. The PD MAC asynchronously detects preamble activity on the CC line and enables the 48 MHz Relaxation Oscillator.
6. The oscillator is operational in approximately 5 us, at which point the PD MAC is operational.
7. The PD MAC receives the remainder of the preamble and stores the message into the RX FIFO, and in accordance with the PD protocol, responds with GoodCRC as required.
8. An interrupt is issued via assertion of the **IRQ_N** pin.
9. Software services the interrupt via I^2C, reads the PD message, and responds as required.
10. The device is then powered down. Software disables the 48 MHz Relaxation Oscillator and Ring Oscillator. via the Clock Control Register (CLK_CTL).
11. The device remains powered down until the next PD message is received.

7.6 Interrupt Assertion from STANDBY

When the device is in STANDBY it is able to detect a number of events that may be configured to assert the **IRQ_N** pin upon being appropriately programmed via software. The logic detecting such events operates off of the 20 KHz Keep Alive Oscillator.

Upon the occurrence of an event, which is programmed to assert **IRQ_N**, the 48 MHz Relaxation Oscillator and Ring Oscillator are enabled. Upon synchronization, the **IRQ_N** pin shall assert.

After software services the source interrupt(s), it should disable the 48 MHz Relaxation Oscillator and Ring Oscillator via the Clock Control Register (CLK_CTL) to place the part back in STANDBY.

The following example sequence illustrates the steps for configuring the device to detect an OCS event on the **OCS_COMP1** pin while in STANDBY.

1. Software enables the OCS Compare Interrupt. OCS_CMP_INT bit in the Interrupt Enable Register (INT_EN).
2. Software enables OCS detection, as defined in Section 8.3, "External Over-current Detection", to sample the **OCS_COMP1** pin.
3. Software disables the 48 MHz Relaxation Oscillator and 1 MHz Ring Oscillator. via the Clock Control Register (CLK_CTL)
4. An OCS event occurs and is detected on **OCS_COMP1**.
5. The Ring Oscillator is enabled and is used to as the device's operational clock.
6. The device enables the 48 MHz oscillator and waits about 5 us for the oscillator to stabilize.
7. The 48 MHz oscillator is stable.
8. After synchronization is complete, the **IRQ_N** pin is asserted.
9. Software detects **IRQ_N** assertion and services the interrupt.
10. Software disables the 48 MHz Relaxation Oscillator and Ring Oscillator. via the Clock Control Register (CLK_CTL) to place the part back in STANDBY.

7.7 Reset Operation

The following chip-level resets are supported by the device:

- Power-On-Reset (POR)
- Pin Reset (**RESET_N**)
- Software Reset (SRESET in Hardware Control Register (HW_CTL))
- Watchdog Timer(WDT_STS)

Chip-level resets trigger the sampling of the **CFG_SEL** configuration strap (see Section 9.9.1, "Configuration Selection," on page 82 for additional information). Chip-level reset completion can be determined by assertion of the RDY_INT bit in the Interrupt Enable Register (INT_EN) and assertion of the **IRQ_N** pin.

The following is s summary of steps that occur after a chip-level reset.

1. System level reset event (POR, **RESET_N**, SRESET, WDT_STS) occurs.
2. The device enables the 20 KHz Keep Alive Oscillator, 48 MHz Relaxation Oscillator, and 1MHz Ring Oscillator.
3. The device samples the **CFG_SEL** pin.
4. The device configures itself in accordance with the **CFG_SEL** pin and settings.
5. The device is enabled, the RDY_INT bit in the Interrupt Enable Register (INT_EN) asserts, the **IRQ_N** pin asserts.
6. The device disables the 48 MHz Relaxation Oscillator and the Ring Oscillator.

7.8 Clocks and Power Management Registers

This section details the clocks and power management registers. For an overview of the entire device register map, refer to Section 4.0, "Register Map," on page 18.

TABLE 7-1: CLOCKS AND POWER MANAGEMENT REGISTERS MAP

Address	Register Name (Symbol)
1000h	Clock Control Register (CLK_CTL)
1001h – 13FFh	Reserved for future expansion

Note:	**RESERVED address space must not be written under any circumstances. Failure to heed this warning may result in untoward operation and unexpected results.**

7.8.1 CLOCK CONTROL REGISTER (CLK_CTL)

Address: 1000h Size: 8 bits

Bits	Description	Type	Default
7:3	**RESERVED**	RO	-
2	**Ring Oscillator Enable** 0: Oscillator is disabled. 1: Oscillator is enabled.	R/W	0b
1	**48 MHz Relaxation Oscillator Enable** 0: Oscillator is disabled. 1: Oscillator is enabled.	R/W	Note 7-1
0	**20 KHz Keep Alive Oscillator Enable** 0: Oscillator is disabled. 1: Oscillator is enabled.	R/W	1b

Note 7-1 This bit will always appear asserted when read by software, as the 48 MHz Relaxation Oscillator must be enabled to facilitate I^2C/SPI transactions.

8.0 SYSTEM CONTROL

This section details the following system controls:

- System Interrupts
- General Purpose I/O
- External Over-current Detection
- System Control Registers

8.1 System Interrupts

The device provides an active-low and open drain interrupt pin, **IRQ_N**, which is asserted as a function of the System Control's Interrupt Status Register (INT_STS)Interrupt Status Register (INT_STS) and Interrupt Enable Register (INT_EN)Interrupt Enable Register (INT_EN). The following interrupts are supported:

- HPD Interrupt
- Power Switch Module Interrupt
- Watchdog Timer Assertion Interrupt
- Port Power Controller Interrupt
- Power Delivery MAC Interrupt
- OCS Comparator Interrupt
- PIO Override Interrupt
- PIO Interrupt
- CC Interrupt
- VBUS Interrupt
- Power Control Interrupt
- Device Ready Interrupt

In order to clear an interrupt status bit, the source of the asserted bit in Interrupt Status Register (INT_STS) must be cleared. This process typically involves dealing with the module (e.g. PD MAC, CC, Port Power Controller, Watchdog Timer, Power Switch, etc.) that triggered the interrupt or configuring the respective module's interrupt enable register to disable assertion.

The PIO interrupts are cleared by writing "1" to the asserted bit(s) in PIO Interrupt Status Register (PIO_INT_STS). The PIO Override interrupts are cleared by writing "1" to the asserted bit(s) in PIO Override Interrupt Status Register (PIO_OVR_INT_STS). The OCS interrupts are cleared by is cleared by writing "1" to the asserted bit(s) in OCS Comparator Change Status Register (OCS_CMP_CHG_STS).

Figure 8-1 illustrates the device's interrupt hierarchy. Figure 8-2 illustrates the generation of the CC InterruptCC_INT bit in the Interrupt Status Register (INT_STS).

FIGURE 8-1: SYSTEM INTERRUPT HIERARCHY

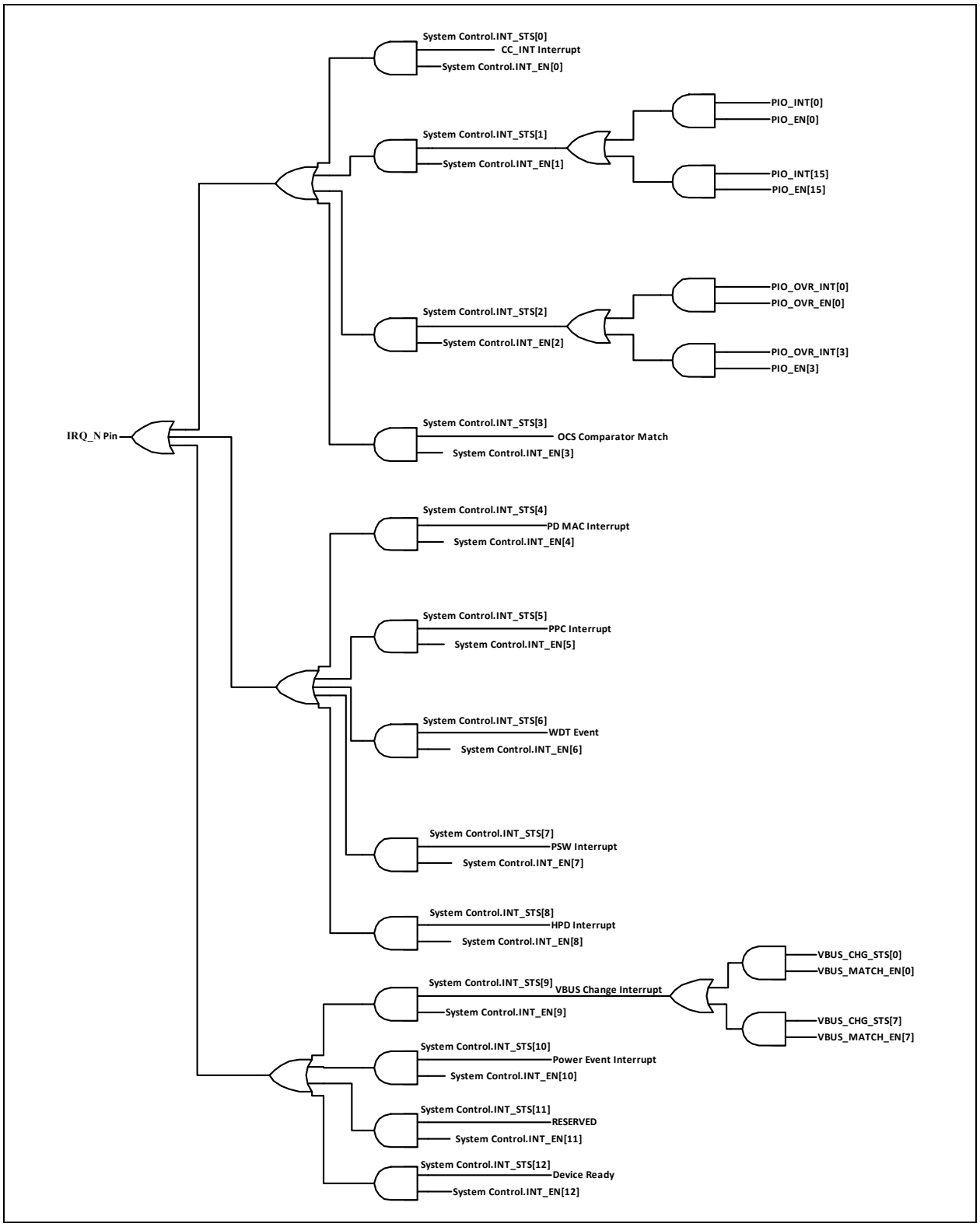
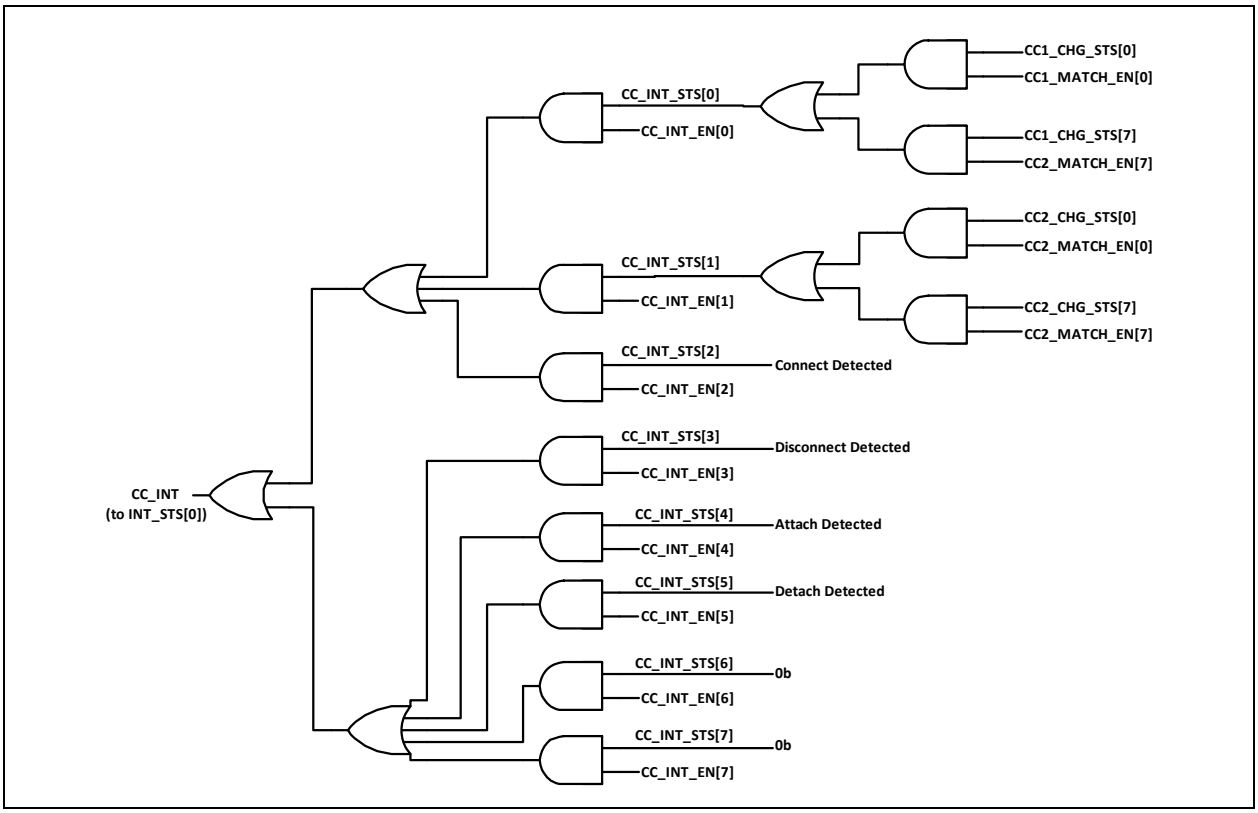


FIGURE 8-2: CC_INT INTERRUPT HIERARCHY



8.2 General Purpose I/O

A key function of the UPD350 is to manage external devices via up to ten PIOs. Usually this is accomplished via host software programming of the PIOs.

In some cases the UPD350 must automatically override the PIO state in response to an error condition. Features have been incorporated into the design to enable such operation. This is required in cases where the latency introduced by reliance on software to affect PIO state is either or too long or not deterministic.

A PIO is configured via the respective Configure PIOx Registers (CFG_PIOx). A PIO may also be configured to assert PIO_INT in Interrupt Status Register (INT_STS). The state of a PIO is known by reading the PIO Status Register (PIO_STS).

A mechanism exists for multiple PIO outputs to be updated via a single register write. This is accomplished via the Configure PIO Output Register (CFG_PIO_OUT). Writes to this register that change the output value of a PIO are reflected in the DataOutput bit of the respective Configure PIOx Registers (CFG_PIOx). Likewise, writes to DataOutput in the Configure PIOx Registers (CFG_PIOx) will update the respective bit in the Configure PIO Output Register (CFG_PIO_OUT).

Note:	**The GPO1 and GPO10 general purpose signals can only function as outputs and must be kept in a low state coincident with de-asserting RESET_N.**

Note:	**GPIO0** and **GPIO1** are not available in the UPD350-B / UPD350-D.

Note:	In Standalone DFP/UFP mode *(UPD350-A/UPD350-C only)*, certain GPIOs have alternate dedicated functions, as defined in Section 3.1.1.1, "UPD350-A / UPD350-C GPIO Functions in Standalone DFP/UFP Modes," on page 10.

8.2.1 PIO SIMULTANEOUS UPDATE

A mechanism has been incorporated to enable multiple PIOs to be updated simultaneously. This is desirable to simplify software and prevent devices controlled by the UPD350 from entering an intermediary unknown state. Of particular concern would be devices responsible for managing power.

The Configure PIOx Registers (CFG_PIOx) CSR is updated by this function. When the simultaneous update request is accepted by hardware, the contents of the PIO registers are simultaneously updated.

The example below illustrates simultaneous PIO update operation:

1. Software changes Configure PIOx Update Registers (CFG_PIOx_UPD) as required.
2. Software changes Configure PIO Output Update Register (CFG_PIO_OUT_UPD) as required.

Note:	The Configure PIO Output Update Register (CFG_PIO_OUT_UPD) and Configure PIOx Update Registers (CFG_PIOx_UPD) must be programmed consistently by software. If the output value bits are different between the two registers, then the value in Configure PIOx Update Registers (CFG_PIOx_UPD) will be used for updating the Configure PIOx Registers (CFG_PIOx) and the respective bit in Configure PIO Output Register (CFG_PIO_OUT).

3. Software sets the Configure PIOx Update Registers (CFG_PIOx_UPD) to initiate the update.
4. A pending PIO override blocks acceptance of the update request. Only after the pending PIO override function is cleared will the device clear PIO_UPD_REQ. The pending PIO override is cleared via the PIO Override Enable Register (PIO_OVR_EN).
5. After the update is accepted, PIO_UPD_REQ clears automatically.

APPLICATION NOTE: If a PIO override is pending, the PIO simultaneous update is blocked.

8.2.2 PIO OVERRIDE

8.2.2.1 Overview

A mechanism is defined in this section to enable the state of a PIO to be changed without software intervention. This provides a mechanism to instantaneously disable a power circuit, controlled by a UPD350 PIO, based upon a change in the state of a selected PIO input, OCS detection or other event.

8.2.2.2 Description

Four override functions are included in device. For each function, a PIO or other source must be selected for monitoring via the corresponding PIO Override x Source Select Registers (PIO_OVRx_SRC_SEL). The value to be monitored is set in the PIO Override Monitor Value Register (PIO_OVR_MON_VAL) and is compared with the state of the respective PIO. When these values match for an override, the enabled PIO outputs change their state accordingly.

PIO outputs controlled by an override are enabled by the PIO Override x Output Enable Registers (PIO_OVRx_OUT_EN). Each override has an instance in this register. A PIO may be used across multiple overrides.

Firmware can be notified that an override was executed via assertion of PIO_OVERRIDE_INT in the Interrupt Status Register (INT_STS), if Interrupt Enable Register (INT_EN) is appropriately configured.

8.3 External Over-current Detection

The device incorporates an analog comparator DAC circuit to detect an over-current condition. This feature is supported via the **OCS_COMP1** and/or **OCS_COMP2** pin.

This feature is enabled as follows:

1. Software clears the OCS Comparator Control bit in the OCS Comparator Control Register (OCS_CMP_CTL) to determine the OCS debouncer.
2. Software polls OCS_DB_ACTIVE until it clears, which indicates the OCS debouncer is disabled.
3. If **OCS_COMP2** is desired, the shared PIO must be disabled via the Configure PIOx Registers (CFG_PIOx).
4. Software programs one or both of the OCS Comparator x Threshold Registers (OCS_CMPx_THR), as required.
5. Software programs the OCS Comparator Debounce Register (OCS_CMP_DEBOUNCE).
6. Software enables the **OCS_COMP*x*** pin(s) via the OCS Comparator Match Enable Register (OCS_CMP_-MATCH_EN).
7. Software enables the respective interrupt in the Interrupt Enable Register (INT_EN).
8. Software sets the OCS Comparator Control bit in the OCS Comparator Control Register (OCS_CMP_CTL) to enable comparison on **OCS_COMP1** and/or **OCS_COMP2**, as desired.
9. Detection of an over current event by the comparator results in the respective bit of the OCS Comparator Change Status Register (OCS_CMP_CHG_STS) and OCS_CMP_INT asserts. The **IRQ_N** pin will assert, if configured.
10. Software writes to the OCS Comparator Change Status Register (OCS_CMP_CHG_STS) to clear OCS_C-MP_INT. For a Lo-to-Hi transition, the respective bit in the OCS Comparator Match Register (OCS_CMP_-MATCH) is set, while for a Hi-to-Lo transition the bit is cleared.

This feature is disabled when software clears the OCS Comparator Control bit in the OCS Comparator Control Register (OCS_CMP_CTL). Either **OCS_COMP*x*** pin may be used to trigger a PIO override.

8.4 General Purpose Timer

The device incorporates a low power general purpose timer that operates off a 20 kHz oscillator and implements a 16-bit one-shot down counter. When the timer underflows, it asserts the and interrupt GP_UFLOW bit in General Purpose Timer Interrupt Source Register (GP_TIMER_INT_SRC) and stops counting. This in turn causes the GP_TIMER_INT to assert in Interrupt Status Register (INT_STS). The interrupt persists until the GP_UFLOW bit is cleared. If the respective bit is set in Interrupt Enable Register (INT_EN), the **IRQ_N** pin will assert.

The timer's upper threshold is loaded via the General Purpose Timer Load Register (GP_TIMER_LOAD). Setting RESET in the General Purpose Timer Control Register (GP_TIMER_CTL) causes the timer's internal counter to be loaded to this value.

The timer is enabled by setting ENABLE in the General Purpose Timer Control Register (GP_TIMER_CTL). This causes the timer to decrement every 1 ms. The timer continues to decrement until it underflows or it is disabled.

Note:	The underflow condition is defined as when the down counter transitions from 0x0000 to 0xFFFF.

Clearing the ENABLE bit in the General Purpose Timer Control Register (GP_TIMER_CTL) disables the timer and it stops counting.

The timer's current count can be read from General Purpose Timer Count Register (GP_TIMER_COUNT) The timer should be halted by clearing ENABLE before reading it. After the count is read, the ENABLE may be set again and the counter will resume.

The DIS_ON_UFLOW bit in the General Purpose Timer Control Register (GP_TIMER_CTL) causes the timer to automatically disable upon underflow detection and assert GP_UFLOW. In this case the assertion of GP_UFLOW is delayed until the timer is fully disabled. in the 20 KHz clock domain.

APPLICATION NOTE: This timer enables clocking sources to be disabled in the host CPU which is useful for timing events such as DRP. This helps provide for a lower total power system solution.

The following steps illustrate the usage model for DRP implementation:

1. Software configures the device as a DFP or UFP. See Section 9.4, "DRP Operation (Legacy)" for further details.
2. Software enables the GP_TIMER_INT in the Interrupt Enable Register (INT_EN).

3. Software programs the General Purpose Timer Load Register (GP_TIMER_LOAD) to a random value that complies with the range for tDRP period as defined in the USB Type-C Specification.
4. Software sets the ENABLE, RESET and DIS_ON_UFLOW bits in the General Purpose Timer Control Register (GP_TIMER_CTL). This causes the value in the General Purpose Timer Load Register (GP_TIMER_LOAD) to be loaded into the timer and it begins decrementing. This also configures the timer to delay assertion of GP_UFLOW in the General Purpose Timer Interrupt Source Register (GP_TIMER_INT_SRC) until the timer is fully disabled (portion operated in 20 KHz domain).

APPLICATION NOTE: Writes to General Purpose Timer Load Register (GP_TIMER_LOAD) and General Purpose Timer Control Register (GP_TIMER_CTL) defined in the prior steps may be done in the same I^2C write burst.

5. Upon timer expiration, the GP_UFLOW bit will be set and the GP_TIMER_INT asserts in Interrupt Status Register (INT_STS).
6. The **IRQ_N** pin asserts.
7. Software services the IRQ and determines whether or not a connection has been established with a UFP, or DFP. If no connection has been established, software changes the device's role. See Section 9.4, "DRP Operation (Legacy)" for further details.
8. Software programs the General Purpose Timer Load Register (GP_TIMER_LOAD) to a random value that complies with the range of the tDRP period as defined in the USB Type-C Specification.
9. Software shall clear the GP_UFLOW bit in the General Purpose Timer Interrupt Source Register (GP_TIMER_INT_SRC).
10. Software sets the ENABLE, RESET and DIS_ON_UFLOW bits in General Purpose Timer Control Register (GP_TIMER_CTL). This causes the value in General Purpose Timer Load Register (GP_TIMER_LOAD) to be loaded into the timer and the timer begins decrementing.

APPLICATION NOTE: Writes to the General Purpose Timer Load Register (GP_TIMER_LOAD), General Purpose Timer Interrupt Source Register (GP_TIMER_INT_SRC) and General Purpose Timer Control Register (GP_TIMER_CTL) defined in the prior steps may be done in the same I^2C write burst.

11. Upon timer expiration, and timer disablement, the GP_UFLOW bit will be set and the GP_TIMER_INT asserts in Interrupt Status Register (INT_STS).
12. The **IRQ_N** pin asserts.
13. Software services the IRQ and determines if a connection has been established with a UFP, or DFP. If no connection has been established SW changes the device's role. See Section 9.4, "DRP Operation (Legacy)" for further details.
14. Goto step 8.

8.5 System Control Registers

This section details the system control registers. For an overview of the entire device register map, refer to Section 4.0, "Register Map," on page 18.

TABLE 8-1: SYSTEM CONTROL REGISTER MAP

Address	Register Name (Symbol)
0000h	Device ID Register (ID_REV)
0004h	USB Vendor ID Register (VID)
0006h	USB Product ID Register (PID)
0008h	USB PD Revision Register (PD_REV)
000Ah	USB Type-C® Revision Register (C_REV)
000Bh - 000Dh	Reserved for future expansion
000Eh	SPI Test Register (SPI_TEST) (UPD350-B/UPD350-D Only)
0010h	Interrupt Status Register (INT_STS)
0014h	Interrupt Enable Register (INT_EN)
0018h	Hardware Control Register (HW_CTL)
001Ah	I2C Slave Address Register (I2C_ADDR) (UPD350-A/UPD350-C Only)
001Bh - 001Fh	Reserved for future expansion
0020h	PIO Status Register (PIO_STS)
0022h	PIO Interrupt Status Register (PIO_INT_STS)
0024h	PIO Interrupt Enable Register (PIO_INT_EN)
0026h - 002Fh	Reserved for future expansion
0030h	Configure PIOx Registers (CFG_PIOx) x=0
0031h	Configure PIOx Registers (CFG_PIOx) x=1
0032h	Configure PIOx Registers (CFG_PIOx) x=2
0033h	Configure PIOx Registers (CFG_PIOx) x=3
0034h	Configure PIOx Registers (CFG_PIOx) x=4
0035h	Configure PIOx Registers (CFG_PIOx) x=5
0036h	Configure PIOx Registers (CFG_PIOx) x=6
0037h	Configure PIOx Registers (CFG_PIOx) x=7
0038h	Configure PIOx Registers (CFG_PIOx) x=8
0039h	Configure PIOx Registers (CFG_PIOx) x=9
003Ah	Configure PIOx Registers (CFG_PIOx) x=10
003Bh	Configure PIOx Registers (CFG_PIOx) x=11
003Ch	Configure PIOx Registers (CFG_PIOx) x=12
003Dh	Configure PIOx Registers (CFG_PIOx) x=13
003Eh	Configure PIOx Registers (CFG_PIOx) x=14
003Fh	Configure PIOx Registers (CFG_PIOx) x=15
0040h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=0
0041h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=1
0042h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=2
0043h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=3
0044h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=4
0045h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=5
0046h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=6
0047h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=7
0048h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=8

TABLE 8-1: SYSTEM CONTROL REGISTER MAP (CONTINUED)

Address	Register Name (Symbol)
0049h	Configure PIOx Update Registers (CFG_PIOx_UPD) x=9
004Ah	Configure PIOx Update Registers (CFG_PIOx_UPD) x=10
004Bh	Configure PIOx Update Registers (CFG_PIOx_UPD) x=11
004Ch	Configure PIOx Update Registers (CFG_PIOx_UPD) x=12
004Dh	Configure PIOx Update Registers (CFG_PIOx_UPD) x=13
004Eh	Configure PIOx Update Registers (CFG_PIOx_UPD) x=14
004Fh	Configure PIOx Update Registers (CFG_PIOx_UPD) x=15
0050h	PIO Override x Source Select Registers (PIO_OVRx_SRC_SEL) x=0
0051h	PIO Override x Source Select Registers (PIO_OVRx_SRC_SEL) x=1
0052h	PIO Override x Source Select Registers (PIO_OVRx_SRC_SEL) x=2
0053h	PIO Override x Source Select Registers (PIO_OVRx_SRC_SEL) x=3
0054h	PIO Override Monitor Value Register (PIO_OVR_MON_VAL)
0055h	PIO Override Enable Register (PIO_OVR_EN)
0056h	PIO Override Status Register (PIO_OVR_STS)
0057h	PIO Override Interrupt Status Register (PIO_OVR_INT_STS)
0058h	PIO Override Interrupt Enable Register (PIO_OVR_INT_EN)
0059h	Configure PIO Output Register (CFG_PIO_OUT)
005Bh	Configure PIO Output Update Register (CFG_PIO_OUT_UPD)
005Dh - 005Fh	Reserved for future expansion
0060h	PIO Override x Output Enable Registers (PIO_OVRx_OUT_EN) x=0
0062h	PIO Override x Output Enable Registers (PIO_OVRx_OUT_EN) x=1
0064h	PIO Override x Output Enable Registers (PIO_OVRx_OUT_EN) x=2
0066h	PIO Override x Output Enable Registers (PIO_OVRx_OUT_EN) x=3
0068h	PIO Override Output Register (PIO_OVR_OUT)
006Ah	PIO Override Direction Register (PIO_OVR_DIR)
006Ch	PIO Debounce Count Ten Millisecond Register (PIO_DEBOUNCE_10MS_COUNT)
006Dh	PIO Debounce Count One Millisecond Register (PIO_DEBOUNCE_1MS_COUNT)
006Eh	PIO Debounce Count One Microsecond Register (PIO_DEBOUNCE_1US_COUNT)
006Fh	PIO Debounce Status Register (PIO_DEBOUNCE_STS)
0070h	PIO Debounce Enable Register (PIO_DEBOUNCE_EN)
0074h – 007Fh	Reserved for future expansion
0080h	OCS Comparator Control Register (OCS_CMP_CTL)
0081h	OCS Comparator Match Register (OCS_CMP_MATCH)
0082h	OCS Comparator Change Status Register (OCS_CMP_CHG_STS)
0083h	OCS Comparator Match Enable Register (OCS_CMP_MATCH_EN)
0084h	OCS Comparator x Threshold Registers (OCS_CMPx_THR) x=1
0086h	OCS Comparator x Threshold Registers (OCS_CMPx_THR) x=2
0088h	OCS Comparator Debounce Register (OCS_CMP_DEBOUNCE)
0089h – 009Ah	Reserved for future expansion
009Bh	IRQ_N PU/PD Control Register (IRQ_PUPD)
009Ch	Voltage Regulator Control Register (VREG_CTL)
009Dh	I2C Trim Register (I2C_TRIM)
009E – 009Fh	Reserved for future expansion
00A0h	General Purpose Timer Load Register (GP_TIMER_LOAD)

TABLE 8-1: SYSTEM CONTROL REGISTER MAP (CONTINUED)

Address	Register Name (Symbol)
00A2h	General Purpose Timer Interrupt Source Register (GP_TIMER_INT_SRC)
00A3h	General Purpose Timer Control Register (GP_TIMER_CTL)
00A4h	General Purpose Timer Count Register (GP_TIMER_COUNT)
00A6h – 03FFh	Reserved for future expansion

Note: RESERVED address space must not be written under any circumstances. Failure to heed this warning may result in untoward operation and unexpected results.

8.5.1 DEVICE ID REGISTER (ID_REV)

Address: 0000h Size: 32 bits

Bits	Description	Type	Default
31:16	**Device ID (ID)**	RO	Note 8-1
15:0	**Device Revision (REV)**	RO	Note 8-2

Note 8-1 The default value of this field is dependent on the version of the device.
UPD350-A: 0350h
UPD350-B: 0351h
UPD350-C: 0352h
UPD350-D: 0353h

Note 8-2 The default value of this field is dependent on the silicon revision of the device.

ID_REV[7:0] = 00h
ID_REV[15:8] = 01h
ID_REV[23:16] = 02h
ID_REV[31:24] = 03h

8.5.2 USB VENDOR ID REGISTER (VID)

Address: 0004h Size: 16 bits

Bits	Description	Type	Default
15:0	**USB Vendor Identification (VID)**	R/W	0424h

VID[7:0] = 04h

VID[15:8] = 05h

8.5.3 USB PRODUCT ID REGISTER (PID)

Address: 0006h Size: 16 bits

Bits	Description	Type	Default
15:0	**USB Product Identification (PID)**	R/W	0350h

PID[7:0] = 06h

PID[15:8] = 07h

8.5.4 USB PD REVISION REGISTER (PD_REV)

Address: 0008h Size: 16 bits

Bits	Description	Type	Default
15:0	**USB Power Delivery Specification Revision (PD_REV)**	R/W	Note 8-3

Note 8-3 The default value of this field is loaded from the respective OTP field. If OTP is not valid, the default value is 3010h.

PD_REV[7:0] = 08h

PD_REV[15:8] = 09h

8.5.5 USB TYPE-C® REVISION REGISTER (C_REV)

Address: 000Ah Size: 16 bits

Bits	Description	Type	Default
15:0	**USB Type-C® Specification Revision (C_REV)**	R/W	Note 8-4

Note 8-4 The default value of this field is loaded from the respective OTP field. If OTP is not valid, the default value is 12h.

C_REV[7:0] = 0Ah

C_REV[15:8] = 0Bh

8.5.6 SPI TEST REGISTER (SPI_TEST) *(UPD350-B/UPD350-D ONLY)*

Address: 000Eh Size: 8 bits

.

Bits	Description	Type	Default
7:0	**SPI Test (SPI_TEST)** This register is used by the host to determine when the chip has come out of powerdown when waking it via the SPI interface.	RO	02h

8.5.7 INTERRUPT STATUS REGISTER (INT_STS)

Address: 0010h Size: 16 bits

Bits	Description	Type	Default
15	**RESERVED**	RO	0h
14	**GP_TIMER_INT** Interrupt generated when the general purpose timer underflows. This interrupt reflects the value of the GP_UFLOW bit of the General Purpose Timer Interrupt Source Register (GP_TIMER_INT_SRC).	RO	0b
13	**RESERVED**	RO	0b
12	**RDY_INT** Interrupt generated when the device is configured and ready for access by host CPU after a system level reset event. This interrupt reflects the value of the DEV_READY bit of the Hardware Control Register (HW_CTL).	RO	0b
11	**EXT_INT** Interrupt generated by the Cable Plug Orientation and Detection module.	RO	0b
10	**PWR_INT** Interrupt generated power event. This interrupt is cleared via the Power Interrupt Status Register (PWR_INT_STS).	RO	0b
9	**VBUS_INT** Indicates that a change occurred in the state of the VBUS Change Status Register (VBUS_CHG_STS).	RO	0b
8	**HPD_INT** Interrupt generated by the HPD detection logic. This interrupt is cleared via the HPD Interrupt Status Register (HPD_INT_STS).	RO	0b
7	**PSW_INT** Interrupt generated by the power switch. This interrupt is cleared via the Power Switch Interrupt Status Register (PWR_INT_STS).	RO	0b

Bits	Description	Type	Default
6	**WDT_INT** Interrupt generated by watchdog timer expiration. This interrupt is cleared via WDT_STS bit in the Watchdog Control Register (WDT_CTL). **Note:** When a Watchdog interrupt is pending, all write operations are blocked until the WDT_STS bit in the Watchdog Control Register (WDT_CTL) is cleared.	RO	0b
5	**PPC_INT** Interrupt generated by PPC.	RO	0b
4	**MAC_INT** Interrupt generated by the PD MAC. This interrupt is cleared via the corresponding PD MAC interrupt register, as detailed in Section 11.4.3, "PD MAC Interrupt Status and Enable Registers," on page 155.	RO	0b
3	**OCS_CMP_INT** Interrupt generated from the detection of an over-current condition. This interrupt is cleared via OCS Comparator Change Status Register (OCS_CMP_CHG_STS).	RO	0b
2	**PIO_OVERRIDE_INT** Interrupt generated from a PIO override. This interrupt is cleared via the PIO Override Interrupt Status Register (PIO_OVR_INT_STS).	RO	0b
1	**PIO_INT** Interrupt generated from configured PIO status change. This interrupt is cleared via the PIO Interrupt Status Register (PIO_INT_STS).	RO	0b
0	**CC_INT** Interrupt generated by Cable Plug Orientation and Detection module. This interrupt is cleared via the CC Interrupt Status Register (CC_INT_STS).	RO	0b

INT_STS[7:0] = 10h

INT_STS[15:8] = 11h

8.5.8 INTERRUPT ENABLE REGISTER (INT_EN)

Address: 0014h Size: 16 bits

Bits	Description	Type	Default
15	**RESERVED**	RO	0h
14	**Interrupt Enable 13** When “0”, prevents generation of the respective interrupt.	R/W	0b
13	**RESERVED**	RO	-
12	**Interrupt Enable 12** When “0”, prevents generation of the respective interrupt.	R/W	1b
11	**Interrupt Enable 11** When “0”, prevents generation of the respective interrupt.	R/W	0b
10	**Interrupt Enable 10** When “0”, prevents generation of the respective interrupt.	R/W	0b
9	**Interrupt Enable 9** When “0”, prevents generation of the respective interrupt.	R/W	0b
8	**Interrupt Enable 8** When “0”, prevents generation of the respective interrupt.	R/W	0b
7	**Interrupt Enable 7** When “0”, prevents generation of the respective interrupt.	R/W	0b
6	**Interrupt Enable 6** When “0”, prevents generation of the respective interrupt.	R/W	1b
5	**Interrupt Enable 5** When “0”, prevents generation of the respective interrupt.	R/W	0b
4	**Interrupt Enable 4** When “0”, prevents generation of the respective interrupt.	R/W	0b
3	**Interrupt Enable 3** When “0”, prevents generation of the respective interrupt.	R/W	0b
2	**Interrupt Enable 2** When “0”, prevents generation of the respective interrupt.	R/W	0b
1	**Interrupt Enable 1** When “0”, prevents generation of the respective interrupt.	R/W	0b
0	**Interrupt Enable 0** When “0”, prevents generation of the respective interrupt.	R/W	0b

INT_EN[7:0] = 12h

INT_EN[15:8] = 13h

The default value is loaded from RESET_MASK OTP field.

8.5.9 HARDWARE CONTROL REGISTER (HW_CTL)

Address: 0018h Size: 16 bits

Bits	Description	Type	Default
15:6	**RESERVED**	RO	-
5	**I2C_GEN_CALL_EN** 0: I2C General Call address not supported. 1: I2C General Call address supported.	R/W	0b
4	**RESERVED**	RO	-
3	**I2C FMPLUS** When set, this bit configures the I2C interface to operate in fast-mode plus mode. **Note:** UPD350-A/UPD350-C only.	R/W	1b
2	**DEV_READY** When set, this bit indicates that the device has been fully configured after a system level reset event.	R/SC	0b
1	**PIO_UPD_REQ** When this bit is set, the PIO registers are updated from the respective simultaneous update registers. After the operation completes, this bit is automatically cleared. The request will not be accepted if an override function is pending. After the override condition is cleared by software, the update request will be accepted. The override is cleared via respective bit of PIO Override Enable Register (PIO_OVR_EN).	R/SC	0b
0	**SRESET** Software reset. When set this bit initiates a chip level reset.	SC	0b

8.5.10 I2C SLAVE ADDRESS REGISTER (I2C_ADDR) *(UPD350-A/UPD350-C ONLY)*

Address: 001Ah Size: 8 bits

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6:0	**I2C Slave Address** Defines the slave address used by the I2C controller.	R/W	Note 8-5

Note 8-5 The default value for this register is defined by the CFG_SEL pin unless the OTP I2C_ADR_OVR_EN bit is set. In this case, the OTP value specified by I2C_ADDR_OVR[6:0] is loaded into the register.

8.5.11 PIO STATUS REGISTER (PIO_STS)

Address: 0020h Size: 16 bits

Bits	Description	Type	Default
15	**PIO15** 0b: PIO is low 1b: PIO is high	RO	0b
14	**PIO14** 0b: PIO is low 1b: PIO is high	RO	0b
13	**PIO13** 0b: PIO is low 1b: PIO is high	RO	0b
12	**PIO12** 0b: PIO is low 1b: PIO is high	RO	0b
11	**PIO11** 0b: PIO is low 1b: PIO is high	RO	0b
10	**PIO10** 0b: PIO is low 1b: PIO is high	RO	0b
9	**PIO9** 0b: PIO is low 1b: PIO is high	RO	0b
8	**PIO8** 0b: PIO is low 1b: PIO is high	RO	0b
7	**PIO7** 0b: PIO is low 1b: PIO is high	RO	0b
6	**PIO6** 0b: PIO is low 1b: PIO is high	RO	0b
5	**PIO5** 0b: PIO is low 1b: PIO is high	RO	0b
4	**PIO4** 0b: PIO is low 1b: PIO is high	RO	0b
3	**PIO3** 0b: PIO is low 1b: PIO is high	RO	0b
2	**PIO2** 0b: PIO is low 1b: PIO is high	RO	0b
1	**PIO1** 0b: PIO is low 1b: PIO is high	RO	0b
0	**PIO0** 0b: PIO is low 1b: PIO is high	RO	0b

PIO_STS[0] = 20h

PIO_STS[1] = 21h

Note:	**The GPO1 and GPO10 general purpose signals can only function as outputs and must be kept in a low state coincident with de-asserting RESET_N.**

Note:	**GPIO0** is not available in the UPD350-B/UPD350-D.

Note:	In Standalone DFP/UFP modes, GPIOs 9-15 have alternate dedicated functions, as defined in Section 3.1.1.1, "UPD350-A / UPD350-C GPIO Functions in Standalone DFP/UFP Modes," on page 10.

8.5.12 PIO INTERRUPT STATUS REGISTER (PIO_INT_STS)

Address: 0022h Size: 16 bits

Bits	Description	Type	Default
15:0	**PIO_INT[15:0]** When set, a level change has occurred on PIO[15:0]. Interrupts are cleared by writing a '1' to the bit.	R/WC	0000h

Note:	Writing a "1" (one) to a bit clears the bit and enables the detection of the next level transition. If not disabled by the corresponding bit in the PIO_EN register, "1" in any bit in this register will force assertion of PIO_INT interrupt.

8.5.13 PIO INTERRUPT ENABLE REGISTER (PIO_INT_EN)

Address: 0024h Size: 16 bits

Bits	Description	Type	Default
15:0	**PIO_EN[15:0]** When set, an interrupt can be generated on a state change from the corresponding PIO line.	R/W	0h

8.5.14 CONFIGURE PIOX REGISTERS (CFG_PIO*X*)

Address: x=0: 0030h
x=1: 0031h
x=2: 0032h
x=3: 0033h
x=4: 0034h
x=5: 0035h
x=6: 0036h
x=7: 0037h
x=8: 0038h
x=9: 0039h
x=10: 003Ah
x=11: 003Bh
x=12: 003Ch
x=13: 003Dh
x=14: 003Eh
x=15: 003Fh

Size: 8 bits

Bits	Description	Type	Default
7	**PullUpEnable** 0b: No pull-up 1b: Pull-up enabled.	R/W	0b
6	**PullDownEnable** 0b: No pull-down 1b: Pull-down enabled.	R/W	0b
5	**FallingAlert** 0b: No alert on falling edge 1b: Assert PIO_INT on falling edge	R/W	0b
4	**RisingAlert** 0b: No alert on rising edge 1b: Assert PIO_INT on rising edge	R/W	0b
3	**DataOutput** 0b: Inactive (low for push-pull or open-drain) 1b: Active (high for push-pull, high-z for open-drain) **Note:** The input buffer status can be determined from the respective bit in PIO Status Register (PIO_STS). **Note:** The associated PIO output bit in the Configure PIO Output Register (CFG_PIO_OUT) will be updated to match the value written to this bit.	R/W	0b
2	**Buffer Type** When set, the output buffer for the corresponding GPIO signal is configured as a push/pull driver. When cleared, the corresponding GPIO signal is configured as an open-drain driver.	R/W	0b
1	**Direction** 0b: Input 1b: Output	R/W	0b
0	**Enable** 0b: GPIO Disabled 1b: GPIO Enabled **Note:** When a GPIO is disabled, the default function, if any, will be functional. All GPIOs default to being enabled and configured as inputs.	R/W	1b

Note: **The GPO1 and GPO10 general purpose signals can only function as outputs and must be kept in a low state coincident with de-asserting RESET_N.**

Note: **GPIO0** is not available in the UPD350-B/UPD350-D.

Note: In Standalone DFP/UFP modes, GPIOs 9-15 have alternate dedicated functions, as defined in Section 3.1.1.1, "UPD350-A / UPD350-C GPIO Functions in Standalone DFP/UFP Modes," on page 10.

8.5.15 CONFIGURE PIOX UPDATE REGISTERS (CFG_PIO*X*_UPD)

Address: x=0: 0040h
x=1: 0041h
x=2: 0042h
x=3: 0043h
x=4: 0044h
x=5: 0045h
x=6: 0046h
x=7: 0047h
x=8: 0048h
x=9: 0049h
x=10: 004Ah
x=11: 004Bh
x=12: 004Ch
x=13: 004Dh
x=14: 004Eh
x=15: 004Fh

Size: 8 bits

Bits	Description	Type	Default
7	**PullUpEnabled**	R/W	0b
6	**PullDownEnabled**	R/W	0b
5	**FallingAlert**	R/W	0b
4	**RisingAlert**	R/W	0b
3	**OutputValue**	R/W	0b
2	**BufferType**	R/W	0b
1	**Direction**	R/W	0b
0	**Enable**	R/W	0b

Note: **The GPO1 and GPO10 general purpose signals can only function as outputs and must be kept in a low state coincident with de-asserting RESET_N.**

Note: **GPIO0** is not available in the UPD350-B/UPD350-D.

Note: In Standalone DFP/UFP modes, GPIOs 9-15 have alternate dedicated functions, as defined in Section 3.1.1.1, "UPD350-A / UPD350-C GPIO Functions in Standalone DFP/UFP Modes," on page 10.

8.5.16 PIO OVERRIDE *X* SOURCE SELECT REGISTERS (PIO_OVR*X*_SRC_SEL)

Address: x=0: 0050h
x=1: 0051h
x=2: 0052h
x=3: 0053h

Size: 8 bits

Bits	Description	Type	Default
7:5	**VBUS Threshold Select** This field selects which VBUS threshold shall be matched when selecting VBUS_DET for override. 000b: VSAFE0V_THR_MATCH 001b: VBUS0_THR_MATCH 010b: VBUS1_THR_MATCH 011b: VBUS2_THR_MATCH 1xxb: VBUS3_THR_MATCH	R/W	0h
4:0	**Override Select** This field selects the source for the respective PIO override. 0h: PIO0 1h: PIO1 2h: PIO2/OCS_COMP2 3h: PIO3 4h: PIO4 5h: PIO5 6h: PIO6 7h: PIO7 8h: PIO8 9h: PIO9 Ah: PIO10 Bh: PIO11 Ch: PIO12 Dh: PIO13 Eh: PIO14 Fh: PIO15 10h: OCS_COMP1 11h: VBUS_DET 12h-1Fh: Reserved	R/W	0h

8.5.17 PIO OVERRIDE MONITOR VALUE REGISTER (PIO_OVR_MON_VAL)

Address: 0054h Size: 8 bits

Bits	Description	Type	Default
7:4	**RESERVED**	RO	-
3	**PIO_MON_VAL3** Override function 3 monitors for this value on the selected PIO.	R/W	0b
2	**PIO_MON_VAL2** Override function 2 monitors for this value on the selected PIO.	R/W	0b
1	**PIO_MON_VAL1** Override function 1 monitors for this value on the selected PIO.	R/W	0b
0	**PIO_MON_VAL0** Override function 0 monitors for this value on the selected PIO.	R/W	0b

Note: When the PIO is mapped to an OCS comparator or VBUS comparator, the monitored case is rising above/falling below the associated threshold. Setting the override monitor value indicates that the source exceeding the threshold is monitored. If the value is cleared then being below the threshold is monitored.

8.5.18 PIO OVERRIDE ENABLE REGISTER (PIO_OVR_EN)

Address: 0055h Size: 8 bits

Bits	Description	Type	Default
7:4	**RESERVED**	RO	-
3	**PIO_OVR_EN3** When set PIO Override 3 is enabled.	R/W	0b
2	**PIO_OVR_EN2** When set PIO Override 2 is enabled.	R/W	0b
1	**PIO_OVR_EN1** When set PIO Override 1 is enabled.	R/W	0b
0	**PIO_OVR_EN0** When set PIO Override 0 is enabled.	R/W	0b

8.5.19 PIO OVERRIDE STATUS REGISTER (PIO_OVR_STS)

Address: 0056h Size: 8 bits

Bits	Description	Type	Default
7:4	**RESERVED**	RO	-
3	**PIO_OVR_STS3** Override function is active on the selected PIO. Set on a match to PIO_MON_VAL3 and clears on override feature disable PIO_OVR_EN3 =0.	RO	0b
2	**PIO_OVR_STS2** Override function is active on the selected PIO. Set on a match to PIO_MON_VAL2 and clears on override feature disable PIO_OVR_EN2 =0.	RO	0b
1	**PIO_OVR_STS1** Override function is active on the selected PIO. Set on a match to PIO_MON_VAL1 and clears on override feature disable PIO_OVR_EN1 =0.	RO	0b
0	**PIO_OVR_STS0** Override function is active on the selected PIO. Set on a match to PIO_MON_VAL0 and clears on override feature disable PIO_OVR_EN0 =0.	RO	0b

8.5.20 PIO OVERRIDE INTERRUPT STATUS REGISTER (PIO_OVR_INT_STS)

Address: 0057h Size: 8 bits

Bits	Description	Type	Default
7:4	**RESERVED**	RO	-
3	**PIO_OVR_INTR3** This interrupt is set when override function 3 detects the programmed monitored value on the selected PIO. The interrupt persists until cleared.	R/WC	0b
2	**PIO_OVR_INTR2** This interrupt is set when override function 2 detects the programmed monitored value on the selected PIO. The interrupt persists until cleared.	R/WC	0b
1	**PIO_OVR_INTR1** This interrupt is set when override function 1 detects the programmed monitored value on the selected PIO. The interrupt persists until cleared.	R/WC	0b
0	**PIO_OVR_INTR0** This interrupt is set when override function 0 detects the programmed monitored value on the selected PIO. The interrupt persists until cleared.	R/WC	0b

8.5.21 PIO OVERRIDE INTERRUPT ENABLE REGISTER (PIO_OVR_INT_EN)

Address: 0058h Size: 8 bits

Bits	Description	Type	Default
7:4	**RESERVED**	RO	0h
3	**PIO_OVR_INTR3** When ‘1’, enables the generation of this interrupt.	R/W	0b
2	**PIO_OVR_INTR2** When ‘1’, enables the generation of this interrupt.	R/W	0b
1	**PIO_OVR_INTR1** When ‘1’, enables the generation of this interrupt.	R/W	0b
0	**PIO_OVR_INTR0** When ‘1’, enables the generation of this interrupt.	R/W	0b

8.5.22 CONFIGURE PIO OUTPUT REGISTER (CFG_PIO_OUT)

Address: 0059h Size: 16 bits

This register enables multiple PIO output values to be updated in a single CSR write.

Bits	Description	Type	Default
15:0	**CFG_PIO_OUT** When this register is written, the output values for the respective PIOs are updated. Specifically, the DataOutput in the respective CFG_PIOx registers are updated. This register is not updated until the byte located at 005Ah is written by software. This is to prevent placing externally driven components in an invalid state while updating.	R/W	0000h

8.5.23 CONFIGURE PIO OUTPUT UPDATE REGISTER (CFG_PIO_OUT_UPD)

Address: 005Bh Size: 16 bits

Bits	Description	Type	Default
15:0	**CFG_PIO_OUTx_UPD**	R/W	0000h

8.5.24 PIO OVERRIDE *X* OUTPUT ENABLE REGISTERS (PIO_OVR*X*_OUT_EN)

Address: x=0: 0060h
x=1: 0062h
x=2: 0064h
x=3: 0066h
Size: 16 bits

Bits	Description	Type	Default
15:0	**PIO_OVR_OUT_EN_X** Each of the overrides has an instance of this CSR to indicate which PIO outputs are controlled by it. When asserted associated PIO output's state is controlled by the respective override function. **Note:** The same PIO output can be enabled by multiple overrides.	R/W	0h

8.5.25 PIO OVERRIDE OUTPUT REGISTER (PIO_OVR_OUT)

Address: 0068h Size: 16 bits

Bits	Description	Type	Default
15:0	**PIO Override Output** This field defines the PIO output value to be used when the respective PIO is enabled for override and the associated override is triggered.	R/W	0h

8.5.26 PIO OVERRIDE DIRECTION REGISTER (PIO_OVR_DIR)

Address: 006Ah Size: 16 bits

Bits	Description	Type	Default
15:0	**PIO Override Direction** This field defines the PIO direction value to be used when the respective PIO is enabled for override and the associated override is triggered.	R/W	0h

8.5.27 PIO DEBOUNCE COUNT TEN MILLISECOND REGISTER (PIO_DEBOUNCE_10MS_COUNT)

Address: 006Ch Size: 8 bits

Bits	Description	Type	Default
7:0	**PIO_DEBOUNCE_10MS_COUNT** This register holds the 10 ms debounce counter. Any transition within the debounce period is suppressed. Each count corresponds to 10 ms, with the default value being 100 ms. This register is used when the debounce granularity is 10 ms. **Note:** The actual count may be either count or count-1.	R/W	0Ah

8.5.28 PIO DEBOUNCE COUNT ONE MILLISECOND REGISTER (PIO_DEBOUNCE_1MS_COUNT)

Address: 006Dh Size: 8 bits

Bits	Description	Type	Default
7:0	**PIO_DEBOUNCE_1MS_COUNT** This register holds the 1 ms debounce counter. Any transition within the debounce period is suppressed. Each count corresponds to 1 ms, with the default value being 10 ms. This register is used when the debounce granularity is 1 ms. **Note:** The actual count may be either count or count-1.	R/W	0Ah

8.5.29 PIO DEBOUNCE COUNT ONE MICROSECOND REGISTER (PIO_DEBOUNCE_1US_COUNT)

Address: 006Eh Size: 8 bits

Bits	Description	Type	Default
7:0	**PIO_DEBOUNCE_1US_COUNT** This register holds the 1 us debounce counter. Any transition within the debounce period is suppressed. Each count corresponds to 1 us, with the default value being 10 us. This register is used when the debounce granularity is 1 us. **Note:** The actual count may be either count or count-1.	R/W	0Ah

8.5.30 PIO DEBOUNCE STATUS REGISTER (PIO_DEBOUNCE_STS)

Address: 006Fh Size: 8 bits

Bits	Description	Type	Default
7:1	**RESERVED**	RO	0h
0	**PIO_DB_ACT** PIO Debounce Active This bit is used to indicate that a PIO debounce is in process. 1b: PIO debounce pending 0b: PIO not pending	RO	0b

8.5.31 PIO DEBOUNCE ENABLE REGISTER (PIO_DEBOUNCE_EN)

Address: 0070h Size: 32 bits

This register enables a PIO for debouncing and selects the debouncer to be used as defined below:

- 00b: Debounce disabled
- 01b: Debounce Enabled (PIO Debounce Count One Microsecond Register (PIO_DEBOUNCE_1US_COUNT))
- 10b: Debounce Enabled (PIO Debounce Count One Millisecond Register (PIO_DEBOUNCE_1MS_COUNT))
- 11b: Debounce Enabled (PIO Debounce Count Ten Millisecond Register (PIO_DEBOUNCE_10MS_COUNT))

Bits	Description	Type	Default
31:30	**PIO_DB_EN15**	R/W	00b
29:28	**PIO_DB_EN14**	R/W	00b
27:26	**PIO_DB_EN13**	R/W	00b
25:24	**PIO_DB_EN12**	R/W	00b
23:22	**PIO_DB_EN11**	R/W	00b
21:20	**PIO_DB_EN10**	R/W	00b
19:18	**PIO_DB_EN9**	R/W	00b
17:16	**PIO_DB_EN8**	R/W	00b
15:14	**PIO_DB_EN7**	R/W	00b
13:12	**PIO_DB_EN6**	R/W	00b
11:10	**PIO_DB_EN5**	R/W	00b
9:8	**PIO_DB_EN4**	R/W	00b
7:6	**PIO_DB_EN3**	R/W	00b

Bits	Description	Type	Default
5:4	**PIO_DB_EN2**	R/W	00b
3:2	**PIO_DB_EN1**	R/W	00b
1:0	**PIO_DB_EN0**	R/W	00b

Note: **GPIO0** is not available in the UPD350-B/UPD350-D.

8.5.32 OCS COMPARATOR CONTROL REGISTER (OCS_CMP_CTL)

Address: 0080h Size: 8 bits

Bits	Description	Type	Default
7	**OCS_DB_ACTIVE** When this bit reads back 0b, the debouncer is disabled. The OCS debouncer is enabled when it reads back 1b.	RO	0b
6:2	**RESERVED**	RO	0h
1:0	**OCS Comparator Control** 00b: OCS Comparator and DAC are powered down 01b: OCS Comparator samples **OCS_COMP1** 10b: OCS Comparator samples **OCS_COMP2** 11b: OCS Comparator samples **OCS_COMP1** and **OCS_COMP2** When set to 01b or 10b, a sample is taken every 100 us. When set to 11b, each **OCS_COMPx** pin is effectively sampled every 200 us.	R/W	00b

8.5.33 OCS COMPARATOR MATCH REGISTER (OCS_CMP_MATCH)

Address: 0081h Size: 8 bits

This bit is always cleared when OCS Comparator Control is not enabled for the respective OCS_COMP pin.

Bits	Description	Type	Default
7:2	**RESERVED**	RO	0h
1	**OCS Comparator Match 2** When set, this bit indicates that a match has occurred and an OCS condition has been detected on **OCS_COMP2** pin.	RO	0b
0	**OCS Comparator Match 1** When set, this bit indicates that a match has occurred and an OCS condition has been detected on **OCS_COMP1** pin.	RO	0b

8.5.34 OCS COMPARATOR CHANGE STATUS REGISTER (OCS_CMP_CHG_STS)

Address: 0082h Size: 8 bits

Bits	Description	Type	Default
7:6	**OCSx Match Valid (OCSx_MATCH_VLD)** Asserts after the OCS debouncer is first enabled, via OCS Comparator Control, and the first match becomes valid in OCS Comparator Match Register (OCS_CMP_MATCH). Bit 7 = OCS*2*_MATCH_VLD Bit 6 = OCS*1*_MATCH_VLD	R/WC	0b
5:2	**RESERVED**	RO	0h
1:0	**OCSx Change Status (OCSx_CHG_STS)** When set, each bit indicates that the respective bit in OCS Comparator Match Register (OCS_CMP_MATCH) has changed. Bit 1 = OCS*2*_CHG_STS Bit 0 = OCS*1*_CHG_STS A write of 1b clears the respective status bit.	R/WC	00b

8.5.35 OCS COMPARATOR MATCH ENABLE REGISTER (OCS_CMP_MATCH_EN)

Address: 0083h Size: 8 bits

Bits	Description	Type	Default
7:6	**OCSx Match Valid Enable (OCSx_MATCH_VLD_EN)** When set, the corresponding bit in OCS Comparator Change Status Register (OCS_CMP_CHG_STS) can cause the assertion of the OCS_CMP_INT interrupt. Bit 7 = OCS*2*_MATCH_VLD_EN Bit 6 = OCS*1*_MATCH_VLD_EN	R/W	0b
5:2	**RESERVED**	RO	0h
1:0	**OCSx Comparator Match Enable (OCSx_CMP_MATCH_EN)** When set the corresponding bit in OCS Comparator Change Status Register (OCS_CMP_CHG_STS) can cause the assertion of the OCS_CMP_INT interrupt. Bit 1 = OCS*2*_CMP_MATCH_EN Bit 0 = OCS*1*_CMP_MATCH_EN	R/W	00b

8.5.36 OCS COMPARATOR *X* THRESHOLD REGISTERS (OCS_CMP*X*_THR)

Address: x=1: 0084h
x=2: 0086h
Size: 16 bits

Bits	Description	Type	Default
15:10	**RESERVED**	RO	0h
9:0	**OCS Comparator Threshold** The lower byte of the threshold must be written before the upper byte. The entire 10-bit threshold is updated when the second write occurs. This has units of ~2.5V/1024. **Note:** This register shall not be modified while the OCS debouncer is enabled.	R/W	0h

APPLICATION NOTE: This register may be dynamically written to by firmware while the OCS comparator is enabled provided the rules for updating defined in the register description are followed.

8.5.37 OCS COMPARATOR DEBOUNCE REGISTER (OCS_CMP_DEBOUNCE)

Address: 0088h
Size: 8 bits

Bits	Description	Type	Default
7:0	**OCS_CMP_DEBOUNCE** Indicates the period of time for which the **OCS_COMP1** and **OCS_COMP2** inputs are debounced. This field has units of 100 us. **Note:** This register shall not be modified while the OCS debouncer is enabled.	R/W	0Fh

8.5.38 IRQ_N PU/PD CONTROL REGISTER (IRQ_PUPD)

Address: 009Bh
Size: 8 bits

This register controls the PU and PU for the **IRQ_N** pin.

Bits	Description	Type	Default
7	**PullUpEnable** 0: No pull-up 1: Pull-up enabled.	R/W	0b
6	**PullDownEnable** 0: No pull-down 1: Pull-down enabled.	R/W	0b
5:0	**RESERVED**	RO	-

8.5.39 VOLTAGE REGULATOR CONTROL REGISTER (VREG_CTL)

Address: 009Ch Size: 8 bits

Bits	Description	Type	Default
7:1	**RESERVED**	RO	-
1	**VREG_LOW_OVR** When set, the 1.8V voltage regulator is placed in low power mode whenever both the Ring Oscillator and 48MHz Relaxation Oscillator are disabled. **Note:** It is expected tor this bit to be set in CPU companion mode. **Note:** In standalone modes, the respective OTP bit shall be set to enable this feature and to ensure maximum power savings.	R/W	0b
0	**VREG Low Power Enable (VREG_LOW_EN)** When set, the 1.8V voltage regulator is placed in low power mode. **Note:** This bit has no meaning when the VREG_LOW_OVR bit is set.	R/W	0b

8.5.40 I2C TRIM REGISTER (I2C_TRIM)

Address: 009Dh Size: 8 bits

BITS	DESCRIPTION	TYPE	DEFAULT
7:0	**I2C_TRIM** This register specifies the delay added by the I2C controller in order to meet the tHD_DAT parameter. This register has units of 20.8 ns.	R/W	9h

Note 8-6 The default is defined by the respective OTP field. If the OTP is not programmed the default value is 9h.

8.5.41 GENERAL PURPOSE TIMER LOAD REGISTER (GP_TIMER_LOAD)

Address: 00A0h Size: 16 bits

BITS	DESCRIPTION	TYPE	DEFAULT
15:0	**GP Timer Load** This register defines the value to be loaded into the general purpose timer from which it will count down. The valid range for this register is 0001h - FFFFh. This field has units of 1 ms. **Note:** Programming 0h into this register is not valid. **Note:** Software shall not change the contents of this register while the RESET bit is set and shall wait until this bit is cleared.	R/W	FFFFh

8.5.42 GENERAL PURPOSE TIMER INTERRUPT SOURCE REGISTER (GP_TIMER_INT_SRC)

Address: 00A2h Size: 8 bits

BITS	DESCRIPTION	TYPE	DEFAULT
7:1	**RESERVED**	RO	-
0	**GP_UFLOW** This bit will be set when the general purpose timer underflows. The GP_TIMER_INT interrupt in Interrupt Status Register (INT_STS) persists until this bit is cleared. When DIS_ON_UFLOW bit of the General Purpose Timer Control Register (GP_TIMER_CTL) is set, the assertion of this bit is delayed until the timer is fully disabled (100s us).	R/WC	0b

8.5.43 GENERAL PURPOSE TIMER CONTROL REGISTER (GP_TIMER_CTL)

Address: 00A3h Size: 8 bits

BITS	DESCRIPTION	TYPE	DEFAULT
7	**STATUS** This bit indicates the operational state of the GP timer. 0: Timer is disabled and not counting 1: Timer is enabled and counting When this bit is cleared the timer has halted operation.	RO	0b
6:3	**RESERVED**	RO	-
2	**DIS_ON_UFLOW** When this bit is set, the GP timer is automatically disabled upon detection of the underflow condition. Additionally, the ENABLE bit is cleared upon underflow detection and the assertion of GP_UFLOW will be delayed until the GP Timer has halted operation and the STATUS bit is cleared.	R/W	0b
1	**RESET** Timer Reset. When set this bit stops the timer and resets the internal counter to the value in the General Purpose Timer Load Register (GP_TIMER_LOAD). 0: Timer is not in reset 1: Timer is in reset **Note:** This bit does not clear until propagation of the reset completes. This may take 100s us.	R/SC	0b
0	**ENABLE** Timer Enable. This bit is used to start and stop the General Purpose Timer Load Register (GP_TIMER_LOAD). This bit does not reset the timer count. This is automatically cleared when the DIS_ON_UFLOW bit is set. 0: Timer is disabled 1: Timer is enabled	R/W	0b

8.5.44 GENERAL PURPOSE TIMER COUNT REGISTER (GP_TIMER_COUNT)

Address: 00A4h Size: 16 bits

BITS	DESCRIPTION	TYPE	DEFAULT
15:0	**GP Timer Counter** This register returns the current value of the general purpose timer. The GP timer must be disabled before reading this register by clearing the ENABLE bit.	RO	0h

9.0 CABLE PLUG ORIENTATION AND DETECTION

This section details the functions that control and monitor the CC pins, monitor the **VBUS_DET** pin, control the VCONN FETs, and sample the **CFG_SEL** pin.

9.1 CC Comparator

The device integrates a comparator and DAC circuit to implement Type-C attach and detach functions. It supports up to eight programmable thresholds for attach detection between UFP and DFP. When operating as a UFP, the device supports detecting changes in the DFP's advertised thresholds to determine current sourcing capability. The default nominal values for the thresholds detected by the CC comparators are:

- 0.20 V
- 0.40 V
- 0.66 V
- 0.80 V
- 1.23 V
- 1.60 V
- 2.60 V
- 3.0 V Proprietary Mode

TABLE 9-1: CABLE DETECT SUMMARY

Parameter	Threshold CSR	Description	Min	Typ	Max
DFP_ACT_DEF	CC_THR0	Detecting an active cable when configured as DFP and advertising default USB current.		0.20 V	
UFP_DFP_DEF	CC_THR0	Detecting DFP attach when configured as UFP and DFP is advertising default USB current.		0.20 V	
DFP_ACT_1A5	CC_THR1	Detecting an active cable when configured as DFP and advertising 1.5A.		0.40 V	
UFP_DFP_1A5	CC_THR2	Detecting DFP attach when configured as UFP and DFP is advertising 1.5A.		0.66 V	
DFP_ACT_3A0	CC_THR3	Detecting an active cable when configured as DFP and advertising 3.0A.		0.80 V	
UFP_DFP_3A0	CC_THR4	Detecting DFP attach when configured as UFP and DFP is advertising 3.0A.		1.23 V	
DFP_UFP_DEF	CC_THR5	Detecting UFP attach when configured as DFP advertising default USB current.		1.60 V	
DFP_UFP_1A5	CC_THR5	Detecting UFP attach when configured as DFP advertising 1.5A.		1.60 V	
DFP_UFP_3A0	CC_THR6	Detecting UFP attach when configured as DFP advertising 3.0A.		2.60 V	

The following tables summarize the expected thresholds to be matched in the CCx Match Registers (CCx_MATCH)CCx Match Registers (CCx_MATCH) for various configurations. Refer to the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) for the debouncer mask used in these configurations.

TABLE 9-2: DFP CC MATCH SUMMARY

CC State	CC THR0	CC THR1	CC THR2	CC THR3	CC THR4	CC THR5	CC THR6	CC THR7
Advertise Default USB Current and connected to powered cable	0	0	0	0	0	0	0	0
Advertise 1.5 A and connected to powered cable	0	0	0	0	0	0	0	0
Advertise 3.0 A and connected to powered cable	0	0	0	0	0	0	0	0
Advertise Default USB Current and connected to UFP	1	0	0	0	0	0	0	0
Advertise 1.5 A and connected to UFP	0	1	0	0	0	0	0	0
Advertise 3.0 A and connected to UFP	0	0	0	1	0	0	0	0
Advertise Default USB Current and no connect (vOpen)	1	0	0	0	0	1	0	0
Advertise 1.5 A and no connect (vOpen)	0	1	0	0	0	1	0	0
Advertise 3.0 A and no connect (vOpen)	0	0	0	1	0	0	1	0
Proprietary Mode and no connect (vOpen)	0	0	0	0	0	0	0	1

TABLE 9-3: UFP CC MATCH SUMMARY

CC State	CC THR0	CC_ THR1	CC THR2	CC THR3	CC THR4	CC THR5	CC THR6	CC THR7
Powered cable detected.	0	0	0	0	0	0	0	0
No Connect (SNK.Open)	0	0	0	0	0	0	0	0
DFP Connected and advertising default USB current	1	0	0	0	0	0	0	0
DFP Connected and advertising 1.5 A	1	0	1	0	0	0	0	0
DFP Connected and advertising 3.0 A	1	0	1	0	1	0	0	0
DFP Connected and advertising proprietary current	1	0	1	0	1	0	0	1

The following examples illustrate the rationale behind the expected matches in the previous tables.

DFP: Advertise 1.5 A and connected to powered cable, UFP not attached

In this scenario the DFP is advertising 1.5 A and connected to a powered cable. The UFP is not attached. The vRd value measured will be less than 0.35V per specification.

In this configuration the following thresholds are enabled for matching:

- CC_THR1 (DFP_ACT_1A5)
- CC_THR5 (DFP_UFP_1A5)

The value programmed into CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) and CCx Match Enable Registers (CCx_MATCH_EN) would therefore be 0x22.

Since the vRd is less than 0.35V no threshold matches would be detected on CCx Match Registers (CCx_MATCH) or CCx Change Status Registers (CCx_CHG_STS).

DFP: Advertise 3.0 A and connected to UFP

In this scenario the DFP is advertising 3.0 A and connected to a UFP. The vRd value measured will be > 0.85V and < 2.45V.

In this configuration the following thresholds are enabled for matching:

- CC_THR3 (DFP_ACT_3A0)
- CC_THR6 (DFP_UFP_3A0)

The value programmed into CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) and CCx Match Enable Registers (CCx_MATCH_EN) would therefore be 0x44.

In this case only CC_THR3 would be matched. Bit 3 would be set in both CCx Match Registers (CCx_MATCH) or CCx Change Status Registers (CCx_CHG_STS) after the debounce interval.

UFP: Connected to DFP advertising 3.0 A

In this case the UFP is connected to a DFP advertising 3.0A. The vRd value measured will be > 1.3V and < 2.04 V.

In this configuration the following thresholds are enabled for matching:

- UFP_DFP_DEF (CC_THR0)
- UFP_DFP_1A5 (CC_THR2)
- UFP_DFP_3A0 (CC_THR4)

The value programmed into CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) and CCx Match Enable Registers (CCx_MATCH_EN) would therefore be 0x15.

In this case all three thresholds would be matched since the value of vRd must exceed CC_THR4 which is 1.23V. Bits 0, 2 and 4 would be set in both CCx Match Registers (CCx_MATCH) or CCx Change Status Registers (CCx_CHG_STS).

9.2 DFP Operation

The device implements current sources to advertise current charging capabilities on both CC pins when operating as a DFP.

When a UFP connection is established, the current driven across the CC pins creates a voltage across the UFP's Rd pull-down that can be detected by the integrated CC comparator. The voltages monitored are summarized in Table 9-4. When connected to an active cable, an alternative pull-down (Ra) appears on the CC pin.

The DFP also integrates two 5V FETs for implementing the VCONN function. This is further discussed in Section 9.8, "VCONN Operation".

TABLE 9-4: SOURCE DETECTION

CC1	CC2	Connection State	CC Comparator State	VBUS	VCONN
Open	Open	Nothing Attached	Monitor both CC pins for attach	Off	Off
Rd	Open	UFP Attached	Monitor **CC1** for detach	On	Off
Open	Rd	UFP Attached	Monitor **CC2** for detach	On	Off
Ra	Open	Powered Cable, No UFP attached	Monitor **CC2** for UFP attach. Monitor **CC1** for cable detach.	Off	Off
Open	Ra	Powered Cable, No UFP attached	Monitor **CC1** for UFP attach. Monitor **CC2** for cable detach.	Off	Off
Ra	Rd	Powered Cable, UFP attached	Monitor **CC2** for UFP detach. **CC1** is not monitored for detach.	On	On
Rd	Ra	Powered Cable, UFP attached	Monitor **CC1** for UFP detach. **CC2** is not monitored for detach.	On	On
Rd	Rd	Debug accessory mode attached	Monitor both CC pins for detach	Off	Off
Ra	Ra	Audio accessory mode attached.	Monitor both CC pins for detach	Off	Off

9.2.1 RP CURRENT SOURCES

In order to advertise the current charging capabilities of the device via the integrated port power controller or external power circuit, Rp current sources are used. The current source can be selected by software. using CC1 RP Value and CC2 RP Value in the CC Control Register (CC_CTL). Table 9-5 summarizes the values supported by the current sources in regards to the programmed value.

TABLE 9-5: RP CURRENT SOURCES

DFP Advertisement	Current source (1.7V to 5.5V)	RPx Value
Disabled		00b
Default USB Power	80 uA +/-20%	01b
1.5A @ 5V	180 uA +/-8%	10b
3.0A @ 5V	330 uA +/-8%	11b

The current source coupled with the CC pins for RP advertisement is also used for sampling the **CFG_SEL** pin. When the **CFG_SEL** pin is sampled, current is steered away from the CC pins and no RP value is advertised. See the VBUS Comparator Control field in VBUS Control Register (VBUS_CTL) for further details.

9.2.2 SOURCE ATTACH DETECTION

When configured as a Source, the following sections describe the steps that are taken to determine if an attach has occurred:

1. Software accesses the device via I^2C/SPI which wakes the device up and enables the 48 MHz oscillator.
2. Software programs the CC Comparator Control to 00b and disables the CC debouncer.
3. Software polls CC_DB_ACTIVE until it reads back to indicate the CC debouncer is inactive.
4. Software programs the Match Debounce Register (MATCH_DEB) as required for the tPDDebounce period.
5. Software programs the VBUS Debounce Register (VBUS_DEB) as required.
6. Software programs the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) and CCx Match Enable Registers (CCx_MATCH_EN) as required to match the thresholds of interest.
7. Software programs the CCx Sample Enable Registers (CCx_SAMP_EN) and VBUS Match Enable Register (VBUS_MATCH_EN) to detect VBUS VSafe0v Match (VSAFE0V_THR_MATCH).
8. Software enables the CC_MATCH_VLD, CC1_MATCH_CHG and CC2_MATCH_CHG interrupt via CC Interrupt Enable Register (CC_INT_EN).
9. Software enables the VBUS_MATCH_VLD interrupt via the Power Interrupt Enable Register (PWR_INT_EN).

Note: Assertion of CC_MATCH_VLD indicates that the initial CC debounce of both CC pins has completed and a valid value is available in the CCx Match Registers (CCx_MATCH). VBUS_MATCH_VLD indicates a valid value is available in the VBUS Match Register (VBUS_MATCH).

10. Software enables **IRQ_N** assertion by enabling the CC_INT, VBUS_INT and PWR_INT in the Interrupt Enable Register (INT_EN).
11. Software programs the RP current sources via CC1 RP Value and CC2 RP Value in the CC Control Register (CC_CTL).
12. Software programs the CC Threshold x Registers (CC_THRx), if required.
13. Software programs the CC Comparator Control to sample both CC pins and enables the CC debouncer. The CC_DB_ACTIVE bit will assert soon after.
14. Software sets the Clock Control Register (CLK_CTL) to disable all clock sources accept for the keep alive clock.
15. When a UFP is attached an Rd pull-down, or Ra pull-down from active cable, is connected to one of the CC pins. The CC comparator detects this.
16. Changes in state of the CC pin are recorded in the CCx Match Registers (CCx_MATCH) and CCx Change Status Registers (CCx_CHG_STS) after the programmed debounce period.
17. CC_MATCH_VLD and CC1_MATCH_CHG or CC2_MATCH_CHG interrupt assert and CC_INT asserts which in turn asserts **IRQ_N** pin. Likewise assertion of VBUS_MATCH_VLD causes PWR_INT to assert and also in turn asserts **IRQ_N**.
18. Software implements a further debounce of the CC match for tCCDebounce in order to detect the attachment.
19. Software must verify, after VBUS_MATCH_VLD assertion, that the VBUS Match Register (VBUS_MATCH) indicates vSafe0v is on VBUS.
20. After an attachment, software programs the device to power VBUS via the integrated PPC, if required. If the power source is external to the device, software configures that source as required.
21. In the event that an active cable is attached, per the CCx Match Registers (CCx_MATCH), VCONN power may be supplied by appropriately setting VCONN1 Control or VCONN2 Control in the VBUS Control Register (VBUS_CTL).
22. CC Communication Select in the CC Control Register (CC_CTL) is set by software to appropriately connect the baseband interface to the CC pin with the Rd pull-down.
23. The DFP may attempt to communicate with attached device utilizing the PD MAC if desired.

9.2.3 SOURCE DETACH DETECTION

When configured as a source, the detachment of the partner UFP is determined by monitoring the appropriate CC pin (with the Rd pull-down) for a voltage exceeding DFP_UFP_DEF, DFP_UFP_1A5 or DFP_UFP_3A0, depending on the charging current advertised by the device.

The following describes the steps that are taken to determine if a detach has occurred when operating in companion mode:

APPLICATION NOTE: Software must not look at the CC detect status while a PD contract is in place, even though the Type-C spec permits doing so, in order to avoid false-disconnects in the Attached.SNK state during heavy PD message traffic.

APPLICATION NOTE: Software should set a 3A advertisement while it's a source and a PD contract is in place in order to avoid false-disconnects in the Attached.SRC state during heavy PD message traffic; a 1.5A advertisement (even though permitted by the Type-C spec) is not sufficient to avoid this issue due to the lower vOPEN threshold. An exception to this recommendation is when “Collision Avoidance” is implemented.

The following steps assume the CC debouncer has previously been used to detect attachment per the prior section.

1. Software accesses the device via I^2C/SPI which wakes the device up and enables the 48 MHz oscillator.
2. Software programs the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) as required to enable per threshold debouncing. Typically this only involves disabling the thresholds not corresponding to vOPEN.
3. Software enables CC1_MATCH_CHG or CC2_MATCH_CHG interrupt via CC Interrupt Enable Register (CC_INT_EN).
4. Software enables **IRQ_N** via asserting the respective CC_INT bit in the Interrupt Enable Register (INT_EN).
5. Software programs the CC Comparator Control to sample appropriate CC pin.
6. The device samples the respective CC pin while operating off of 20 kHz clock.
7. Changes in state of the CC are reflected in the respective CCx Match Registers (CCx_MATCH) and CCx Change Status Registers (CCx_CHG_STS).
8. CC1_MATCH_CHG, or CC2_MATCH_CHG, interrupt asserts, CC_INT asserts which in turn asserts the **IRQ_N** pin.
9. Software implements a further debounce of the CC match for tCCDebounce in order to detect the detachment.
10. If VCONN is being supplied, than the VCONN FET shall be disabled in software by appropriately setting the VCONN1 Control or VCONN2 Control in the VBUS Control Register (VBUS_CTL). In this case, software must also discharge VCONN as defined in Section 9.8.1, "VCONN Discharge Programming Model".
11. Software disables the PPC which causes the internal 5V power switch to open if required. Power may have alternatively been provided by external power source.
12. Internal 100 Ohm VBUS discharge switch is closed if required.
13. Discharge switch remains closed until vSafe0V threshold is crossed.
14. If vSafe0V threshold is not crossed within VBUS Off Register (VBUS_OFF) which asserts VBUS Discharge Error interrupt and asserts **IRQ_N** pin.

Note:	Occurrence of VBUS Discharge Error interrupt indicates a potentially catastrophic system power issue.

15. Internal discharge switch is opened.
16. The part is configured by software to detect a UFP attach. The 48 MHz oscillator is disabled. Only the keep-alive clock remains enabled.

9.3 UFP Operation

When operating as a UFP, the device applies an Rd pull-down on both CC lines and waits for a DFP connection from the assertion of VBUS. The CC comparator is used to determine the advertised current charger capabilities supported by the DFP. The supported Rd values are defined in Table 9-6.

TABLE 9-6: RD TERMINATION

Rd Implementation	Min	Typ	Max
Rd (Trimmed)	***TBD***	5.1 kOhm	***TBD***
Rd (Dead Battery)	4.1 kOhm	5.1 kOhm	6.1 kOhm
Threshold Voltage	***TBD***	0.9 V	***TBD***

9.3.1 SINK ATTACH DETECTION

The following steps illustrate how the device may be programmed to detect an attachment when operating as a Sink. The following discussion does not cover dead-battery cases, which are described in Section 9.7.3, "Dead Battery (UPD350-A/UPD350-B Only)".

1. Software accesses the device via I^2C/SPI which wakes the device up and enables the 48 MHz oscillator.
2. Software programs the CC Comparator Control to 00b and disables the CC debouncer.
3. Software programs the VBUS Comparator Control to 00b and disables the VBUS debouncer.
4. Software polls CC_DB_ACTIVE and VBUS_DB_ACTIVE until they read back 0b to indicate that the CC debouncer and VBUS debouncer are inactive. Software programs the Match Debounce Register (MATCH_DEB) as required for the tPDDebounce interval.
5. Software programs the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) as required to match the thresholds of interest.
6. Software programs the CC1 Pull-Down Value and CC2 Pull-Down Value in the CC Control Register (CC_CTL) to advertise trimmed Rd.
7. Software enables CC_MATCH_VLD, CC1_MATCH_CHG or CC2_MATCH_CHG interrupt via the CC Interrupt Enable Register (CC_INT_EN).
8. Assertion of CC_MATCH_VLD indicates that the initial CC debounce of both CC pins has completed and a valid value is available in the CCx Match Registers (CCx_MATCH).
9. Software programs the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) as required to enable the respective threshold(s) debouncing.
10. Software enables VBUS_INT to detect VBUS by configuring the VBUS Match Register (VBUS_MATCH).
11. Software enables VBUS_MATCH_VLD by configuring the Power Interrupt Enable Register (PWR_INT_EN).
12. Software enables **IRQ_N** via configured respective bits, for CC_INT, PWR_INT and VBUS_INT, in the Interrupt Enable Register (INT_EN).
13. Software programs the CC Comparator Control to sample both CC pins.
14. Software programs VBUS Comparator/DAC to detect vSafe5V via the VBUS Comparator Control in the VBUS Control Register (VBUS_CTL) and setting VBUS_THR0 and VBUS_THR1 via the VBUS Threshold x Registers (VBUS_THRx).
15. Software turns off all clock sources with the exception of 20 KHz keep-alive clock via Clock Control Register (CLK_CTL) to save power.
16. Upon connection to a partner DFP VBUS is powered to 5V.
17. After the programmed debounce interval, the respective CCx Change Status Registers (CCx_CHG_STS) is updated and the respective CC1_MATCH_CHG or CC2_MATCH_CHG interrupt asserts which in turn asserts CC_INT which in turn asserts **IRQ_N**.

> **Note:** Assertion of CC_MATCH_VLD indicates that the initial CC debounce of both CC pins has completed and a valid value is available in the CCx Match Registers (CCx_MATCH).

18. VBUS is detected by the VBUS_DET comparator and debounced for the period defined by the VBUS Debounce Register (VBUS_DEB). The VBUS_INT interrupt asserts which in turn asserts IRQ_N.

> **Note:** Assertion of VBUS_MATCH_VLD indicates that the initial CC debounce of both CC pins has completed and a valid value is available in the VBUS Match Register (VBUS_MATCH).

19. Software must debounce the CC match for tCCDebounce and VBUS for the PD Debounce Register (PD_DEB)

in order to detect the attach condition.

20. After attachment, software configures CC Communication Select in the CC Control Register (CC_CTL) to appropriately connect the baseband interface to the CC with the Rp pull-up if PD communication is desired.
21. DFP may communicate with the device.

9.3.2 SINK DETACH DETECTION

The DFP detach is detected by the removal of VBUS. The VBUS comparator must always be enabled for the UFP to detect this condition.

1. Software programs the VBUS Comparator Control to 00b and disables the VBUS debouncer.
2. Software polls VBUS_DB_ACTIVE until it reads back 0b to indicate that the VBUS debouncer is inactive.
3. Software programs the VBUS Debounce Register (VBUS_DEB) as required.
4. Software programs the CCx Sample Enable Registers (CCx_SAMP_EN) as required to enable the respective threshold(s) debouncing.
5. Software enables VBUS_INT, which is then used to detect VBUS, via the VBUS Match Enable Register (VBUS_MATCH_EN).
6. Software enables **IRQ_N** by configuring the respective bit for VBUS_INT in the Interrupt Enable Register (INT_EN).

Note:	VBUS_THR0 and VBUS_THR1 are adjusted via the VBUS Threshold x Registers (VBUS_THRx) to vSafe5V, if required.

7. Software enables the VBUS Comparator/DAC via the VBUS Comparator Control bit in the VBUS Control Register (VBUS_CTL).
8. Software turns off all clock sources with the exception of 20 KHz keep-alive clock via the Clock Control Register (CLK_CTL) to save power.
9. DFP removes VBUS.
10. After the VBUS debounce period, the VBUS Match Register (VBUS_MATCH) and VBUS Change Status Register (VBUS_CHG_STS) are updated. The VBUS_INT interrupt asserts which in turn asserts **IRQ_N**.
11. Software continues to monitor VBUS to determine if it stays below vSafe5V for a duration of tPdDebounce.

9.4 DRP Operation (Legacy)

This section describes a usage of the device for implementing a DRP attach. In this configuration, software utilizes the device to alternate between a Source and Sink advertisement with an interval of tDRP per the USB Type-C® Specification.

The steps for initially configuring the device to advertise Source capabilities follows the steps defined in Section 9.2.2, "Source Attach Detection". Software must also implement the tDRP timer. If a Sink is not detected within this time, software shall change the device's role to Sink and again attempt an attach detection.

1. Software accesses the device via I^2C/SPI which wakes the device up and enables the 48 MHz oscillator.
2. Software programs the CC Comparator Control to 00b and disables the CC debouncer.
3. Software polls CC_DB_ACTIVE until it reads back to indicate the CC debouncer is inactive.
4. Software programs the Match Debounce Register (MATCH_DEB) as required for the tPDDebounce period.
5. Software programs the CC Threshold x Registers (CC_THRx) if required.
6. Software programs the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) and CCx Match Enable Registers (CCx_MATCH_EN) as required to match the thresholds of interest.
7. Software enables the CC_MATCH_VLD, CC1_MATCH_CHG and CC2_MATCH_CHG interrupts via the CC Interrupt Enable Register (CC_INT_EN).
8. Software enables **IRQ_N** assertion by enabling the CC_INT in the Interrupt Enable Register (INT_EN).
9. Software programs the RP current sources via the CC1 RP Value and CC2 RP Value in the CC Control Register (CC_CTL).
10. Software programs the CC Comparator Control to sample both CC pins and enable the CC debouncer. The CC_DB_ACTIVE bit will assert soon after.
11. Software sets the Clock Control Register (CLK_CTL) to disable all clock sources accept for the keep alive clock.

Note:	Assertion of CC_MATCH_VLD indicates that the initial CC debounce of both CC pins has completed and a valid value is available in the CCx Match Registers (CCx_MATCH).

If after time tDRP an attachment is not detected, software shall configure the device to be a Sink and attempt to detect the presence of a Source. This is similar to the steps defined in Section 9.3.1, "Sink Attach Detection".

1. Software accesses the device via I2C/SPI which wakes the device up and enables the 48 MHz oscillator
2. Software programs the CC Comparator Control to 00b and disables the CC debouncer.
3. Software polls CC_DB_ACTIVE until it reads back 0b to indicate that the CC debouncer is inactive.
4. Software programs the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) as required to match the thresholds of interest.
5. Software programs detection for vSafe5V by setting VBUS_THR0 and VBUS_THR1 in the VBUS Threshold x Registers (VBUS_THRx). The CCx Sample Enable Registers (CCx_SAMP_EN) is programmed debounce these thresholds.
6. Software programs CC1 Pull-Down Value and CC2 Pull-Down Value in the CC Control Register (CC_CTL) to advertise trimmed Rd.
7. Software enables the CC_MATCH_VLD, CC1_MATCH_CHG or CC2_MATCH_CHG interrupts via the CC Interrupt Enable Register (CC_INT_EN).
8. Software enables VBUS_MATCH_VLD by configuring the Power Interrupt Enable Register (PWR_INT_EN).
9. Software enables **IRQ_N** via the configured respective bits, for CC_INT, PWR_INT and VBUS_INT, in the Interrupt Enable Register (INT_EN).
10. Software programs the CC Comparator Control to sample both CC pins.
11. Software turns off all clock sources with the exception of 20 KHz keep-alive clock via the Clock Control Register (CLK_CTL) to save power.

If after time tDRP an attachment is not detected, software shall configure the device to be a Source and attempt to detect the presence of a Sink.

Note:	Assertion of CC_MATCH_VLD indicates that the initial CC debounce of both CC pins has completed and a valid value is available in CCx Match Registers (CCx_MATCH).

Note:	Assertion of VBUS_MATCH_VLD indicates that the initial CC debounce of both CC pins has completed and a valid value is available in VBUS Match Register (VBUS_MATCH).

9.5 DRP Offload

DRP offload enables the device to manage the DRP toggle. This is beneficial as it allows the host CPU to remain in a low power state until a connection is detected.

DRP offload toggles between Source and Sink advertisement by alternating between enabling Rp current sources and Rd pull-downs for a period of tDRP (DRP Time Register). The duty cycle between Source and Sink advertisement is determined by the DRP Duty Cycle Register. The DRP Time Register may be written by firmware or generated automatically via a pseudo random number generator. The latter approach should be used to reduce the probability of collisions when connecting. It is selectable whether the DRP cycle shall first advertise UFP or DFP via DRP Initial State (DRP_INIT) bit in DRP Control Register.

A connection is detected after a successful debounce for a period defined by the Match Debounce Register. VBUS is checked to be below vSafe0v for the DFP case. This results in **IRQ_N** assertion and automatic disablement of the DRP toggle. Firmware must further debounce for the period tPDDebounce before determining if a valid match is present. If a match has not occurred, firmware shall enable DRP again.

A pseudo random number generator, implemented via a LFSR, is utilized to generate the DRP period. The 16-bit seed is defined by DRP LFSR Seed Register. The LFSR operates off of the 20 KHz clock and updates every 100 us when enabled. LFSR Enable (LFSR_EN) in DRP Control Register is set.

Hardware limits the total DRP period to be between 50 ms and 100 ms in order to comply with the USB Type-C® specification.

9.6 Collision Avoidance

An alternative mode of operation is required to enable the CC detection circuit to facilitate software implementation of collision detection which was incorporated into version 3.0 of the USB PD Specification.

In order to avoid message collisions due to asynchronous Messaging (AMS) sent from the Sink, the Source sets Rp to SinkTxOk (3A@5V) to indicate to the Sink that it is OK to initiate an AMS. When the Source wishes to initiate an AMS it sets Rp to SinkTxNG (1.5A@5V). When the Sink detects that Rp is set to SinkTxOk it may initiate an AMS. When the Sink detects that Rp is set to SinkTxNG it shall not initiate an AMS and shall only send Messages that are part of an AMS the Source has initiated.

When operating as a Sink, a mechanism is required for quickly determining whether the Source is advertising SinkTxNG or SinkTxOK on Rp.

A collision avoidance mechanism exists to enable software to instruct the device to sample only a single threshold on a single CC pin. This results in a cycle through both thresholds taking only 100 us, making it easier for software to meet the timing constraints mandated by SinkTxOk in the specification. Two CCx Sample Enable Registers (CCx_SAMP_EN) are provided to enable software to specify which subsets of the CC thresholds shall be sampled via setting the respective bit. In the case of collision avoidance, only the threshold corresponding to SinkTxOK shall be set on the connected CC pin. This corresponds to UFP_DFP_3A0 (CC_THR4). The CC Comparator Control field in CC Control Register (CC_CTL) shall be set to the CC pin utilized for PD communication.

In order to prevent false positive detection of SinkTXOK when a PD packet is being transmitted or received, a debounce value in the order to 7 ms would need to be programmed into Match Debounce Register (MATCH_DEB). Such a large value would impair the ability of software, when operating as Sink, to detect SinkTXOK and transmit a PD message within the time defined in the PD specification by SinkTxTime. The BLK_PD_MSG bit in CC Hardware Control Register (CC_HW_CTL) has been incorporated to handle the above scenario and enable software to program a sub millisecond debounce value. A minimum value of 100 us shall be used for the Match Debounce Register (MATCH_DEB) when using this mode of operation.

The below sequence illustrates the steps to enable Collision Avoidance for a Sink. This sequence is used after a connection and PD contract has been established.

Note:	The units of the Match Debounce Register (MATCH_DEB) is determined by MATCH_DB_UNITS bit in CC Hardware Control Register (CC_HW_CTL). When this bit is clear, the units are 1.6 ms. When this bit is set the units are 100 us.

1. Sink establishes connection with Source and a PD contract is negotiated. Sink wants to transmit AMS to Source.
2. Via the CC Hardware Control Register (CC_HW_CTL), the MATCH_DB_UNITS is set to 1b to utilize 100 us units for the CC match debouncer. The BLK_PD_MSG bit is set to filter PD traffic from the debouncer.
3. Software sets the Match Debounce Register (MATCH_DEB) to 400 us by writing four into the register.
4. Software configures CC_THR4 to enable interrupt assertion via the CCx Match Enable Registers (CCx_MATCH_EN).
5. Software configures CC_THR4 for sampling via the CCx Sample Enable Registers (CCx_SAMP_EN).
6. Via the CC Control Register (CC_CTL), the CC Comparator Control selects the CC pin connected to the Source's Rp.
7. Software waits for indication that Source's Rp is SinkTxOK via a match detected on CC_THR4.
8. Software initiates transmission of PD message to Source.

APPLICATION NOTE: Software must guarantee that a sufficient gap exists in between consecutively transmitted PD messages to enable the source RP value to settle on the CC line. Studies by USB-IF members have showed the settling time may take in excess of 40 us.

9.7 Fast Role Swap (FRS)

This feature is used to detect when a partner source has lost power. Upon detection of FRS signaling, the “Old Sink” transitions to be a Source and begins supplying VBUS.

When operating as a Sink, the FRS mode of operation enables detection of FRS signaling. Detection results in **IRQ_N** assertion and this event may also be mapped as a PIO override source. When operating as a Source, upon detection of loss of power, the device will transmit FRS signaling. This is initiated by either assertion of a selected PIO or a CSR write.

The following FRS related features are supported:

- Ability to detect reception of FRS signaling
- High bandwidth and current boost mode for CC comparator to increase sampling frequency
- Interrupt, and PIO, assertion upon FRS detection
- PIO override support for FRS detection as a source
- Ability to initiate FRS signaling via GPIO assertion or register write.
- Control 5 Ohm (Rsw) pull-down resistor

9.7.1 FRS SINK OPERATION

When operating as a Sink, the device is configured to detect FRS signaling by setting FRS Detect Enable (FRS_DET_EN) in FRS Control Register. The CC detection logic is programmed to detect three thresholds (SinkTxOK, SinkTXNG, FRSWAP) and samples each threshold in round robin fashion. The sampling rate is determined by CC Sample Clock Register.

When a match is detected on FRSWAP threshold, the CC detection logic will “park” at this threshold and continue monitoring the output of the comparator. While “parked” the output of the CC comparator will be sampled at an increased rate of 12 MHz. This higher sampling rate will prevent PD messages from inadvertently looking like FRS signaling which will happen on occasion when the sampling rate is similar or slower than the ~270 kbps rate of PD messages.

It will continue debouncing for the amount of time specified in FRS CC Debounce Register. The FRSWAP threshold is indicated by FRS Threshold Select Register.

If the debounce is successful, then the FRS_RCV_STS interrupt is asserted and the CC detection logic resumes sampling all enabled thresholds. If the FRS debounce fails, the CC detection logic resumes sampling all enabled thresholds.

In this mode of operation the CC Comparator operates at a faster rate in order to minimize the FRS detection latency. This is enabled by placing this the comparator into a high bandwidth mode. The CC High Bandwidth Mode Enable (CC_DET_HBW_EN) in CC Hardware Control Register must be set.

After detecting the FRS signaling, the “old Sink” must start supplying vSafe5V at USB Type-C® current VBUS no later than tSrcFRSwap (150 us) after VBUS has dropped below vSafe5V. This must be accomplished via circuitry external to the device.

Note:	The upper threshold used for vSafe5V should be used for determining when VBUS has dropped below vSafe5v to help meet tSrcFRSwap requirement.

Note:	Matching of a VBUS threshold may be selected as a source to a PIO override. Additionally, VBUS threshold match ANDed with FRS signal detect may also be used as a PIO override source.

9.7.2 FRS SOURCE OPERATION

The initial Source shall signal a FRS request by driving the CC pin to ground with a resistance of less than 5 Ohms for a period defined by FRS Transmission Length Register. The FRS request signaling is initiated by either a CSR write or GPIO assertion.

The former case is implemented by setting the FRS Request (FRS_REQ_EN) bit in FRS Control Register. This bit self clears after the FRS request is transmitted. For the latter case, the PIO is selected by the FRS Request PIO (FRS_REQ_PIO) field in FRS Control Register.

Transmission of FRS signaling will take precedence over PD MAC TX communication. The FRS PD resistor is enabled in tandem on the CC pin determined by FRS CC Select (FRS_CC_SEL) in FRS Control Register. This configuration remains until FRS transmission has completed.

9.7.3 DEAD BATTERY *(UPD350-A/UPD350-B Only)*

Two variations of the Rd resistor are implemented as detailed in Table 9-6: Rd (Dead Battery) and Rd (Trimmed). The **CC1_DB_EN** and **CC2_DB_EN** pins exist to determine the operation of the CC pins in dead battery conditions and are to be connected externally via the PCB to the respective CC pin. The CC pins are configured to present either Hi-Z or an untrimmed Rd pull-down resistance when connected to a DFP advertising a pull-up resistance.

Figure 9-1 illustrates the configuration for supporting dead battery cases via hair pining **CC*x*_DB_EN** and **CC*x*** together on the PCB. The UFP pull-up activates the FET in series with RD_DB and enables the untrimmed dead battery pull-down.

FIGURE 9-1: CC RD (DEAD BATTERY)

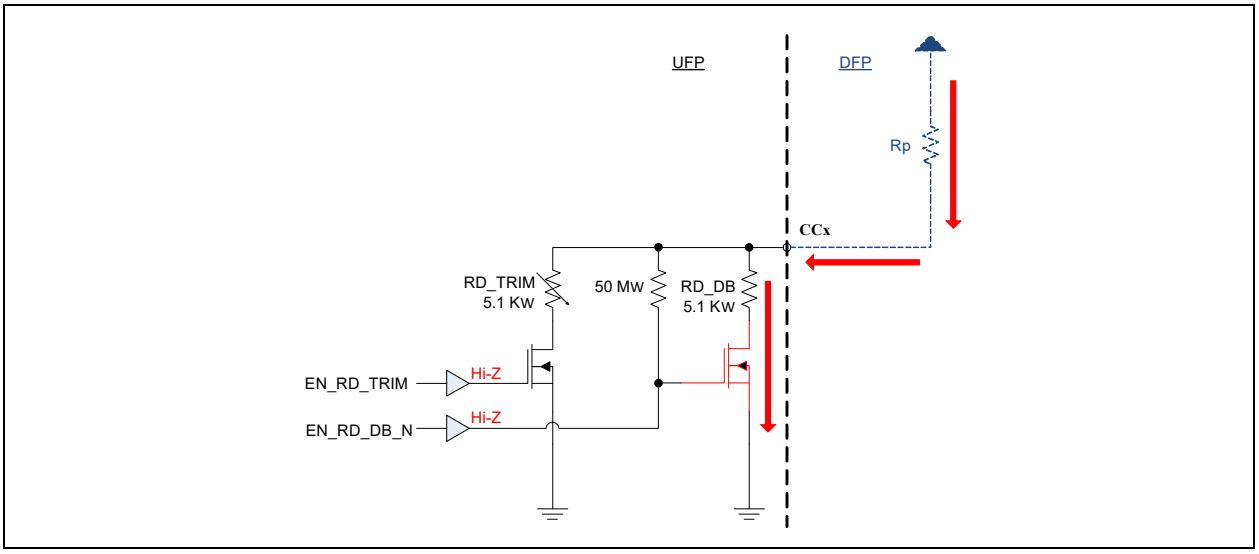
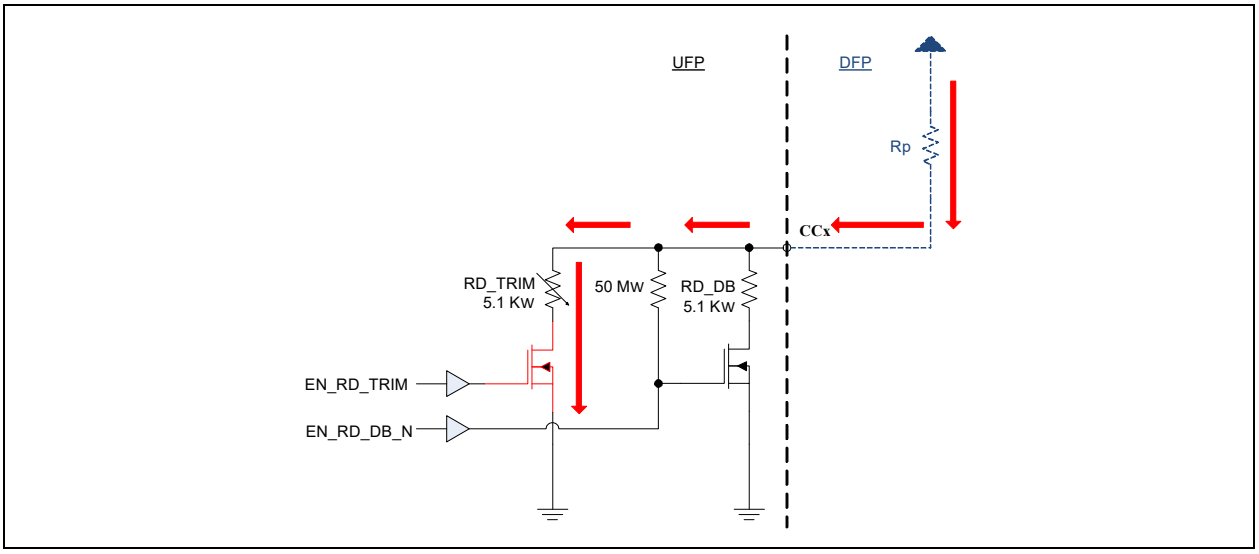


Figure 9-2 illustrates operation after the UFP has been powered over VBUS by the DFP. After the device is powered, EN_RD_DB asserts by default to keep the RD_DB pull-down activated.

Upon powering the host CPU, software simultaneously deasserts EN_RD_DB and asserts EN_RD_TRIM. Going forward the device presents RD_TRIM.

FIGURE 9-2: CC RD (TRIM)



The Rd resistor presented, trimmed or untrimmed, is controlled by the CC1 Pull-Down Value and CC2 Pull-Down Value in the CC Control Register (CC_CTL). These register fields serve the basis for the EN_RD_TRIM and EN_RD_DB_N control signals depicted.

9.8 VCONN Operation

VCONN is a 5V supply that is used to power circuitry in the USB Type-C® plug, which is required to implement Electronically Marked Cables. By default, the DFP always sources VCONN when connected to an active cable. However, this may be changed by software by using PD VCONN_SWAP.

The VCONN FETs are enabled/disabled by software via the VCONN1 ControlVCONN1 Control and VCONN2 Control control bits in the CC Control Register (CC_CTL).

APPLICATION NOTE: In standalone DFP mode (UPD350-A/UPD350-C only), the device independently enables/disables the VCONN FETs. This mode is intended for configurations where no host CPU is available or the CPU is not capable of managing VCONN. The Standalone OperationStandalone Operation bit in the CC Hardware Control Register (CC_HW_CTL)CC Hardware Control Register (CC_HW_CTL) enables this mode of operation.It is not envisioned to ever enable both FETs simultaneously.

VCONN is monitored for an over current condition via an internal monitoring circuit. A VCONN over current condition is recognized when the event persists for a time longer than specified. in the VCONN OCS and Back-Drive Debounce Register (VCONN_DEB). VCONN OCS monitoring is enabled via the VCONN OCS Enable bit in the VBUS Control Register (VBUS_CTL). When the VCONN Discharge Error (VCONN_DISCH_ERR) interrupt in the Power Interrupt Status Register (PWR_INT_STS) asserts. The device may be configured to automatically disable the VCONN FET upon detection of a CC1 Back-Drive Error/CC2 Back-Drive Error or VCONN Discharge Error (VCONN_DISCH_ERR). In the event of the detection of a debounced over-current VCONN event, the enabled VCONN FET will be disabled. The OCS event also results in an automatic disablement of the respective VCONN1 Control and VCONN2 Control control bits in the CC Control Register (CC_CTL).

9.8.1 VCONN DISCHARGE PROGRAMMING MODEL

Software may use the following programming model for implementing VCONN discharge when the device is operating as a DFP and a UFP disconnect has been detected..

This section does not apply to standalone DFP operation.

1. Software determines the attached UFP has disconnected and disconnects the VCONN FET.
2. Software disables monitoring the CC thresholds corresponding to the attached CC pin.
3. The Rp current sources are disabled on both CC pins via CC1 RP Value or CC2 RP Value.
4. The Ra pull-down on the CC pin previously sourcing VCONN is selected via CC1 Pull-Down Value or CC2 Pull-Down Value. This initiates the VCONN discharge.
5. Software sets the CC threshold 0 to 150 mv (41d), via the CC Threshold x Registers (CC_THRx), which is the VCONN discharge threshold.
6. Software programs 01h into the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN).
7. Software programs 01h into the CCx Match Enable Registers (CCx_MATCH_EN).
8. Software enables CC1_MATCH_CHG or CC2_MATCH_CHG interrupt via the CC Interrupt Enable Register (CC_INT_EN).
9. Via the CC Control Register (CC_CTL), the CC Comparator Control selects the CC pin that was sourcing VCONN.
10. Changes in state of the CC pin are recorded in CCx Match Registers (CCx_MATCH) and CCx Change Status Registers (CCx_CHG_STS) after the programmed debounce period.
11. After VCONN discharges below 150mv, **IRQ_N** asserts which indicates VCONN has been discharged. **IRQ_N** assertion of CC_MATCH_VLD with no threshold matches is also indicative of a complete discharge.
12. After the discharge has been completed, the device is in the Unattached.SRC state (Type-C Source FSM).

Note: Software should implement a timer to indicate a VCONN discharge error. This may be implemented using the General Purpose Timer. While there is no specific requirement in the Type-C specification for a maximum discharge time

Note: After the UFP disconnect is detected, firmware must disconnect the VCONN supply within tVconnOff (35 ms) per release 1.1 of the Type-C specification.

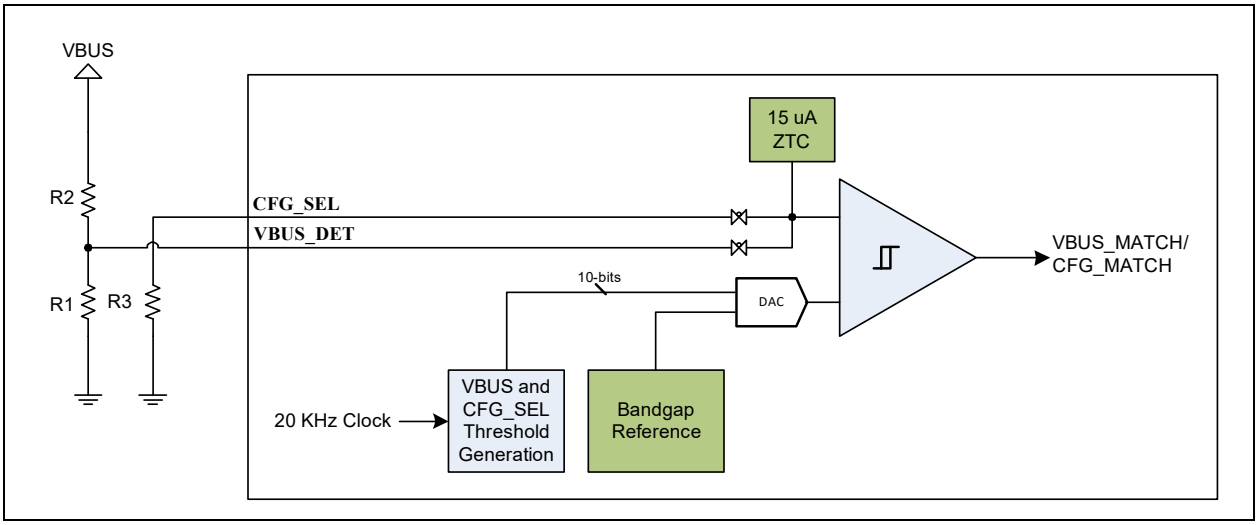
9.9 VBUS Detection

The device implements a comparator for determining when **VBUS** is within a programmed range, vSafe5V, or vSafe0v. VBUS is divided down externally via a 1:9 resistor divider to generate **VBUS_DET**. **VBUS_DET** is compared with an 8-bit threshold generated by an integrated DAC. The comparator is also shared by the **CFG_SEL** pin which is sampled automatically after a system reset

Figure 9-3 illustrates the **VBUS_DET** circuit. In a typical use case, **VBUS_DET** thresholds are programmed to track the following voltage ranges as defined in Table 9-7.

Note: Table 9-7 illustrates the values of **VBUS_DET** utilizing +/-1% accurate resistors where R1 is 10K Ohms and R2 is 90 kOhms.

FIGURE 9-3: VBUS_DET COMPARATOR



For a DFP, the VBUS comparator is useful to detect when VBUS is within the desired range per PD negotiations. This is the case when VBUS is generated by a source external to the device.

For a UFP, the VBUS comparator is required to determine when a DFP is attached or detached. It may also use the comparator to determine when VBUS is within a new voltage range negotiated via PD.

TABLE 9-7: VBUS DETECTION THRESHOLDS

VBUS	Range	VBUS_DET	Comments
20	21.5	2.11	
	18.5	1.82	
12	13.1	1.29	
	10.9	1.07	
8	8.9	0.88	
	7.1	0.69	
5	5.5	0.51	vSafe5V
	3.67	0.33	
0.68	0.68	0.068	vSafe0V

If supported, the ranges 8V, 12V and 20V may be programmed in VBUS Threshold 2 and VBUS Threshold 3 registers (see VBUS Threshold x Registers (VBUS_THRx)). Likewise 5V range, vSafe5v, can be programmed in VBUS Threshold 0 and VBUS Threshold 1 registers.

The threshold for vSafe0V is programmable via the VBUS VSafe0V Threshold Register (VSAFE0V_THR).

VBUS_DET monitoring logic operates off of the 20 KHz oscillator which cycles through each threshold. Including vSafe0v, a total of five values are compared.

Results of the comparison adjust the respective bits of the VBUS Match Register (VBUS_MATCH) and VBUS Change Status Register (VBUS_CHG_STS) after a debounce period defined in VBUS Debounce Register (VBUS_DEB).

The VBUS Match Register (VBUS_MATCH) indicates when the value on **VBUS_DET** is higher than the corresponding programmed threshold and can therefore be used to determine is **VBUS** is in the desired range.

A change in the state of the VBUS Match Register (VBUS_MATCH) may trigger assertion of the **IRQ_N** pin if appropriately configured in the Power Interrupt Status Register (PWR_INT_STS).

The VBUS Debounce Clear Enable Register (VBUS_DBCLR_EN) is functionality equivalent to the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN), but it applies to the VBUS debouncer. Software programs the VBUS Debounce Clear Enable Register (VBUS_DBCLR_EN) as required to enable debouncing the thresholds of interest.

The following example illustrates the programming model when it is desired to move **VBUS** from 5V to 20V after PD contract negotiation. Initially the **VBUS** de-bouncer is enabled and VSafe5v, VBUS_THR0 and VBUS_THR1, is sampled. VBUS_THR2 and VBUS_THR3 have an initial value in excess of 5V such as 25V. See Section 9.13.24 for details on the VBUS_THRx registers.

1. Software programs the VBUS Match Enable Register (VBUS_MATCH_EN) to include new thresholds (VBUS_THR2 and VBUS_THR3).
2. Software programs the VBUS Debounce Clear Enable Register (VBUS_DBCLR_EN) as required to enable per threshold debouncing.
3. Software programs new thresholds for 20V in VBUS_THR2 and VBUS_THR3 (e.g. 18.5V and 21.5V). This operation restarts the VBUS debouncer.
4. Software waits until **IRQ_N** asserts.
5. Software confirms the VBUS_MATCH_VLD bit is set.
6. Software reads the VBUS Match Register (VBUS_MATCH) and VBUS Change Status Register (VBUS_CHG_STS).
7. If VBUS Change Status Register (VBUS_CHG_STS) is non-zero, software clears the status bits.

Note:	If no bits are set in the VBUS Match Register (VBUS_MATCH), after VBUS_MATCH_VLD asserts, then the voltage observed on **VBUS** is less than VSafe0V.

8. Future changes in **VBUS** results in **IRQ_N** asserting with VBUS Match Register (VBUS_MATCH) and VBUS Change Status Register (VBUS_CHG_STS) being appropriately updated.

9.9.1 CONFIGURATION SELECTION

The **CFG_SEL** pin shares the comparator with **VBUS** as shown in Figure 9-3. The **CFG_SEL** pin is connected to a resistor divider, typically pulled up to VDDIO. After a system level reset (POR, **RESET_N**, Software Reset), the **CFG_SEL** pin is automatically sampled to configure the device. The internal CFG_SEL_MATCH register CFG_SELx Match Registers (CFG_SELx_MATCH) is updated automatically and the device configures itself accordingly if standalone mode is detected. The various resistor settings for the **CFG_SEL** pin are detailed in Table 9-8.

Note:	The **CFG_SEL** pin is used to determine the default I^2C slave address and operating mode in the UPD350-A / UPD350-C. For the UPD350-B / UPD350-D, this pin determines the mode in which the part operates (companion or standalone UFP) and can be used for customer specific purposes to provide a discrete value (0-15) based upon the attached resistor value.

Note:	For additional information on device resets, refer to Section 7.7, "Reset Operation," on page 33.

TABLE 9-8: CONFIGURATION SELECT (CFG_SEL) I^2C ADDRESS SETTINGS *(UPD350-A/UPD350-C ONLY)*

Resistor (+/-1%)	Description	CFG_SEL1_MATCH Register
GND	**Companion Mode I^2C Slave Address = 1011_111**	0000h
0.475 K	**Companion Mode I^2C Slave Address = 1011_110**	0001h
0.953 K	**Companion Mode I^2C Slave Address = 1011_101**	0003h
1.43 K	**Companion Mode I^2C Slave Address = 1011_100**	0007h
1.87 K	**Companion Mode I^2C Slave Address = 1101_011**	000Fh
2.37 K	**Companion Mode I^2C Slave Address = 1101_010**	001Fh
2.87 K	**Companion Mode I^2C Slave Address = 1101_001**	003Fh
3.32 K	**Companion Mode I^2C Slave Address = 1101_000**	007Fh
3.83 K	**Standalone UFP Mode I^2C Slave Address = 1011_111**	00FFh
4.22 K	**Standalone UFP Mode I^2C Slave Address = 1011_110**	01FFh
4.75 K	**Standalone UFP Mode I^2C Slave Address = 1011_101**	03FFh
5.23 K	**Standalone UFP Mode I^2C Slave Address = 1011_100**	07FFh
5.62 K	**Standalone UFP Mode I^2C Slave Address = 1101_011**	0FFFh
6.19 K	**Standalone DFP Mode I^2C Slave Address = 1101_010**	1FFFh
6.65 K	**Standalone DFP Mode I^2C Slave Address = 1101_001**	3FFFh
7.15 K	**Standalone DFP Mode I^2C Slave Address = 1101_000**	7FFFh
>7.15 K	**Standalone DFP Mode I^2C Slave Address = 1001_000**	FFFFh

9.10 Back-Drive Detection

Back-drive detection is implemented on both CC pins, which prevents backwards current flow. The back-drive protection circuit is always operational and triggers when VCCx > VS.

Detection of the back-drive condition causes the CC1 or CC2 Back-Drive Error interrupt to assert.

Hardware supports automatically disabling a VCONN FET on a CC pin in which back-drive was detected after a specified debounce period.

9.11 Standalone DFP *(UPD350-A/UPD350-C Only)*

9.11.1 OVERVIEW

The device supports standalone DFP operation in which no CPU is available to configure the device. A key application for this mode is a USB Type-C DFP companion for the Microchip USB58xx/USB59xx family of USB Hubs.

9.11.2 CONFIGURATION

This mode is entered by the appropriate setting of the **CFG_SEL** pin. The current charge advertised and supplied is defined by the **PWR_CAP0** and **PWR_CAP1** pins. The device auto configures itself after a system level reset event.

APPLICATION NOTE: The Standalone Operation field in CC Hardware Control Register (CC_HW_CTL) may be used to disable standalone operation.

9.12 Standalone UFP *(UPD350-A/UPD350-C Only)*

9.12.1 OVERVIEW

The device supports standalone UFP operation in which no CPU is available to configure the device. A key application for this mode is a USB Type-C® UFP companion for the Microchip USB58xx/USB59xx family of USB Hubs.

9.12.2 CONFIGURATION

This mode is entered by the appropriate setting of the **CFG_SEL** pin. The device auto-configures itself after a system level reset event.

APPLICATION NOTE: The Standalone Operation field in the CC Hardware Control Register (CC_HW_CTL) may be used to disable standalone operation.

9.13 Cable Orientation and Detection Registers

This section details the cable plug orientation and detection registers. For an overview of the entire device register map, refer to Section 4.0, "Register Map," on page 18.

TABLE 9-9: SYSTEM CONTROL AND STATUS REGISTERS MAP

Address	Register Name (Symbol)
0800h	CC Hardware Control Register (CC_HW_CTL)
0803h	CC Interrupt Status Register (CC_INT_STS)
0804h	CCx Change Status Registers (CCx_CHG_STS) x=1
0805h	CCx Change Status Registers (CCx_CHG_STS) x=2
0806h	CCx Match Registers (CCx_MATCH) x=1
0807h	CCx Match Registers (CCx_MATCH) x=2
0808h	VBUS Match Register (VBUS_MATCH)
0809h	VBUS Change Status Register (VBUS_CHG_STS)
080Ah	Power Interrupt Status Register (PWR_INT_STS)
080Bh	Debug Interrupt Status Register (DBG_INT_STS)
080Ch – 0810h	Reserved for future expansion
0811h	CC Interrupt Enable Register (CC_INT_EN)
0812h	CCx Match Enable Registers (CCx_MATCH_EN) x=1
0813h	CCx Match Enable Registers (CCx_MATCH_EN) x=2
0814h	VBUS Match Enable Register (VBUS_MATCH_EN)
0815h	Power Interrupt Enable Register (PWR_INT_EN)
0816h	Debug Interrupt Enable Register (DBG_ENT_EN)
0817h	Match Debounce Register (MATCH_DEB)
0818h	PD Debounce Register (PD_DEB)
0819h	VCONN OCS and Back-Drive Debounce Register (VCONN_DEB)
081Ah	CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) x=1
081Bh	CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) x=2
081Ch	VBUS Debounce Clear Enable Register (VBUS_DBCLR_EN)
081Dh	CCx Sample Enable Registers (CCx_SAMP_EN) x=1
081Eh	CCx Sample Enable Registers (CCx_SAMP_EN) x=2
081Fh	Reserved for future expansion
0820h	CC Control Register (CC_CTL)
0822h	CC Threshold x Registers (CC_THRx) x=0
0824h	CC Threshold x Registers (CC_THRx) x=1
0826h	CC Threshold x Registers (CC_THRx) x=2
0828h	CC Threshold x Registers (CC_THRx) x=3
082Ah	CC Threshold x Registers (CC_THRx) x=4
082Ch	CC Threshold x Registers (CC_THRx) x=5
082Eh	CC Threshold x Registers (CC_THRx) x=6
0830h	CC Threshold x Registers (CC_THRx) x=7
0832h	CC Debounce Register (CC_DEB)
0834h – 083Fh	Reserved for future expansion
0840h	VBUS Control Register (VBUS_CTL)
0842h	VBUS Threshold x Registers (VBUS_THRx) x=0
0844h	VBUS Threshold x Registers (VBUS_THRx) x=1
0846h	VBUS Threshold x Registers (VBUS_THRx) x=2

TABLE 9-9: SYSTEM CONTROL AND STATUS REGISTERS MAP (CONTINUED)

Address	Register Name (Symbol)
0848h	VBUS Threshold x Registers (VBUS_THRx) x=3
084Ah	VBUS Debounce Register (VBUS_DEB)
084Bh	VBUS Off Register (VBUS_OFF)
084Ch	VBUS Error Register (VBUS_ERR)
084Dh	Reserved for future expansion
084Eh	VBUS VSafe0V Threshold Register (VSAFE0V_THR)
0850h	CFG_SELx Match Registers (CFG_SELx_MATCH) x=0
0852h	CFG_SELx Match Registers (CFG_SELx_MATCH) x=1
0854h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=0
0856h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=1
0858h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=2
085Ah	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=3
085Ch	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=4
085Eh	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=5
0860h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=6
0862h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=7
0864h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=8
0866h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=9
0868h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=10
086Ah	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=11
086Ch	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=12
086Eh	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=13
0870h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=14
0872h	CFG_SEL Threshold x Registers (CFG_SEL_THRx) x=15
0874h	CFG_SEL Debug Register (CFG_SEL_DBG)
0875h – 0885h	Reserved for future expansion
0886h	VCONN Discharge Threshold Register (VCONN_DIS_THR)
0888h	VCONN Discharge Time Register (VCONN_DIS_TIME)
088Ah – 0BFFh	Reserved for future expansion

Note: RESERVED address space must not be written under any circumstances. Failure to heed this warning may result in untoward operation and unexpected results.

9.13.1 CC HARDWARE CONTROL REGISTER (CC_HW_CTL)

Address: 0800h Size: 16 bits

Bits	Description	Type	Default
15:14	**SHORT_DET** Defines behavior in standalone DFP mode for handling the short circuit condition. 0xb: A0 functionality maintained where VBUS short is not checked before initiating VBUS discharge. 10b: Device initiates VBUS discharge only after short circuit condition has been removed. 11b: Device initiates VBUS discharge after short circuit condition has been removed or a time equal to TCYCLE. **Note:** This field only has meaning for standalone DFP operation.	R/W	00b
13	**RESERVED**	RO	-
12	**BLK_PD_MSG** This bit causes the CC debouncer to pause whenever a PD message is detected by the MAC. This applies for both transmit and receive messages. After the message is processed 0b: Disable PD message filtering. 1b: Enable PD message filtering. **Note:** This must be enabled for collision detection.	R/W	0b
11	**MATCH_DB_UNITS** This bit defines the units of the Match Debounce Register (MATCH_DEB). 0b: Match debounce has units of 1.6 ms. 1b: Match debounce has units of 100 us. **Note:** 1.6 ms is derived from the time to cycle through all thresholds on both CC pins.	R/W	0b
10	**DEVICE_MODE** Indicates the current mode of the device. 0b: Device is in Companion Mode 1b: Device is in Standalone Mode *(UPD350-A/UPD350-C only)*	RO	0b
9	**CC_DB_ACTIVE** When this bit reads back 0b, the debouncer is disabled. The CC debouncer is enabled when it reads back 1b. **Note:** Software may poll this bit to determine when the CC debouncer is disabled, which can be used as a condition for programming a new configuration.	RO	0b
8	**DEVICE_STATE** Indicates the current state of the device attachment. 0b: Device is not attached 1b: Device is attached	RO	0b

Bits	Description	Type	Default
7:6	**Power Capability** Indicates the charging current capacity of the device as defined by the **PWR_CAPx** pins. *(UPD350-A/UPD350-C only)* This field only has meaning for configurations where the **PWR_CAPx** pins are available *(UPD350-A/UPD350-C only)*.	RO	Note 9-2
5:3	**Detach Threshold Select** Defines which CC threshold shall be used for determining a detach when operating as a standalone DFP *(UPD350-A/UPD350-C only)*.	R/W	Note 9-2
2	**Device Role** 0b: Device configured as UFP. 1b: Device configured as DFP. **Note:** This bit must not be modified while the CC debouncer is enabled.	R/W	Note 9-3
1	**RESERVED**	RO	-
0	**Standalone Operation** 0b: Companion Mode 1b: Standalone mode *(UPD350-A/UPD350-C only)* Standalone mode: This is the mode of operation supported for standalone operation. During standalone operation, the I^2C interface may not be available and the host CPU is not capable of managing the UPD350. The following functions can be handled by the internal logic when this bit is set: • VCONN Enabled/Disable • Attach detection and assertion of **ATTACH** • Detach detection and de-assertion of **ATTACH** • Orientation detection and assertion/de-assertion of **ORIENTATION** pin • Enables CC de-bouncer.	R/W	Note 9-1

Note 9-1 Default is 1b when a standalone configuration is selected otherwise the default is 0b.

Note 9-2 Default is a function of the **PWR_CAP0** and **PWR_CAP1** pins, when the configuration supports these pins. Otherwise the default is 0h.

Note 9-3 Default is a function of the **CFG_SEL0** and **CFG_SEL1** pins. When the configuration indicates UFP configuration this bit defaults to 0b. For DFP operation this bit defaults to 1b. When neither UFP or DFP is specified the default shall be 0b.

9.13.2 CC INTERRUPT STATUS REGISTER (CC_INT_STS)

Address: 0803h Size: 16 bits

Bits	Description	Type	Default
7	**CC_MATCH_VLD** Asserts after the CC debouncer is first enabled, via CC Comparator Control, and the first match becomes valid in CCx Match Registers (CCx_MATCH).	R/WC	0b
6	**RP_CHG** When operating as a standalone UFP, this interrupt indicates a change in the state of the RP value advertised by the DFP has been detected *(UPD350-A/UPD350-C only).* **Note:** This bit is RO when the device is not configured as a standalone UFP and will always read back 0b. **Note:** The source of this input is a pulse and does not persist after being cleared. **Note:** The updated current advertisement is available via the DFP Current Advertisement field in the CC Control Register (CC_CTL).	R/WC	0b
5	**DETACH** Indicates a detach event has occurred when the device has been configured to support standalone mode per the Standalone Operation field in the CC Control Register (CC_CTL) *(UPD350-A/UPD350-C only)*. DFP Operation: CC pins are monitored for detecting a detach. UFP Operation: VBUS is monitored for detecting a detach. **Note:** The source of this input is a pulse and does not persist after being cleared.	R/WC	0b
4	**ATTACH** Indicates an attach event has occurred when the device has been configured to support standalone mode per the Standalone Operation field in the CC Control Register (CC_CTL) *(UPD350-A/UPD350-C only)*. DFP Operation: CC pins are monitored for an attach. UFP Operation: CC pins and VBUS are monitored for an attach. **Note:** The source of this input is a pulse and does not persist after being cleared.	R/WC	0b
3:2	**RESERVED**	RO	-
1	**CC2_MATCH_CHG** Indicates the a change occurred in the state of the respective CCx Change Status Registers (CCx_CHG_STS). **Note:** The source of this input is a pulse and does not persist after being cleared.	RO	0b
0	**CC1_MATCH_CHG** Indicates the a change occurred in the state of the respective CCx Change Status Registers (CCx_CHG_STS). **Note:** The source of this input is a pulse and does not persist after being cleared.	RO	0b

9.13.3 CC*X* CHANGE STATUS REGISTERS (CC*X*_CHG_STS)

Address: x=1: 0804h
x=2: 0805h

Size: 8 bits

Bits	Description	Type	Default
7:0	**CCx Change Status** When set, each bit indicates that the respective bit in the CCx Match Registers (CCx_MATCH) has changed. A write of 1b clears the respective status bit.	R/WC	0h

9.13.4 CC*X* MATCH REGISTERS (CC*X*_MATCH)

Address: x=1: 0806h
x=2: 0807h

Size: 8 bits

Bits	Description	Type	Default
7:0	**CCx Threshold Match (CCx_MATCH)** When set, each bit indicates that the respective threshold programmed in the CC Threshold x Registers (CC_THRx) was matched. A match is determined when the measured voltage exceeds the programmed threshold. These registers are updated after the Match Debounce Register (MATCH_DEB) while in the Unattached state. While in AttachWait state, these registers are updated after the PD Debounce Register (PD_DEB) if vOpen is seen for a time greater than in the PD Debounce Register (PD_DEB). While in Attached Source / Attached Sink states (e.g. detecting detach) the match registers are updated after the PD Debounce Register (PD_DEB). **Note:** This register will always read the default value until the CC comparator is enabled when not operating in standalone mode. **Note:** The contents of this register are debounced per the settings in the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN), otherwise the raw value shall be shown.	RO	Note 9-4

Note 9-4 Defaults to FFh when the device is configured as a DFP and 00h when configured as a UFP, per the Device Role field in the CC Control Register (CC_CTL).

9.13.5 VBUS MATCH REGISTER (VBUS_MATCH)

Address: 0808h Size: 8 bits

Indicates which VBUS thresholds are matched on **VBUS_DET** pin. A match is determined when the measured voltage exceeds the programmed threshold.

Note: When not operating in standalone mode, this register will always read 0h until the VBUS comparator is enabled.

Note: The contents of this register shall be debounced per the settings in VBUS Debounce Clear Enable Register (VBUS_DBCLR_EN).

Bits	Description	Type	Default
7:6	**RESERVED**	RO	-
5	**VBUS Threshold 3 Match (VBUS3_THR_MATCH)**	RO	0b
4	**VBUS Threshold 2 Match (VBUS2_THR_MATCH)**	RO	0b
3	**VBUS Threshold 1 Match (VBUS1_THR_MATCH)**	RO	0b
2	**VBUS Threshold 0 Match (VBUS0_THR_MATCH)**	RO	0b
1	**RESERVED**	RO	-
0	**VBUS VSafe0v Match (VSAFE0V_THR_MATCH)**	RO	0b

9.13.6 VBUS CHANGE STATUS REGISTER (VBUS_CHG_STS)

Address: 0809h Size: 8 bits

Bits	Description	Type	Default
7:0	**VBUS Change Status (VBUS_CHG_STS)** When set, each bit indicates that the respective bit in VBUS Match Register (VBUS_MATCH) has changed. A write of 1b clears the respective status bit.	R/WC	0h

9.13.7 POWER INTERRUPT STATUS REGISTER (PWR_INT_STS)

Address: 080Ah Size: 8 bits

Bits	Description	Type	Default
7	**VBUS_MATCH_VLD** Asserts after the VBUS debouncer is first enabled, via VBUS Comparator Control, and the first match becomes valid in VBUS Match Register (VBUS_-MATCH).	R/WC	0b
6	**VCONN Discharge Error (VCONN_DISCH_ERR)** When set this bit indicates that a VCONN discharge error has been detected. This bit is only available in standalone DFP mode and is otherwise reserved. A discharge error is tracked when VCONN Discharge Control is set to 00b or 01b. A discharge error is asserted when VCONN fails to fall below VCONN Discharge Threshold (VCONN_DIS_THR) after a time greater than VCONN Discharge Time (VCONN_DIS_TIME) elapses.	R/WC	0b
5	**CC2 Back-Drive Error** When set, indicates that back-drive has been detected on the CC2 pin. **Note:** The source of this input is a level and persists until the error condition stops.	R/WC	0b
4	**VBUS Discharge Error** When set, indicates that the an interval greater than defined in the VBUS Off Register (VBUS_OFF) has elapsed while attempting to discharge VBUS. **Note:** The source of this input is a pulse and does not persist after being cleared.	R/WC	0b
3	**VCONN2 FET Power** The integrated VCONN2 FET is enabled and providing power. This bit only has usefulness for standalone operation or where VCONN is explicitly enabled by the host. **Note:** The source of this input is a level and persists until the VCONN is no longer present.	R/WC	0b
2	**VCONN1 FET Power** The integrated VCONN1 FET is enabled and providing power. This bit only has usefulness for standalone operation or where VCONN is explicitly enabled by the host. **Note:** The source of this input is a level and persists until the VCONN is no longer present.	R/WC	0b
1	**CC1 Back-Drive Error** When set indicates that back-drive has been detected on the CC1 pin. **Note:** The source of this input is a level and persists until the error condition stops.	R/WC	0b
0	**VCONN Over-Current Error** Indicates an over-current has been detected on the integrated VCONN FET. **Note:** The source of this input is a level and persists until the error condition stops.	R/WC	0b

9.13.8 DEBUG INTERRUPT STATUS REGISTER (DBG_INT_STS)

Address: 080Bh Size: 8 bits

Bits	Description	Type	Default
7:4	**RESERVED**	RO	0h
3	**VCONN_DISCH_STS** When set, indicates that the CC pin previously sourcing VCONN is being discharged. This bit only exists in standalone DFP mode and is otherwise reserved.	R/WC	0b
2	**VBUS_DISCH** When set indicates this indicates that VBUS is being discharged. **Note:** The source of this input is a level and it persists until the discharge has completed.	R/WC	0b
1	**CFG_SEL1 Done** When set, indicates that all CFG_SEL Threshold x Registers (CFG_SEL_THRx) have been measured in response to the request enabled by VBUS Comparator Control to sample the **CFG_SEL1** pin and the results are readable in the respective CFG_SELx Match Registers (CFG_SELx_-MATCH) *(UPD350-A/UPD350-C only)*. **Note:** The source of this input is a pulse and does not persist after being cleared.	R/WC	0b
0	**CFG_SEL0 Done** When set, indicates that all CFG_SEL Threshold x Registers (CFG_SEL_THRx) have been measured in response to the request enabled by VBUS Comparator Control to sample the **CFG_SEL0** pin and the results are readable in the respective CFG_SELx Match Registers (CFG_SELx_-MATCH). **Note:** The source of this input is a pulse and does not persist after being cleared.	R/WC	0b

9.13.9 CC INTERRUPT ENABLE REGISTER (CC_INT_EN)

Address: 0811h Size: 8 bits

Bits	Description	Type	Default
7:4	**CC Interrupt Enable** When “0”, prevents generation of the respective interrupt.	R/W	0000b
3:2	**RESERVED**	-	00b
1:0	**CC Interrupt Enable** When “0”, prevents generation of the respective interrupt.	R/W	00b

9.13.10 CC*X* MATCH ENABLE REGISTERS (CC*X*_MATCH_EN)

Address: x=1: 0812h
x=2: 0813h
Size: 8 bits

Bits	Description	Type	Default
7:0	**CCx Match Enable** When set, the corresponding bit in the CCx Change Status Registers (CCx_CHG_STS) can cause the assertion of the respective CCx Match Change interrupt (CC1_MATCH_CHG/CC2_MATCH_CHG).	R/W	00h

9.13.11 VBUS MATCH ENABLE REGISTER (VBUS_MATCH_EN)

Address: 0814h
Size: 8 bits

When set, the corresponding bit in the VBUS Change Status Register (VBUS_CHG_STS) can cause the assertion of the VBUS_INT interrupt.

Bits	Description	Type	Default
7:6	**RESERVED**	RO	-
5	**VBUS Match Enable[5]**	R/W	0b
4	**VBUS Match Enable[4]**	R/W	0b
3	**VBUS Match Enable[3]**	R/W	0b
2	**VBUS Match Enable[2]**	R/W	0b
1	**VBUS Match Enable[1]**	RO	0b
0	**VBUS Match Enable[0]**	R/W	0b

9.13.12 POWER INTERRUPT ENABLE REGISTER (PWR_INT_EN)

Address: 0815h
Size: 8 bits

Bits	Description	Type	Default
7:0	**Power Interrupt Enable [7:0]** When “0”, prevents generation of the respective interrupt.	R/W	0h

9.13.13 DEBUG INTERRUPT ENABLE REGISTER (DBG_ENT_EN)

Address: 0816h Size: 8 bits

Bits	Description	Type	Default
7:4	**RESERVED**	RO	0h
3	**Debug Interrupt Enable 3** When “0”, prevents generation of the respective interrupt. This bit only exists in standalone DFP mode and is otherwise reserved.	R/W	0b
2:0	**Debug Interrupt Enable [2:0]** When “0”, prevents generation of the respective interrupt.	R/W	0h

9.13.14 MATCH DEBOUNCE REGISTER (MATCH_DEB)

Address: 0817h Size: 8 bits

Bits	Description	Type	Default
7:0	**Match Debounce** Defines the debounce period utilized before updating the CCx Match Registers (CCx_MATCH) when not operating in standalone mode. The units of this register is determined by MATCH_DB_UNITS bit in CC Hardware Control Register (CC_HW_CTL). When this bit is clear, the units are 1.6 ms. When this bit is set, the units are 100 us. **Note:** This register must not be modified while the CC debouncer is enabled. **Note:** The actual debounce time may be +/-1 from the cycle time programmed. **Note:** The value programmed in this CSR should be at least equal to the number of thresholds enabled in CCx Sample Enable Registers (CCx_SAMP_EN). This is only an issue when MATCH_DB_UNITS is set to 1b.	R/W	2h

9.13.15 PD DEBOUNCE REGISTER (PD_DEB)

Address: 0818h Size: 8 bits

Bits	Description	Type	Default
7:0	**PD Debounce (PD_DEB)** Period used for implementing $t_{PdDebounce}$. **Note:** This register must not be modified while the CC debouncer is enabled. **Note:** This register has units of 1 ms.	R/W	Ah

9.13.16 VCONN OCS AND BACK-DRIVE DEBOUNCE REGISTER (VCONN_DEB)

Address: 0819h Size: 8 bits

Bits	Description	Type	Default
7:0	**VCONN and Back-Drive Debounce (VCONN_DEB)** Period used for implementing debounce of over-current detected on VCONN FET as well as back-drive detected on the CC pins. **Note:** This register has units of 1 ms. **Note:** This register should not be changed when VCONN OCS Enable is set.	R/W	2h

9.13.17 CC*X* DEBOUNCE CLEAR ENABLE REGISTERS (CC*X*_DBCLR_EN)

Address: x=1: 081Ah x=2: 081Bh Size: 8 bits

Bits	Description	Type	Default
7:0	**CC Debounce Clear Enable (CC_DBCLR_DEB)** When a bit is set, the respective threshold shall be included in the CC debouncer. Alternatively, when cleared the respective threshold shall no longer be considered by the debouncer. When CCx_DBCLR_EN bits are set on-the-fly, if a mismatch between the current raw match vector (for the new CCx_DBCLR_EN) and the previous raw match vector (for the old CCx_DBCLR_EN) exists, the DB will be reset. **Note:** Clearing bits in this register at run time does not reset the debouncer. **Note:** The CC debouncer encompasses both **CC1/CC2** pins. A detected change for a threshold on either pin results in the debouncer resetting. **Note:** Even though this register may change on-the-fly, the internal logic will enable the change only at the end of the scan cycle, which is a function of whether the **CC1/CC2** pins are actively sampled and the CCx Sample Enable Registers (CCx_SAMP_EN).	R/W	Note 9-5

Note 9-5 The default depends on the device's configuration, as shown in table Table 9-10 which depends upon **CFG_SEL0**, **PWR_CAP0**, and **PWR_CAP1** pins.

APPLICATION NOTE: Clearing a bit in CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) shall cause the respective bit in CCx Match Registers (CCx_MATCH) to be immediately updated upon a change in state of the associated threshold. This causes a state change in CCx Change Status Registers (CCx_CHG_STS) and assertion of CC_INT, if enabled. To prevent this CCx Match Enable Registers (CCx_MATCH_EN) should be updated before the CCx Debounce Clear Enable Registers (CCx_DBCLR_EN) by having the associated threshold cleared.

TABLE 9-10: CCX_DBLCLR_DEB DEFAULTS

Configuration	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Standalone UFP *(UPD350-A/UPD350-C only)*	1	0	1	0	1	0	0	0
Standalone DFP (Default Current) *(UPD350-A/UPD350-C only)*	1	0	0	0	0	1	0	0
Standalone DFP (1.5 A) *(UPD350-A/UPD350-C only)*	0	1	0	0	0	1	0	0
Standalone DFP (3.0 A) *(UPD350-A/UPD350-C only)*	0	0	0	1	0	0	1	0
Other	0	0	0	0	0	0	0	0

9.13.18 VBUS DEBOUNCE CLEAR ENABLE REGISTER (VBUS_DBCLR_EN)

Address: 081Ch Size: 8 bits

When a bit is set, the respective threshold shall be included in the VBUS debouncer. Alternatively, when cleared, the respective threshold shall no longer be considered by the debouncer.

When VBUS_DBCLR_EN bits are set on-the-fly, if a mismatch between the current raw match vector (for the new VBUS_DBCLR_EN) and the previous raw match vector (for the old VBUS_DBCLR_EN) exists, the DB will be reset.

Note: Clearing bits in this register at run time does not reset the debouncer.

APPLICATION NOTE: Clearing a bit in VBUS Debounce Clear Enable Register (VBUS_DBCLR_EN) shall cause the respective bit in VBUS Match Register (VBUS_MATCH) to be immediately updated upon a change in state of the associated threshold. This causes a state change in VBUS Change Status Register (VBUS_CHG_STS) and assertion of VBUS_INT, if enabled. To prevent this, the VBUS Match Register (VBUS_MATCH) should be updated before the VBUS Debounce Clear Enable Register (VBUS_DBCLR_EN) by having the associated threshold cleared.

BITS	DESCRIPTION	TYPE	DEFAULT
7:6	**RESERVED**	RO	-
5	**VBUS Debounce Clear Enable (VBUS3_DBCLR_DEB)**	R/W	Note 9-6
4	**VBUS Debounce Clear Enable (VBUS2_DBCLR_DEB)**	R/W	Note 9-6
3	**VBUS Debounce Clear Enable (VBUS1_DBCLR_DEB)**	R/W	Note 9-6
2	**VBUS Debounce Clear Enable (VBUS0_DBCLR_DEB)**	R/W	Note 9-6
1	**RESERVED**	RO	-
0	**VSAFE0V Debounce Clear Enable 9 (VSAFE0V_DBCLR_DEB)**	R/W	Note 9-6

Note 9-6 The default depends on the device's configuration, as shown in table Table 9-11 which depends upon **CFG_SEL0** pin.

TABLE 9-11: VBUS_DBCLR_EN DEFAULTS

Configuration	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Standalone UFP *(UPD350-A/UPD350-C only)*	1	0	1	1	0	0	0	0
Standalone DFP *(UPD350-A/UPD350-C only)*	1	0	1	1	0	0	0	0
Other	0	0	0	0	0	0	0	0

9.13.19 CC*X* SAMPLE ENABLE REGISTERS (CC*X*_SAMP_EN)

Address: x=1: 081Dh x=2: 081Eh Size: 8 bits

BITS	DESCRIPTION	TYPE	DEFAULT
7:0	**CC Sample Enable (CC_SAMP_EN)** When a bit is set, the respective CC threshold will be sampled by the CC debouncer. When a bit cleared to 0b, the corresponding bit(s) in the CCx Match Registers (CCx_MATCH) and CCx Change Status Registers (CCx_CHG_STS) will always read 0b. This register enables a reduction in latency for taking threshold measurements by only having thresholds of interest being sampled. **Note:** For standalone DFP/UFP operation this register may remain set to FFh, as latency for sampling CC thresholds in this case is not a limitation. **Note:** This register must be used to implement Collision Avoidance.	R/W	FFh

9.13.20 CC CONTROL REGISTER (CC_CTL)

Address: 0820h Size: 16 bits

The register controls the pull-down resistors and current sources on the respective **CC1/CC2** pin.

Bits	Description	Type	Default
15	**RA Detect** When set, indicates that an RA resistor has been detected. This bit is set by the device when Standalone Operation is configured for standalone mode. The device updates this field after an attach has been detected.	R/WC	0b
14:13	**CC Comparator Control** 00b: CC Comparator and DAC powered-down 01b: CC Comparator samples **CC1** 10b: CC Comparator samples **CC2** 11b: CC Comparator samples **CC1** and **CC2** **Note:** A sample is taken every 100 us. The current source reference shall also be powered down when a value of 00b is set. **Note:** This field is RO and the contents are **11b** when standalone mode is enabled. **Note:** This field is used in conjunction with the CCx Sample Enable Registers (CCx_SAMP_EN) which constrains the number of thresholds monitored on the enabled CC pin(s).	Note 9-7	00b
12	**CC Communication Select** 0b: **CC1** is used for baseband communication. 1b: **CC2** is used for baseband communication. This bit is set by the device when Standalone Operation is configured for standalone mode. The device updates this field after an attach has been detected. This bit is RO and reflects the state determined by the internal logic. In standalone mode, this bit matches the state of the **ORIENTATION** pin *(UPD350-A/UPD350-C only)*.	Note 9-7	0b
11:10	**CC2 RP Value** 00b: RP current source disabled 01b: RP current source enabled, default USB power 10b: RP current source enabled, 1.5A 11b: RP current source enabled, 3.0A Controls RP value on **CC2** pin.	R/W	00b Note 9-9
9:8	**CC1 RP Value** 00b: RP current source disabled 01b: RP current source enabled, default USB power 10b: RP current source enabled, 1.5A 11b: RP current source enabled, 3.0A Controls RP value on **CC1** pin.	R/W	00b Note 9-9

Bits	Description	Type	Default
7:6	**DFP Current Advertisement** When Standalone Operation is configured for standalone mode and the device is configured as UFP, Device Role, this field indicates the DFP's advertised current. Otherwise this field shall be read back 00b. The device updates this field after an attach has been detected. 0xb: RP current advertises default USB current 10b: RP current advertises 1.5A 11b: RP current advertises 3.0A	RO	00b
5	**RESERVED**	RO	-
4:3	**CC2 Pull-Down Value** 00b: Dead battery RD resistor selected 01b: Trimmed RD resistor selected 10b: Trimmed RA resistor selected. 11b: Open Disconnect.	R/W	Note 9-10
2	**RESERVED**	RO	-
1:0	**CC1 Pull-Down Value** 00b: Dead battery RD resistor selected 01b: Trimmed RD resistor selected 10b: Trimmed RA resistor selected. 11b: Open Disconnect.	R/W	Note 9-10

Note 9-7 This bit is RO when operating in standalone mode or attach detection. Otherwise it is R/W.

Note 9-8 This bit is RO when operating in standalone mode. Otherwise it is R/W.

Note 9-9 This field's default is a function of the **PWR_CAP0** and **PWR_CAP1** pins when in standalone DFP mode (see Section 9.9.1, "Configuration Selection") *(UPD350-A/UPD350-C only)*. Otherwise the default is 00b.

Note 9-10 For standalone DFP and standalone UFP, the value is 00b until the system reset completes.the OTP calibration completes as shown in Figure x.x "System Reset". Afterwards, the default for standalone DFP is 11b and for standalone UFP is 01b.

9.13.21 CC THRESHOLD *X* REGISTERS (CC_THR*X*)

Address: x=0: 0822h
x=1: 0824h
x=2: 0826h
x=3: 0828h
x=4: 082Ah
x=5: 082Ch
x=6: 082Eh
x=7: 0830h

Size: 16 bits

Bits	Description	Type	Default
15:10	**RESERVED**	RO	-
9:0	**CC Threshold X (CC_THRX)** CC Threshold X register. **Note:** The units of this register are ~2.44 mV from a 2.5V/1024. **Note:** This register must not be modified while the CC debouncer is enabled.	R/W	Note 9-11

Note 9-11 The default varies as shown in Table 9-12Defaults values are loaded from OTP. If the OTP is not configured..

TABLE 9-12: CC_THR DEFAULTS

CC_THR	Type-C Threshold	R/2R Divider	VALUE
0	0.2	0.13	55
1	0.4	0.27	109
2	0.66	0.44	180
3	0.8	0.53	219
4	1.23	0.82	336
5	1.6	1.07	437
6	2.6	1.73	710
7	3	2	820

APPLICATION NOTE: The CC Comparator must be powered down before updating these registers.

9.13.22 CC DEBOUNCE REGISTER (CC_DEB)

Address: 0832h

Size: 8 bits

Bits	Description	Type	Default
7:0	**CC Debounce (CC_DEB)** Period used for implementing $t_{CCDebounce}$. **Note:** This register must not be modified while the CC debouncer is enabled. **Note:** This register has units of 10 ms.	R/W	Fh

9.13.23 VBUS CONTROL REGISTER (VBUS_CTL)

Address: 0840h Size: 16 bits

Bits	Description	Type	Default
15:12	**RESERVED**	RO	-
11	**CC Back-Drive Enable** Enables the monitoring of the back-drive condition on both CC pins. When back-drive is detected, if the VCONN FET is enabled on the erred CC pin, it shall be automatically disabled. 0b: CC Back-Drive disabled 1b: CC Back-Drive enabled	R/W	Note 9-12
10	**IBUS_LOW** Determines whether IBUS Low is asserted during standalone mode when PPC's DISCH_SEL is cleared and a VBUS discharge is occurring. 0b: Do not assert IBUS Low 1b: Assert IBUS Low	R/W	0b
9:8	**OCS_MIN** Defines the minimum guaranteed assertion time for $\overline{OCS\_N}$ when operating in standalone DFP mode *(UPD350-A/UPD350-C only).* 00b: 5 ms 01b: 10 ms 10b: 20 ms 11b: 30 ms	R/W	00b
7	**VBUS_DB_ACTIVE** When this bit reads back 0b, the debouncer is disabled. The VBUS debouncer is enabled when it reads back 1b. **Note:** Firmware polls this bit to determine when debouncer is disabled and a new configuration may be programmed.	RO	0b
6	**VCONN OCS Enable** Enables the monitoring of over current condition on the internal VCONN FETs. 0b: VCONN OCS monitor is disabled 1b: VCONN OCS monitor is enabled	R/W	Note 9-12

Bits	Description	Type	Default
5:4	**VCONN Discharge Control** This field determines the VCONN discharge behavior. It only has meaning in DFP standalone mode and should not be used by firmware when operating in companion mode. The discharge occurs on the CC pin that was supplying VCONN. 00b: Discharge VCONN until either the threshold defined by the VCONN Discharge Threshold Register (VCONN_DIS_THR) is reached or the time specified by the VCONN Discharge Time Register (VCONN_DIS_TIME) has expired. 01b: VCONN discharge is not supported. 10b: Discharge VCONN until threshold defined by VCONN Discharge Threshold Register (VCONN_DIS_THR) is reached. 11b: Discharge VCONN for the time specified by VCONN Discharge Time Register (VCONN_DIS_TIME). **Note:** For options 0xb a VCONN discharge error is detected if the timer expires and VCONN has not discharged below VCONN Discharge Time Register (VCONN_DIS_TIME).	R/W	00b
3	**VCONN2 Control** Enables the VCONN2 FET. 0b: VCONN2 FET is disabled 1b: VCONN2 FET is enabled This bit has no meaning when in standalone mode (see Standalone Operation). This bit automatically clears when a debounce VCONN OCS event occurs per the assertion of VCONN Discharge Error (VCONN_DISCH_ERR).	R/W	0b
2	**VCONN1 Control** Enables the VCONN1 FET. 0b: VCONN1 FET is disabled 1b: VCONN1 FET is enabled This bit has no meaning when in standalone mode (see Standalone Operation). This bit automatically clears when a debounce VCONN OCS event occurs per the assertion of VCONN Discharge Error (VCONN_DISCH_ERR)	R/W	0b
1:0	**VBUS Comparator Control** 00b: Comparator and DAC disabled 01b: Sample VBUS 10b: Sample **CFG_SEL** 11b: RESERVED After the sample of **CFG_SEL** has completed, this register resets itself to 00b and disables the VBUS comparator. **Note:** This field is forced to 01b by hardware when operating as a standalone UFP.	Note 9-13	00b

Note 9-12 This default value is 1 when operating in standalone DFP mode. Otherwise, the default value is 0.

Note 9-13 This field is RO and reads back 01b when operating in UFP standalone mode. In standalone mode this value does not indicate that VBUS is being actively measured. Whether or not VBUS is being actively measured can be determined by reading VBUS_DB_ACTIVE.

9.13.24 VBUS THRESHOLD *X* REGISTERS (VBUS_THR*X*)

Address: x=0: 0842h
x=1: 0844h
x=2: 0846h
x=3: 0848h

Size: 16 bits

Bits	Description	Type	Default
15:10	**RESERVED**	RO	-
9:0	**VBUS Threshold X (VBUS_THRX)** VBUS Threshold X register. The lower byte of the threshold must be written before the upper byte. The entire 10-bit threshold is updated when the second write occurs. **Note:** The units of this register are ~2.44 mV from a 2.5V FS.	R/W	Note 9-14

Note 9-14 The defaults are defined in Table 9-13. Defaults values are loaded from OTP. If the OTP is not configured.

TABLE 9-13: VBUS_THR DEFAULTS

VBUS_THR	VBUS Threshold	1R/9R Divider	VALUE
0	3.67	0.36	148
1	5.5	0.54	222
2	5.5	0.54	222
3	5.5	0.54	222

APPLICATION NOTE: This register may be dynamically written to by FW while the VBUS comparator is enabled provided the rules for updating defined in the CSR description are followed.

9.13.25 VBUS DEBOUNCE REGISTER (VBUS_DEB)

Address: 084Ah

Size: 8 bits

This register has units of 1 ms.

Bits	Description	Type	Default
7:0	**VBUS Debounce (VBUS_DEB)** Indicates debounce interval for the VBUS threshold comparators. **Note:** This register must not be modified while the VBUS debouncer is enabled.	R/W	1h

9.13.26 VBUS OFF REGISTER (VBUS_OFF)

Address: 084Bh Size: 8 bits

This register has units of 10 ms.

Bits	Description	Type	Default
7:0	**VBUS Off** Defines timing after **VBUS_DET** discharges below VSafe0V.	R/W	10h

Note 9-15 Default is loaded from OTP see VBUS_OFF. If OTP is not configured the default is 10h.

9.13.27 VBUS ERROR REGISTER (VBUS_ERR)

Address: 084Ch Size: 8 bits

Bits	Description	Type	Default
7:3	**RESERVED**	RO	-
3	**VCONN Discharge** When set the VBUS Discharge error shall be enabled to place the device in the error state.	R/W	0b
2	**VBUS Discharge** When set, the VBUS Discharge error shall be enabled to place the device in the error state. **Note:** This bit is not applicable when the PPC is selected for VBUS discharge.	R/W	0b
1	**VCONN OCS** When set, the CC OCS error shall be enabled to place the device in the error state.	R/W	0b
0	**CC Back-drive** When set, the CC back-drive error shall be enabled to place the device in the error state.	R/W	0b

9.13.28 VBUS VSAFE0V THRESHOLD REGISTER (VSAFE0V_THR)

Address: 084Eh Size: 16 bits

BITS	DESCRIPTION	TYPE	DEFAULT
15:10	**RESERVED**	RO	-
9:0	**VSAFE0V Threshold (VSAFE0V_THR)** VSAFE0V Threshold register. The lower byte of the threshold must be written before the upper byte. The entire 10-bit threshold is updated when the second write occurs. **Note:** The units of this register are ~2.44 mV from a 2.5V FS.	R/W	Note 9-16

Note 9-16 Defaults values are loaded from OTP. If the OTP is not configured The defaults are defined in Table 9-14.

TABLE 9-14: VSAFE0V_THR DEFAULTS

VSAFE0V_THR	VBUS Threshold	1R/9R Divider	VALUE
VSAFE0V_THR	0.8	0.08	32

APPLICATION NOTE: This register may be dynamically written to by software while the VBUS comparator is enabled, provided the rules for updating defined in the register description are followed.

9.13.29 CFG_SEL*X* MATCH REGISTERS (CFG_SEL*X*_MATCH)

Address: x=0: 0850h Size: 16 bits
x=1: 0852h

Bits	Description	Type	Default
15:0	**Configuration Select Match (CFG_SELX_MATCH)** Indicates which configuration select threshold is matched on the CFG_SEL pin. A match is determined when the measured voltage exceeds the programmed threshold. See Section 9.9.1, "Configuration Selection," on page 82.	RO	0h

9.13.30 CFG_SEL THRESHOLD *X* REGISTERS (CFG_SEL_THR*X*)

Address: x=0: 0854h
x=1: 0856h
x=2: 0858h
x=3: 085Ah
x=4: 085Ch
x=5: 085Eh
x=6: 0860h
x=7: 0862h
x=8: 0864h
x=9: 0866h
x=10: 0868h
x=11: 086Ah
x=12: 086Ch
x=13: 086Eh
x=14: 0870h
x=15: 0872h

Size: 16 bits

A total of 16 thresholds are supported for decoding the resistor value on the **CFG_SEL*x*** pins.

Bits	Description	Type	Default
15:10	**RESERVED**	RO	-
9:0	**CFG_SEL Threshold (CFG_SEL_THR)** **Note:** The units of this register are ~2.44 mV from a 2.5V FS.	R/W	Note 9-17

Note 9-17 The defaults are defined in Table 9-15. Defaults values are loaded from OTP. If the OTP is not configured

TABLE 9-15: CFG_SEL_THR DEFAULTS

CFG_SEL_THR	Default
0	32
1	96
2	160
3	224
4	288
5	352
6	416
7	480
8	544
9	608
10	672
11	736
12	800
13	864
14	928
15	992

9.13.31 CFG_SEL DEBUG REGISTER (CFG_SEL_DBG)

Address: 0874h Size: 8 bits

Bits	Description	Type	Default
7:4	**RESERVED**	RO	-
3:0	**CFG_SEL_VAL** This register stores a snapshot of the highest threshold matched when the device samples the CFG_SEL pin after a system level reset event.	RO	0h

9.13.32 VCONN DISCHARGE THRESHOLD REGISTER (VCONN_DIS_THR)

Address: 0886h Size: 16 bits

Bits	Description	Type	Default
15:10	**RESERVED**	RO	-
9:0	**VCONN Discharge Threshold (VCONN_DIS_THR)** This register defines the threshold used in standalone DFP mode for discharging VCONN. **Note:** The units of this register are ~2.44 mV from a 2.5V/1024.	R/W	41h Note 9-18

Note 9-18 Loaded from OTP. If the OTP is not configured, the default is 41 equates to 150mV after accounting for the R/2R divider.

9.13.33 VCONN DISCHARGE TIME REGISTER (VCONN_DIS_TIME)

Address: 0888h Size: 16 bits

Bits	Description	Type	Default
7:0	**VCONN Discharge Time (VCONN_DIS_TIME)** Defines the amount of time the CC pin supplying VCONN is discharged. **Note:** The units of this register are 10 ms.	R/W	04h

Note 9-19 Loaded from OTP. If the OTP is not configured, the default is 04h.

10.0 BASEBAND CC INTERFACE (BCI)

The device integrates a Baseband CC Interface (BCI) to facilitate USB Power Delivery communication. This module bridges between the PD MAC/BMC and the analog front end. Baseband communication is initiated by the PD MAC, which interfaces to the BCI. The BCI implements the digital functions required to control TX baseband components.

10.1 Baseband TX Data-flow

The key responsibility of the BCI is to generate the wave form required for baseband communication. To this end, the BMC has a group of eight registers that define the Lo-Hi and Hi-Lo transitions for the generated BMC signal.

When instructed to transition from Lo-Hi, the BCI steps through all BB TX Risex Registers (BB_TX_RISEx). Likewise when instructed to transition from Hi-Lo, the BCI steps through all BB TX Fallx Registers (BB_TX_FALLx). The BCI always presents the value at BB_TX_RISE0 or BB_TX_FALL0 first.

APPLICATION NOTE: The user may replicate values if it is desired to use less than twelve unique values for this purpose.

The following steps should be followed to program the BCI for data transmission:

1. Software programs the BB TX Risex Registers (BB_TX_RISEx) and BB TX Fallx Registers (BB_TX_FALLx) to define the slew rate for rising and falling transitions.
2. Software enables the PD MAC.
3. The PD MAC initiates, either via firmware, or autonomously via a data transmission (GoodCRC). The PD MAC instructs the BCI to take the BB TX analog components out of power-down.
4. After a sufficient time elapses for the analog to power up, the PD MAC begins transmission to the BMC encoder which drives the analog components.
5. If the MAC requests a rising transition, the BCI steps through the BB TX Risex Registers (BB_TX_RISEx). Alternatively, if the MAC requests a falling transition, the BCI steps through the BB TX Fallx Registers (BB_TX_-FALLx).
6. When the PD MAC indicates the transmission has completed, the BCI powers down the TX analog components.

10.2 Baseband RX Data-flow

Baseband RX data is received by the BCI from the RX analog front end where it is compared to a threshold programmed by software. The CC RX DAC Value defines the trip point used for reception of baseband data. The field shall be programmed to be 175 mV below the RX Eye center, as defined in the PD Specification for the mode in which the device is operating (Sourcing Power, Sinking Power, Power Neutral).

In order to program the required trip point, the RX DAC Enable bit must be set and the CC RX DAC Value field in the CC RX DAC Control Register (CC_RX_DAC_CTL) must be programmed.

10.3 Baseband CC Interface Registers

This section details the baseband CC interface registers. For an overview of the entire device register map, refer to Section 4.0, "Register Map," on page 18.

TABLE 10-1: BASEBAND CC INTERFACE REGISTER MAP

Address	Register Name (Symbol)
2800h	CC RX DAC Control Register (CC_RX_DAC_CTL)
2802h	CC RX DAC Filter Register (CC_RX_DAC_FILT)
2803h	Reserved for future expansion
2804h	CC TX DAC Filter Register (CC_TX_DAC_FILT)
2805h – 280Fh	Reserved for future expansion
2810h	BB TX Risex Registers (BB_TX_RISEx) x=0
2812h	BB TX Risex Registers (BB_TX_RISEx) x=1
2814h	BB TX Risex Registers (BB_TX_RISEx) x=2
2816h	BB TX Risex Registers (BB_TX_RISEx) x=3
2818h	BB TX Risex Registers (BB_TX_RISEx) x=4
281Ah	BB TX Risex Registers (BB_TX_RISEx) x=5
281Ch	BB TX Risex Registers (BB_TX_RISEx) x=6
281Eh	BB TX Risex Registers (BB_TX_RISEx) x=7
2820h	BB TX Risex Registers (BB_TX_RISEx) x=8
2822h	BB TX Risex Registers (BB_TX_RISEx) x=9
2824h	BB TX Risex Registers (BB_TX_RISEx) x=10
2826h	BB TX Risex Registers (BB_TX_RISEx) x=11
2828h – 282Fh	Reserved for future expansion
2830h	BB TX Fallx Registers (BB_TX_FALLx) x=0
2832h	BB TX Fallx Registers (BB_TX_FALLx) x=1
2834h	BB TX Fallx Registers (BB_TX_FALLx) x=2
2836h	BB TX Fallx Registers (BB_TX_FALLx) x=3
2838h	BB TX Fallx Registers (BB_TX_FALLx) x=4
283Ah	BB TX Fallx Registers (BB_TX_FALLx) x=5
283Ch	BB TX Fallx Registers (BB_TX_FALLx) x=6
283Eh	BB TX Fallx Registers (BB_TX_FALLx) x=7
2840h	BB TX Fallx Registers (BB_TX_FALLx) x=8
2842h	BB TX Fallx Registers (BB_TX_FALLx) x=9
2844h	BB TX Fallx Registers (BB_TX_FALLx) x=10
2846h	BB TX Fallx Registers (BB_TX_FALLx) x=11
2848h – 2BFFh	Reserved for future expansion

Note: RESERVED address space must not be written under any circumstances. Failure to heed this warning may result in untoward operation and unexpected results.

10.3.1 CC RX DAC CONTROL REGISTER (CC_RX_DAC_CTL)

Address: 2800h Size: 16 bits

Bits	Description	Type	Default
15	**RX DAC Enable** 0: Disable the CC RX DAC 1: Enable the CC RX DAC	R/W	0b
14:10	**RESERVED**	RO	-
9:0	**CC RX DAC Value** This register defines the trip point used for reception of baseband data. **Note:** The full scale range of this DAC is 1.8V. **Note:** The DAC must be programmed to be 175mV below the desired RX Eye center. **Note:** For a source, a value of 288 is recommended. For a sink, a value of 146 is recommended.	R/W	0h

10.3.2 CC RX DAC FILTER REGISTER (CC_RX_DAC_FILT)

Address: 2802h Size: 8 bits

Bits	Description	Type	Default
7:2	**RESERVED**	RO	-
1	**Select CC Rx Filter Configuration**	R/W	0b
0	**CC RX DAC Filter Enable**	R/W	0b

10.3.3 CC TX DAC FILTER REGISTER (CC_TX_DAC_FILT)

Address: 2804h Size: 8 bits

Bits	Description	Type	Default
7:5	**RESERVED**	RO	-
4	**CC TX Filter Enable** Enables CC TX filter and driver.	R/W	0
3:0	**CC TX Filter** Selects CC TX filter bandwidth.	R/W	8h

10.3.4 BB TX RISE*X* REGISTERS (BB_TX_RISE*X*)

Address: x=0: 2810h
x=1: 2812h
x=2: 2814h
x=3: 2816h
x=4: 2818h
x=5: 281Ah
x=6: 281Ch
x=7: 281Eh
x=8: 2820h
x=9: 2822h
x=10: 2824h
x=11: 2826h

Size: 16 bits

The BB TX Rise registers define the characteristics of the baseband waveform on rising transitions.

Bits	Description	Type	Default
15:10	**RESERVED**	RO	-
9:0	**BB TX Rise Value** Code presented to the CC TX DAC when implementing the rising transition for a baseband transmission. **Note:** The recommended values for this field are as follows: x=0: 0 x=1: 57 x=2: 115 x=3: 172 x=4: 230 x=5: 287 x=6: 345 x=7: 402 x=8: 460 x=9: 517 x=10: 574 x=11: 631	R/W	0h

10.3.5 BB TX FALL*X* REGISTERS (BB_TX_FALL*X*)

Address: x=0: 2830h, x=1: 2832h, x=2: 2834h, x=3: 2836h, x=4: 2838h, x=5: 283Ah, x=6: 283Ch, x=7: 283Eh, x=8: 2840h, x=9: 2842h, x=10: 2844h, x=11: 2846h

Size: 16 bits

The BB TX Fall registers define the characteristics of the baseband waveform on rising transitions.

Bits	Description	Type	Default
15:10	**RESERVED**	RO	-
9:0	**BB TX Fall Value** Code presented to the CC TX DAC when implementing the falling transition for a baseband transmission. **Note:** The recommended values for this field are as follows: x=0: 631 x=1: 574 x=2: 517 x=3: 460 x=4: 402 x=5: 345 x=6: 287 x=7: 230 x=8: 172 x=9: 115 x=10: 57 x=11: 0	R/W	0h

11.0 POWER DELIVERY MAC

The PD MAC implements certain features of the protocol layer and physical layer of the Universal Serial Bus Power Delivery Specification. On one end the PD MAC interfaces to the software implementing the bulk of protocol and higher level layers and on the other end it interfaces to a BMC encoder / decoder module.

In addition to the normal TX and RX functions, the PD MAC implements the test mode logic defined in the USB PD specification (BIST).

The PD MAC supports the following features:

- Automatic TX Mode for packet framing and CRC32 insertion.
- Raw TX Mode for bit level packet control.
- Automatic GoodCRC response to received messages.
- Automatic BIST Error Count Message in BIST RX Mode.
- GoodCRCTimer implementation.
- Automatic retries with programmable retry count.
- Redundant receive packets automatically dropped in auto response mode.
- 74 byte TX queue.
- 128 byte RX queue.
- Programmable TX Bit-time. Allows for changing operating frequency.
- Programmable preamble length.
- BIST TX and RX logic.
- Programmable TX and RX queue modes - buffer mode and FIFO mode.
- CRC32 generator for TX.
- CRC32 calculator and comparator for RX.

11.1 PD MAC Transmitter

The PD MAC transmitter is comprised of three major blocks:

- TX Queue**:**
  The TX Queue is where software loads the message to be transmitted.
- TX Control**:**
  The TX Control implements the necessary control logic. It is responsible for reading the data from the TX queue and based on the data processing mode (automatic or raw), processing the data to make it suitable (nibbles with control information) for use by the TX Comm. It is also responsible for generating packet framing and terminating the packet in automatic mode, and generating messages for automatic response (GoodCRC and BIST Error Count). TX Control also handles the selection of the SOP type that is to be transmitted.
- TX Comm**:**
  The TX Comm is comprised of a TX CRC generator, a 4b5b encoder, serializer, preamble generator, and TX bit timer. It takes the nibble data, computes and inserts the CRC, 5b encodes, and generates the baseband serial data. Preamble insertion is also performed by this logic.

The TX Queue is where software loads the message to be transmitted.

The TX Control implements the necessary control logic. It is responsible for reading the data from the TX queue and based on the data processing mode (automatic or raw), processing the data to make it suitable (nibbles with control information) for use by the TX Comm. It is also responsible for generating packet framing and terminating the packet in automatic mode, and generating messages for automatic response (GoodCRC and BIST Error Count). TX Control also handles the selection of the SOP type that is to be transmitted.

The TX Comm is comprised of a TX CRC generator, a 4b5b encoder, serializer, preamble generator, and TX bit timer. It takes the nibble data, computes and inserts the CRC, 5b encodes, and generates the baseband serial data. Preamble insertion is also performed by this logic.

The following sub-sections describe the various blocks and sub-blocks in more detail. Some of the supported TX features are also described.

11.1.1 TX QUEUE

The TX Queue is where software loads the message to be transmitted. The following sub-sections describe the TX Queue in more detail.

11.1.1.1 TX Queue Modes of Operation

The TX Queue's write interface (MCU side) has two modes of operation: FIFO Mode and Buffer Mode.

11.1.1.1.1 FIFO Mode

This mode is enabled by setting the EN_FMQ bit in the TX Control Register A (TX_CTL_A). In this mode, software writes data into the TX Queue like a FIFO. Software can use any offset address in the range of 1800h-1849h (although 1800h would be logical to use). The FIFO is 74 entries deep. Data written to the FIFO cannot be read back by software.

11.1.1.1.2 Buffer Mode

This mode is selected when the EN_FMQ bit in the TX Control Register A (TX_CTL_A) is cleared (default after POR). In this mode, software writes data into the TX Queue as if it were writing to registers at different addresses. The offset address range is 1800h-1849h and the buffer has 74 locations.

Only byte access should be used while accessing the TX Queue in this mode. Note that buffer offset address 1800h contains the least-significant-byte (LSB) (byte that goes out first). Buffer offset address 0x049 has the most-significant-byte (MSB) (byte that goes out last).

Software can arbitrarily write or read any location in the buffer. Only one packet can be queued into the TX Queue at one time. Queuing of multiple packets is not supported.

11.1.1.2 TX Queue Data and Processing Modes

The data placed in the TX Queue depends on the selected processing mode. Two modes are supported: “Auto Mode Data Processing” (AMDP) and “Raw Mode Data Processing” (RMDP).

11.1.1.2.1 Auto Mode Data Processing (AMDP)

This mode is selected by clearing the EN_RMDP bit in the TX Control Register A (TX_CTL_A). In this mode, only the message data (header and data objects) to be transmitted is queued in the TX Queue. Further, the data is queued as bytes. Since there is no framing information with the data, the TX Packet Length Register (TX_PKT_LEN) is used to provide information about the length of message data. Hardware uses this information to determine when and where to append the CRC. Packet framing (preamble, SOP, and EOP) is automatically inserted by the hardware.

11.1.1.2.2 Raw Mode Data Processing (RMDP)

This mode is selected by setting the EN_RMDP bit in the TX Control Register A (TX_CTL_A). In this mode, software is responsible for constructing the entire packet including framing, except the preamble and CRC. Since framing involves K-codes, the data placed in the queue is coded with control information. The data in each byte of the queue is treated as a “nibble” of packet data that either needs to be 5b encoded by hardware or a K-codes that should be transmitted without any 5b encoding. There are also some special control bytes to control insertion of CRC and termination of packet.

When tx_queue_data[5] is “0” then tx_queue_data[3:0] is treated as 4b regular data. This 4b data goes through the CRC32 generator for CRC calculation and is encoded to 5b per Table 5-1 of USB PD Specification R1.0 (Table 11-1 shows the 4b5b encoding).

When tx_queue_data[5] is “1” then tx_queue_data[4:0] is treated as 5b K-code (see Table 11-1). This data is eliminated from the CRC calculation and does not go through any further encoding prior to transmission.

The tx_queue_data[7:0] values of 8'hFF and 8'hFE have special meaning. 8'hFF implies that the packet data is done and hardware should now insert the calculated CRC32 (TX_INS_CRC). 8'hFE implies that the transmission should be stopped immediately (TX_STOP).

Note that in RMDP, software can compute its own CRC and place it in the Queue as encoded data. In this case, the TX_INS_CRC code would not be added to the Queue and instead software would proceed with adding the EOP and terminating the transfer with TX_STOP.

Table 11-1 shows how the byte wide queued data is interpreted or encoded by the transmission logic in raw mode.

TABLE 11-1: RAW MODE DATA ENCODING

tx_queue_data[5:0]	Encoded Symbol
6'h00	5'b11110
6'h01	5'b01001
6'h02	5'b10100
6'h03	5'b10101
6'h04	5'b01010
6'h05	5'b01011
6'h06	5'b01110
6'h07	5'b01111

tx_queue_data[5:0]	Encoded Symbol
6'h08	5'b10010
6'h09	5'b10011
6'h0A	5'b10110
6'h0B	5'b10111
6'h0C	5'b11010
6'h0D	5'b11011
6'h0E	5'b11100
6'h0F	5'b11101

tx_queue_data[5:0]	K-Code	Transmitted Data
6'b1_11000	Sync-1	5'b11000
6'b1_10001	Sync-2	5'b10001
6'b1_01101	EOP	5'b01101
6'b1_00111	RST1	5'b00111
6'b1_11001	RST2	5'b11001

tx_queue_data[5:0]	Special Meaning
8'hFE	TX_STOP
8'hFF	TX_INS_CRC

Note that the tx_queue_data[4:0] is passed as-is when tx_queue_data[5] is set to 1. Thus, software can send reserved symbols for error testing. The only caveat is that hardware cannot be used to generate and insert CRC. In this case, software should compute the necessary CRC32 and add it the data packet and skip the 0xFF code in the queue for CRC insertion.

11.1.1.2.3 TX Queue Programming Sequence - AMDP

The following sequence should be used when programming a sequence in AMDP. This example assumes hardware performs packet framing and CRC insertion.

1. Make sure the GO bit in the TX Control Register B (TX_CTL_B) is cleared, i.e., hardware is done with previous TX request.
2. If using FIFO mode, clear the TX Queue WRI pointer by writing a "1" to the RST_TXQ_FIFO_WRI_PTR bit of the TX Control Register B (TX_CTL_B).
3. Write the two header bytes.
4. Write the payload data, if any.
5. Write the number of bytes to the TX Packet Length Register (TX_PKT_LEN).
6. Check to see if it is OK to transmit via the OK_TO_TX bit in the TX Control Register B (TX_CTL_B)).
7. If OK_TO_TX: set the GO bit in the TX Control Register B (TX_CTL_B) to start transmission.
   Else: repeat step 6. Note: If an RX is in progress, the current TX may need to be abandoned.

11.1.1.2.4 TX Queue Programming Sequence - RMDP

The following sequence should be used when programming a sequence in RMDP. This example assumes hardware performs CRC insertion.

1. Make sure the GO bit in the TX Control Register B (TX_CTL_B) is cleared, i.e., hardware is done with previous TX request.
2. If using FIFO mode, clear the TX Queue WRI pointer by writing a “1” to the RST_TXQ_FIFO_WRI_PTR bit of the TX Control Register B (TX_CTL_B).
3. Write the SOP K-Codes to the FIFO *Sync-1, Sync-1, Sync-1, Sync-2).
4. Write two header bytes, splitting each byte into nibbles.
5. Write the payload data, if any, splitting each byte into nibbles.
6. Write the value “0xFF” to insert the CRC.
7. Write the EOP K-Code (EOP).
8. Write the value “0xFE” to terminate the transmission. Failure to write this value will cause transmission to continue indefinitely looping on the TX buffer.
9. Check to see if it is OK to transmit via the OK_TO_TX bit in the TX Control Register B (TX_CTL_B)).
10. If OK_TO_TX: set the GO bit in the TX Control Register B (TX_CTL_B) to start transmission.
    Else: repeat step 9. Note: If an RX is in progress, the current TX may need to be abandoned.

Note:	The value of the TX Packet Length Register (TX_PKT_LEN) is ignored for this mode since hardware knows when to terminate the packet based on detection of TX_STOP control code.

11.1.2 TX CONTROL

The TX Control implements the necessary control logic. It is responsible for reading the data from the TX queue and based on the data processing mode (automatic or raw), processing the data to make it suitable (nibbles with control information) for use by the TX Comm. It is also responsible for generating packet framing and terminating the packet in automatic mode, and generating messages for automatic response (GoodCRC and BIST Error Count). TX Control also handles the selection of the SOP type that is to be transmitted.

11.1.2.1 Transmit Requests

There are four sources of transmit requests:

- GoodCRC Ack from the receiver for a soft-reset
- GoodCRC Ack from the receiver for a normal packet
- BIST error count message from the BIST receiver
- Transmit go from the software

11.1.2.2 Transmit Aborts

Once packet transmission is initiated, the device will power up the analog and then wait for the bus turn-around timers and power-up timers to expire. During this wait, the transmission can be aborted by software and will be aborted by the device, when a good packet (including soft-reset), a hard reset, or a cable reset (if enabled and the SOP type of the pending TX is SOP', SOP'', SOP'_Debug or SOP''_Debug) is received.

Once the bus turn-around time has expired (or if it was already expired) and the power up time is expired, the bus is checked for idle. If the bus is idle, the transmission is started. If the bus is not idle, the transmission is discarded and, typically, an abort status set. This discard can be disabled with the WAIT4LINE_IDLE bit in the TX Control Register A (TX_CTL_A) for all packets and is also disabled for auto response triggered GoodCRC Acks for received soft-resets (unless the feature is disabled via the DIS_SPCL_SR_GCRC_ACK bit).

If the transmission is not discarded due to the bus being non-idle, the device waits for the bus to become idle and then, once again, waits the turn-around time. During the wait for bus idle, the transmission can be aborted by software and will be aborted by the hardware if a good packet, a hard reset or a cable reset (if enabled and the SOP type of the pending TX is SOP', SOP'', SOP'_Debug or SOP''_Debug) is received.

Software issued hard and cable resets are not aborted due to received good packets, hard resets or cable resets.

Note that for a received good packet to abort a pending transmission, the SOP type of the received packet must match the SOP type of the pending transmission, or the SOP type of the received packet must be SOP with the SOP type of the pending transmission being non-SOP (the latter can be disabled via the DIS_SOP_ABRTS_NON_SOP bit in the TX Control Register A (TX_CTL_A)) and, for messages other than Soft-Reset, the received message ID must indicate a non-duplicated packet (Soft-Resets are never considered duplicates). Ping and GoodCRC messages do not cause a transmission to abort.

Separate TX interrupt abort bits and separate abort status registers are provided for software issued and auto-response packets.

11.1.2.3 Transmit Retries

Transmitted packets are retried under two scenarios: bus idle violations, and GoodCRC response timeout.

If the transmission is discarded due to the bus being non-idle, the option exists to retry instead of treating it as an abort. In order for this to occur, the packet must have been initiated by software, not be a hard or cable reset, auto response mode must be enabled, the retry count must be non-zero, and the RETRY_ON_LINE_BUSY bit in the TX Control Register A (TX_CTL_A) must be set. If after the specified number of retries, the transmission failed due to bus busy, the TX_FAILED status is set (not TX_ABORTED). Hard and Cable resets are not retried.

Following the transmission of a software initiated frame (other than Hard and Cable resets), the device will start a timer and wait for a GoodCRC response to be indicated by the receiver. This assumes retries and / or the wait for GoodCRC are enabled.

If a GoodCRC is received with the correct SOP type and message ID, then the wait is finished and the transmission is done. If a GoodCRC is received with the wrong SOP type or message ID, it is ignored by the transmitter (the transmitter is not even notified) and silently dropped by the receiver.

If the wait for CRC timer expires and the remaining retry count is non-zero, the original packet is re-transmitted. If the remaining retry count is zero, then the packet is not retried and a failed status is indicated.

The wait for GoodCRC will be aborted if any of the following are received:

- A hard reset
- A cable reset (if cable reset reception is enabled and the SOP type of the pending TX is SOP', SOP'', SOP'_Debug or SOP''_Debug)
- A soft-reset (if the SOP type of the RX is the same as that of the pending TX or the SOP type of the RX is SOP with the SOP type of the pending transmission being non-SOP)
- A good packet other than a GoodCRC or Ping (if the SOP type of the RX is the same as that of the pending TX or the SOP type of the RX is SOP with the SOP type of the pending transmission being non-SOP (the latter can be disabled via the DIS_SOP_ABRTS_NON_SOP bit in the TX Control Register A (TX_CTL_A)) and the package is not a duplicate message)

The latter two causes are considered to be protocol errors.

The wait for GoodCRC can also be aborted by software. An aborted wait for GoodCRC is not retried.

11.1.2.4 Transmitter Disable

In order to avoid a race condition where the software is currently issuing a transmit and the hardware is receiving a packet, the EN_FWTX bit in the TX Parameters Register A (TX_PARAM_A) is automatically cleared if a hard reset to a cable reset (if enabled - no SOP type checking is done since a transmission may not be pending) has been received (based on the RX_CABLE_RST and RX_HARD_RST bits in the RX Interrupt Status Register (RX_IRQ_STAT)) or if there is any data in the RX FIFO. Using the interrupt and FIFO status (level sensitive) instead of the even occurrence (edge) avoids another race condition where the software has set the EN_FWTX bit just following the event.

11.1.3 TX COMM

The TX Comm is responsible for taking the coded nibbles from TX Control, encoding it if necessary, and serializing to make it ready for transmission. It is responsible for preamble insertion, CRC calculation, and CRC insertion. It also provides the TX clock signal for the BMC encoder.

11.1.3.1 Preamble Insertion

The device automatically adds the alternating “0” and “1” preamble to the transmitted packet. When the transmission of the preamble is completed, data from TX Queue is processed under direction of the TX Control.

The number of preamble bits sent is programmable by the value of the PREAMBLE_LEN bit in the TX Parameters Register B (TX_PARAM_B).

Note:	Setting the PREAMBLE_LEN field to an odd value will cause violation of the USB PD Specification which states “The preamble shall start with a “0” and shall end with a “1”.” An odd value will cause preamble to start with a value of “0” but it will also end in a value of “0.” Setting this field to an odd value may or may not lead to functional issues. Therefore, this field should be set to an even value to remain USB PD Specification compliant.

11.1.3.2 TX CRC32

The transmit CRC is reset at the start of transmission (setting of the GO bit in the TX Control Register B (TX_CTL_B) or by triggering of automatic response by hardware).

The coded nibble data is streamed into the TX CRC32 one nibble at a time for CRC calculation. Only valid D-Code data is used in CRC calculation. All framing data (invalid D-Code) is excluded from CRC calculation.

Note:	The CRC32 algorithm is described in the USB PD Specification.

11.1.3.3 TX Bit Timing

The baseband transmit signal bit time is controlled by the value in the TX Bit-Time Count Register (TX_BITTIME_CNT). This value determines the transmit data rate.

The count value is based on clock frequency and is given by:
((Clock Freq KHz / Bit Rate Kbps) - 1)

For example:

Clock Frequency = 48000KHz (48MHz)
Nominal Bit Rate = 300 Kbps
bit_time_cnt = (48000 / 300) - 1 = 159

11.1.4 AUTOMATIC RESPONSE MODE

The device supports an Automatic Response Mode (not to be confused with automatic-data processing mode). In this mode, the device will automatically send a GoodCRC message upon successful packet reception, or a BIST Error Count message upon reception of a BIST PRBS frame. This mode can be enabled by setting the EN_AUTO_RSP_MODE bit in the TX Control Register A (TX_CTL_A).

During normal auto-response, if the bus if found to be busy when the device attempts to send the auto-response (should not happen, except if there is noise) it will abort the transmission and the AUTO_RSP_ABORTED bit in the TX Interrupt Status Register (TX_IRQ_STAT) will be set. This behavior can be altered by setting the WAIT4LINE_IDLE bit in the TX Control Register A (TX_CTL_A). Setting this bit will force the device to wait until the bus becomes idle (refer to the WAIT4LINE_IDLE bit description for additional information). GoodCRCs and BIST Error Count messages are not retried even if RETRY_ON_LINE_BUSY in TX_CTL_A is set.

In USB Power Delivery Revision 1.0, Soft Reset had a one-strike rule and failure of Soft Reset led to Hard Reset, leading to the link being brought down, therefore the transmission of GoodCRC ACK in response to reception of Soft Reset is handled a little differently. If hardware finds that the line is busy when it attempts to send the GoodCRC ACK it will wait until the line becomes idle and then re-try the transmission. This process will happen indefinitely. This behavior is same as what happens to normal GoodCRC ACK when WAIT4LINE_IDLE bit is set. GoodCRCs are not retried even if the RETRY_ON_LINE_BUSY bit in the TX Control Register A (TX_CTL_A) is set. For USB Power Delivery Revision 3.0, Soft Resets are normally retried, therefore this function should be disabled via the DIS_SPCL_SR_GCRC_ACK bit.

The SOP type that is used for the automatically sent GoodCRC packet is the SOP type of the received packet (parsed by the receiver). The SOP type that is used for the BIST Error Count message, is selected via the TX_SOP_SELECT bit in the TX Parameters Register A (TX_PARAM_A).

For both GoodCRC and BIST Error Count messages, bit 8 in the packet header is taken from either the PORT_POWER_ROLE (for SOPs) or the CABLE_PLUG (for SOP', SOP'' and _debugs) bits within the TX Parameters Register C (TX_PARAM_C), depending on the SOP type that was received. Bit 5 in the packet header is taken from either the PORT_DATA_ROLE bit (for SOPs) or set to zero (for SOP', SOP'' and _debugs), depending on the SOP type that was received.

In order to avoid missing an auto response GoodCRC request, a flag is used. The flag is set with a request pulse from the receiver and is cleared when the response is sent or aborted. Along with the request flag being set, the SOP type and message ID of the request as well as the Soft Reset response indication are saved. In the event that there was a

pending auto response GoodCRC request and another packet was received resulting in another auto response GoodCRC request, the second request will replace the first request, unless the first request had a SOP type of SOP and the second request had a SOP type of SOP'_Debug or SOP''_Debug. This replacement occurs regardless of the setting of the DIS_SOP_ABRTS_NON_SOP bit in the TX Control Register A (TX_CTL_A). Even if a received packet does not abort a pending auto response, the pending auto response may get replaced by a new auto response.

11.1.5 AUTOMATIC RETRY MODE

The device also supports an Automatic Retry Mode. In this mode, when a GoodCRC message is not received in the appropriate time (i.e., before CRCReceiveTimer expires) the current message is retried.

Hardware will retry the message until the retry value specified in the N_RETRY_CNT field in the TX Parameters Register C (TX_PARAM_C) is satisfied. If the N_RETRY_CNT is set to zero, Automatic Retry Mode is disabled.

The number of retries used by the device to complete the transaction are tracked and made available to software via the N_HW_RETRIES field in the TX Status Register (TX_STAT).

An N_RETRY_CNT value of zero implies the message is attempted only once, i.e., hardware will not retry it. N_RETRY_CNT specifies the number of retries so the number of attempts is (N_RETRY_CNT + 1). Stated another way, if you want hardware to make “N” attempts to send a message then the N_RETRY_CNT field must be set to “N-1.”

Reception of Hard Reset or Cable reset (when enabled and the SOP type of the pending TX is SOP', SOP'', SOP'_Debug or SOP''_Debug) from port partner, messages other than GoodCRC or Ping message (if the SOP types of the RX and TX match or if the SOP type of the RX is SOP with the SOP type of TX being non-SOP (the latter can be disabled via the DIS_SOP_ABRTS_NON_SOP bit in the TX Control Register A (TX_CTL_A)) and the packet was not a duplicate), or issuance of abort by software will terminate any pending auto retries.

Detection of bus collision will cause the device to abort the current transfer including any pending auto retries unless the RETRY_ON_LINE_BUSY bit in the TX Control Register A (TX_CTL_A) is set.

Note that disabling of automatic-retry by setting N_RETRY_CNT to zero will also disable the device’s ability to wait for a GoodCRC message before a successful message transmission is indicated. This applies, for example, to the case when Soft Reset is transmitted. To circumvent this limitation, a EXPECT_GOODCRC bit in the TX Parameters Register A (TX_PARAM_A) is available. Setting this bit will cause the device to wait for a GoodCRC in response to a TX message, even when the N_RETRY_CNT field is set to zero.

Note:	Setting N_RETRY_CNT to zero after the GO bit has been set will have no effect for the current transfer.

The expected message ID within the GoodCRC message is set by software via the MSG_ID field in the TX Parameters Register A (TX_PARAM_A). This is compared to the message ID within the received GoodCRC message.

The expected SOP type within the GoodCRC message is set by software via the TX_SOP_SELECT field in the TX Parameters Register A (TX_PARAM_A). This is compared to the SOP type received in the GoodCRC message. In order for the GoodCRC message to be received, it is assumed that the SOP type has been enabled via the RX_SOP_ENABLE field in the RX Control Register B (RX_CTL_B).

11.1.6 IFG TIMER

The TX turn-around timer is used to insure that a minimum bus idle time is guaranteed between packet reception and packet transmission, i.e., sending of GoodCRC message upon successful packet reception.

The value of this timer is programmable via the TX Turnaround Time Register (TX_TA_TIME). The value is specified in uSec. This timer uses the free-running 1us pulse so the value of this register should be 1 more than desired to ensure the minimum time.

11.1.7 CRC RECEIVE TIMER

This timer is enabled when the device is expected to wait for a GoodCRC ACK (i.e., automatic-retry mode is enabled by non-zero value in the N_RETRY_CNT field of the TX Parameters Register C (TX_PARAM_C) or if zero, EXPECT_GOODCRC bit in the TX Parameters Register A (TX_PARAM_A) is set) at the time transmission is requested. This timer can also be enabled for software usage via the EN_CRC_RCV_TMR bit in the RX Control Register A (RX_CTL_A).

If enabled, the timer starts whenever a TX packet transmission stops.

If software aborts a TX packet in between transmission with auto-retry mode enabled, this timer will not be triggered and will be disabled. If software aborts a TX packet with auto-retry disabled, then it should also disable the timer by clearing the EN_CRC_RCV_TMR bit in the RX Control Register A (RX_CTL_A).

The timeout value for CRCReceiveTimer is programmable via the TRECEIVE field in the RX tReceive Time Register (RX_TRECEIVE_TIME) in multiples of 10 uSec.

This timer can be used by software as the BISTReceiveErrorTimer in the BIST TX mode (the device does not automatically time BISTReceiveError) and as the CRCReceiveTimer when not using automatic retry or wait for GoodCRC modes.

11.1.8 ABORTING A TX IN PROCESS

Software can abort a transmission that has already started by issuing an abort via the ABORT bit in the TX Control Register B (TX_CTL_B). The packet will be aborted as follows based on current phase of transmission:

- If packet transmission had not yet started because the device was waiting for the turn-around timer to expire or line to become idle, transmission will be aborted immediately.
- If the current phase is Preamble, the device will complete transfer of the current bit, append EOP, and turn the transmitter off.
- If the current phase is SOP, Data, or CRC, the device will complete transfer of the current nibble, append EOP, and turn the transmitter off.

A Hard Reset should follow this operation, which is the software's responsibility.

11.2 PD MAC Receiver

The PD MAC receiver is comprised of three major blocks:

- RX Queue**:**
  The RX Queue is where software reads the received messages.
- RX Control**:**
  The RX Control implements the necessary control logic. It is responsible for validating the received packet, updating the RX Queue status, and triggering automatic responses, if required.
- RX Comm**:**
  The RX Comm is comprised of the Clock and Data Recovery (CDR), RX DES (de-serializer) (serial-to-parallel converter, 4b5b decoder, and framing detector), RX CRC32 (CRC calculator, receive timer), and other logic to detect valid packet reception.

The RX Queue is where software reads the received messages.

The RX Control implements the necessary control logic. It is responsible for validating the received packet, updating the RX Queue status, and triggering automatic responses, if required.

The RX Comm is comprised of the Clock and Data Recovery (CDR), RX DES (de-serializer) (serial-to-parallel converter, 4b5b decoder, and framing detector), RX CRC32 (CRC calculator, receive timer), and other logic to detect valid packet reception.

The following sub-sections describe the various blocks and sub-blocks in more detail.

11.2.1 RX QUEUE

The decoded RX data (header, data objects, and CRC) is saved into an integrated 128 byte RX FIFO. The RX FIFO that is capable of storing multiple packets and provides read and write pointers.

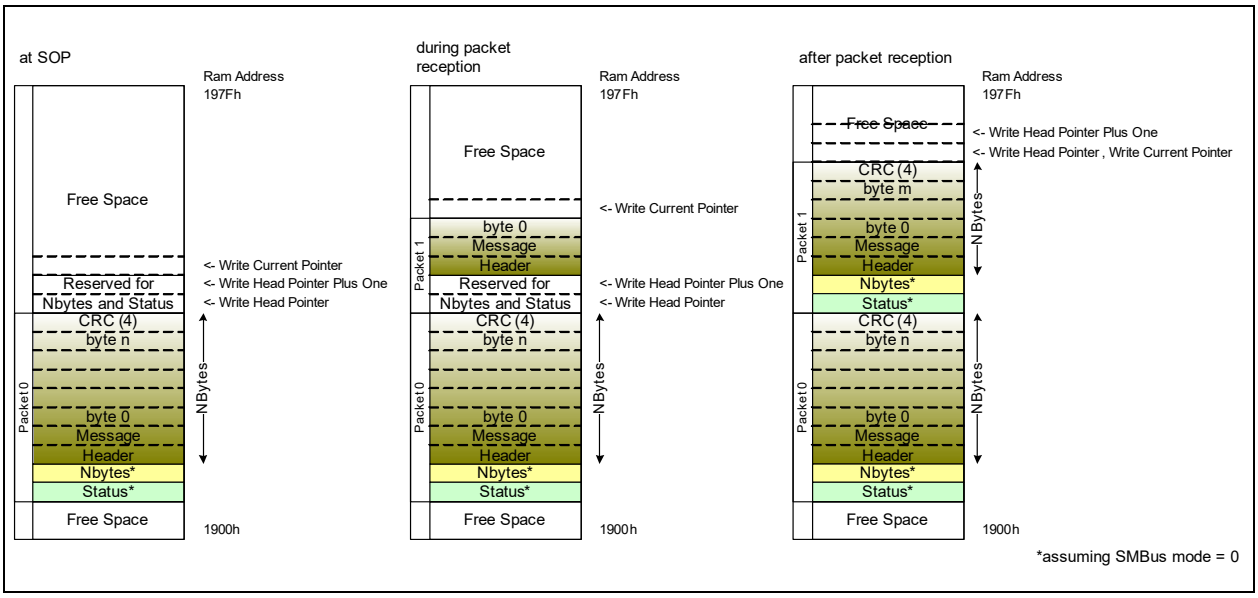
Two bytes are added to the beginning of each packet. Byte 0 holds the packet status (SOP type and the legacy “buffer” valid bit) and byte 1 holds the packet length. An option exists to swap these values (length in byte 0, status in byte 1) and add one to the packet length (to account for the status byte). This option is used for an SMBus like block read where the first byte read indicates the length of the transfer and the remaining bytes (the status and the packet) follow. Note that the packet includes the 4 byte CRC, which is included in the length.

The following sub-sections present details of how the write and read interfaces to the FIFO appear.

11.2.1.1 RX FIFO Write Interface

Figure 11-1 shows how the write interface (from receive hardware's perspective) appears at various points of reception.

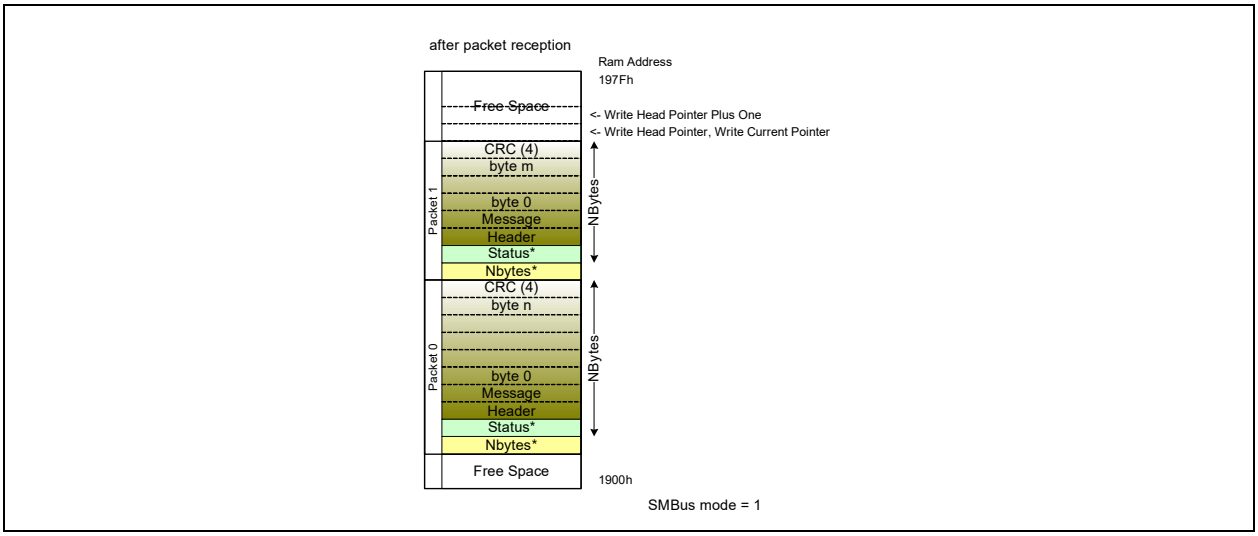
FIGURE 11-1: WRITE INTERFACE VIEW OF RX FIFO



Each RX Packet is comprised of two bytes of control information and the received packet data. The Status byte indicates whether the data is valid and the SOP type that was received. A value of 0x00 in the status byte implies that the data is not valid (including the NBytes field).When bit 0 of the status byte is 1'b1, the Nbytes field indicates the length of the packet and bits 6:4 of the status byte indicates the SOP type that was received. An empty FIFO always returns a value of 0x00. However, the status byte should not be polled, but rather the RX_FIFO_NOT_EMPTY bit in the RX Interrupt Status Register (RX_IRQ_STAT) should be used. The valid bit is retained for legacy compatibility. When software reads the packet data from the FIFO, it should only read number of bytes indicated by Nbytes (in addition to the Nbytes and Status entries). Note that the byte count indicated by NBytes includes the CRC32 data.

When SMBus mode is enabled (via the EN_SMBUS_MODE bit in the RX Control Register A (RX_CTL_A)), the Status and Nbytes fields are swapped and Nbytes is incremented by 1 to include the status byte. The swapping makes the (not-recommended) polling of the Status byte impossible for this mode. When software reads the packet data from the FIFO, it should only read number of bytes indicated by NBytes (in addition to the Nbytes entry). Figure 11-2 details the format while in SMBus mode.

FIGURE 11-2: RX FIFO SMBUS MODE



The packet status and NBytes format are shown in Table 11-2 and Table 11-3, respectively.

TABLE 11-2: RXQ STATUS FIELD

Bits	Description	Type
7	**RESERVED** Always reads 0.	RO
6:4	**RX_SOP_TYPE** This field indicates the SOP type that was received. 000b: SOP 001b: SOP' 010b: SOP" 011b: SOP'_Debug 100b: SOP"_Debug 101b - 111b: Reserved	RO
3:1	**RESERVED**	RO
0	**Status** Status of queue data 0b: Data is invalid 1b: Data is valid	RO

TABLE 11-3: RXQ NUMBER OF BYTES (NBYTES) FIELD

Bits	Description	Type
0	**NBYTES** If EN_SMBUS_MODE is 0, this field represents the number of bytes of valid data in the packet. If EN_SMBUS_MODE is 1, this field represents the number of bytes of valid data in the packet plus 1.	RO

When the device detects valid start-of-packet (SOP) signaling, the current write pointer is set to the write head pointer plus 2. As packet data is received, it is written into the FIFO at the current write pointer and following each write the current write pointer is incremented. The current write pointer points to the next location to be written. Upon reaching the top of the RAM, the current write pointer wraps to 0. The byte count is maintained by the RX control block.

When a valid EOP is received and a good CRC residue is indicated by CRC logic, hardware will write the status (valid bit and received SOP type) to the location addressed by the write head pointer and the number of received bytes to the location addressed by the write head pointer plus one. SMBus mode will swap the data (and increment the number of bytes). Following the writing of the status and length, the write head pointer is updated to the current write pointer and the write head pointer plus one is updated to the current write pointer plus 1 (accounting for the wrap at the top of the RAM).

If a valid EOP is not received and the bus becomes idle, then an abnormal termination is detected. In this case, the packet is dropped, i.e., the device will not update the NBytes location or the Status location.

If a valid EOP is detected but the CRC generator indicates an invalid CRC, then the packet is corrupted. The device will drop the packet and not update the NBytes location or the Status location.

If a more data is received than will fit in the FIFO, then an overrun is detected. The excess data does not get written into the FIFO and the current write pointer is not incremented.

A separate test is made at the start of the packet to determine if there is sufficient room for the status and length bytes. In the event of a “no room for header” condition, an overrun is detected. The excess data does not get written into the FIFO and the current write pointer is not initialized.

If a packet has reached the maximum size allowed (as specified by the RX Maximum Packet Size Register (RX_MAX_SIZE)) and more data is received, then an oversize is detected. The excess data does not get written into the FIFO and the current write pointer is not incremented.

All pointers consist of a RAM address and a wrap indication. The wrap indication is toggled each time the pointer wraps past the top of the RAM. The wrap indication is used in the full, “no room for header” and empty comparisons to distinguish between the write pointers being a full RAM size ahead of the read pointer (full conditions in which case the wrap toggles would be opposite) and the write pointers being equal to the read pointer (empty condition in which case the wrap toggles would be equal).

All pointers and status bits are readable by the software. All pointers and wrap around status bits are writable by the software. Software is responsible for maintaining coherency between the pointer and the wrap toggle values.

The FIFO pointers can be reset by setting and clearing the RST_RECEIVER bit in the RX Control Register A (RX_CTL_A).

11.2.1.1.1 Duplicate/Repeated Packet Handling

When the device is setup for automatic-response (sending of GoodCRC message), it may receive a packet multiple times because the GoodCRC response was lost. If the original packet was received successfully then the subsequent messages of that SOP type with matching message IDs are automatically dropped.

What the device does depends on the CRC status from CRC generator at EOP. If the CRC is good and the message ID from current message matches that of the previous message (stored MSG_ID per SOP type), the device will not update the Packet's status and length locations nor will it update the write head pointer and write head pointer plus one. The FIFO space will be reclaimed at the start of the next packet. The RX_PKT_DROPPED bit in the RX Error Interrupt Status Register (RX_ERR_IRQ_STAT) will be set to indicate this condition. The device will then resend a GoodCRC response. If, however, the CRC generator indicates a bad CRC, or EOP is not received, the device will process the packet like a corrupted packet. The stored MSG_ID will not be updated.

Note that a duplicate packet is dropped if Automatic Retry was enabled, the transmitter was waiting for a GoodCRC and a message other than a GoodCRC was received (a protocol error).

The device maintains individual count of duplicate packets dropped (RX Duplicate Packet Count Register (RX_DUP_PKT_CNT)) and corrupt packets received (RX BadCRC Packet Count Register (RX_BADCRC_PKT_CNT)). If either of these counts exceed 127, the DBG_EVENT bit in the RX Error Interrupt Status Register (RX_ERR_IRQ_STAT) will be set.

11.2.1.1.2 GoodCRC Packet Storing and Dropping

In the case where Automatic Retry is enabled, and the transmitter was waiting for a GoodCRC, the GoodCRC response will be dropped. The device will not update the packet's status and length locations nor will it update the write head pointer and write head pointer plus one. The FIFO space will be reclaimed at the start of the next packet. Note that the GoodCRC response is dropped even if the SOP type or message ID did not match the expected value.

Table 11-4 summarizes the conditions and results for storing or dropping good packets.

TABLE 11-4: CONDITIONS AND RESULTS FOR DROPPING GOOD PACKETS

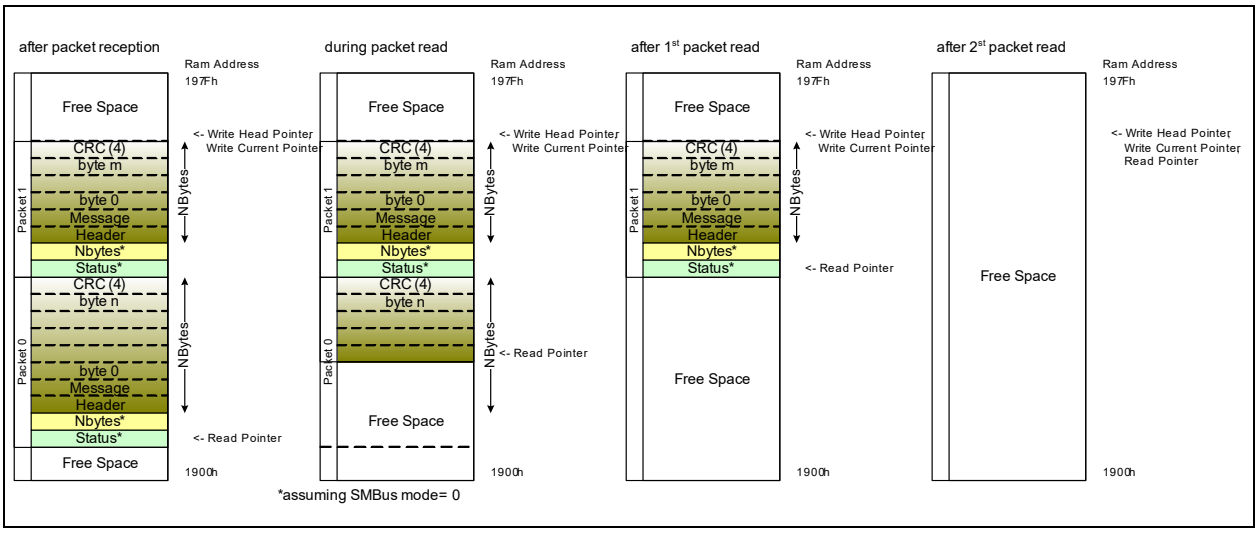
Event		Result	
IF Expecting GoodCRC	Received soft-reset with SOP of RX matching SOP of TX	Packet stored with error status	RX_PKT_STAT[2] (PCOL_ERROR)
	Received GoodCRC with correct SOP type and message ID	Packet dropped	
	Received GoodCRC with wrong SOP type or wrong message ID		
	Received other than GoodCRC or Ping with SOP or RX matching SOP of TX and first message of this SOP type or message ID not matching last message ID of this SOP type (i.e., not a duplicate)	Packet stored, with error status	RX_PKT_STAT[2] (PCOL_ERROR)
ELSE IF Auto Response Enabled	Received soft-reset	Packet stored, message ID saved	
	Received other than GoodCRC or soft-reset and first message of this SOP type		
	Received other than GoodCRC or soft-reset and message ID does not match last message ID of this SOP type		
	Received other than GoodCRC or soft-reset and message ID matches last message ID of this SOP type	Packet dropped with duplicate status	RX_PKT_STAT[1] (DUPLICATE_PACKET)
	Received GoodCRC	Packet stored	
ELSE	Received other than GoodCRC (including soft reset)	packet stored, message ID saved	
	Received GoodCRC	Packet stored	

Note: If auto-response is enabled, any time a packet other than a GoodCRC is stored, a GoodCRC response is transmitted.

11.2.1.2 RX FIFO Read Interface

Figure 11-3 shows how the read interface appears at various points of the software read.

FIGURE 11-3: READ INTERFACE VIEW OF RX FIFO



The RX FIFO read interface always uses a FIFO access mode. The read is destructive, returning the current data and releasing the FIFO space. Although the software may rewind the read pointer for test purposes, there is no guarantee that the data has not been overwritten.

After determining the validity of data in the FIFO queue by reading the RX_FIFO_NOT_EMPTY bit in the RX Interrupt Status Register (RX_IRQ_STAT), software may read the data from the RX FIFO. Software may read the entire packet without the need to recheck the RX_FIFO_NOT_EMPTY bit.

Although written as each byte is received, valid data in the FIFO is only indicated once the entire packet is received and validated.

Depending on the setting of the EN_SMBUS_MODE bit in the RX Control Register A (RX_CTL_A), either the status or the Nbytes field is read first, followed by the other. If SMBus mode is enabled, the Nbytes field is read first and will include the status byte, otherwise it is read second and just indicates the packet length. Note that the packet includes the 4 byte CRC and the length includes this.

As the FIFO is accessed, packet data is read from the FIFO at the read pointer and following each read the read pointer is incremented. The read pointer points to the next location to be read. Upon reaching the top of the RAM, the read pointer wraps to 0.

The FIFO space is released as the packet is read. There is no need for software to specifically release the buffer space.

If software needs to flush a packet without reading the data, it may do so by writing the RX FIFO Read Pointer Register (RX_FIFO_RD_PTR) and RX FIFO Read Pointer Control Bits Register (RX_FIFO_RD_PTR_CTL_BITS). Software should add the appropriate value to the read pointer, taking into account the amount of data already read. Wrap at the top of the RAM must be accounted, for both the read pointer and the RX_FIFO_RX_PTR_WRAP control bit. In order to maintain coherency, the device will hold a write to the RX FIFO Read Pointer Register (RX_FIFO_RD_PTR) in a temporary register and transfer it to the actual register when the RX FIFO Read Pointer Control Bits Register (RX_FIFO_RD_PTR_CTL_BITS) is written.

Although accessed as a FIFO, software can use any offset address in the 128 byte FIFO address range.

Although not recommended, the software can read from the FIFO if it is empty. The returned data will be 0 and the FIFO will not underrun.

All pointers consist of a RAM address and a wrap indication. The wrap indication is toggled each time the pointer wraps past the top of the RAM. The wrap indication is used in the full, “no room for header” and empty comparisons to distinguish between the write pointers being a full RAM size ahead of the read pointer (full conditions in which case the wrap toggles would be opposite) and the write pointers being equal to the read pointer (empty condition in which case the wrap toggles would be equal).

All pointers and status bits are readable by the software. All pointers and wrap around status bits are writable by the software. Software is responsible for maintaining coherency between the pointer and the wrap toggle values.

The FIFO pointers can be reset by setting and clearing the RST_RECEIVER bit in the RX Control Register A (RX_CTL_A).

Each entry in the RX FIFO represents a byte of received data. Data written to the FIFO is the header, the payload, and the CRC32. As stated earlier, only 5b symbols that correspond to valid D-codes are processed and written to the RX Queue. K-Codes cannot be passed to the RX Queue, i.e., there is no “Raw Mode” receive capability. Software will see de-framed packets only.

As mentioned earlier, the data in FIFO is the actual data and not symbols so it can be processed by software directly. Corrupted packets are not written to the Queue.

11.2.2 RX CONTROL

RX Control implements the logic necessary for validating a received packet, updating the RX Queue status, and triggering automatic responses, if required.

11.2.2.1 Duplicate Packet Detection

For duplicate packet detection, the device maintains the last message ID received per SOP type. A flag (first_msg) is maintained per SOP type and indicates if the stored last message ID is valid (there has been a previous packet). This information is used when a good packet is received to detect if the message ID matches the last one received for that SOP type.

The device stores the current message ID into the appropriate last message ID (as selected by the SOP of the current received packet). The message ID and flag for a specified SOP type can be reset via the RX Message ID Stored Register (RX_MSG_ID_STORED). All bits are cleared when software issues a SW_RESET or PD_RESET via the Reset Control Register (RESET_CTL).

11.2.2.2 RX Control of Transmit Abort

Although the actual aborting of pending transmission is done by the Transmit Control block, it is the RX Control block that makes the decision.

For pending transmissions (transmission triggered but waiting for bus idle or the bus turnaround time), the RX control block generates a pulse at the EOP. It checks 1) that the packet is a good packet (good CRC, proper nibble alignment and no symbol errors), 2) that the packet is not a Ping or GoodCRC message, 3a) that the SOP type of the received packet matches the SOP type of the pending transmission or 3b) the SOP type of the received packet is SOP with the SOP type of pending transmission being non-SOP (the latter can be disabled via the DIS_SOP_ABRTS_NON_SOP bit in the TX Control Register A (TX_CTL_A)) and 4) that the packet is not a duplicate or is a Soft-Reset message.

For completed transmissions that are waiting for a GoodCRC to be returned, the RX Control checks 1) that the packet is a good packet (good CRC, proper nibble alignment and no symbol errors), 2) that the packet is a Soft-Reset or is not a GoodCRC message, 3a) that the SOP type of the received packet matches the SOP type of the pending transmission or 3b) the SOP type of the received packet is SOP with the SOP type of pending transmission being non-SOP (the latter can be disabled via the DIS_SOP_ABRTS_NON_SOP bit in the TX Control Register A (TX_CTL_A)) and 4) that the packet is not a duplicate.

11.2.3 RX COMM

The RX Comm is comprised of the clock and data recovery (CDR), RX DES (de-serializer) (serial-to-parallel converter, 4b5b decoder, and framing detector), RX CRC32 (CRC calculator, receive timer), and other logic to detect valid packet reception.

11.2.3.1 CDR

The BMC decodes the received data and outputs the bit stream signal. This signal is applied to the input of CDR module which locks (period and phase) on to the data stream during the preamble phase and outputs the serial data and a capture strobe centered in the data window.

The CDR relies on edge-to-edge time measurement (two bit-times) during the preamble phase for period lock. An average of four successive good period measurements is used for period lock. Valid period measurement is determined by the range specified by the RX Maximum Bit-Rate Bit Period Count Register (RX_BIT_PER_CNT_MAX_BR) and RX Minimum Bit-Rate Bit Period Count Register (RX_BIT_PER_CNT_MIN_BR). Phase lock is achieved by comparing in-phase and quadrature data samples to determine whether the phase needs to be advanced or retarded.

11.2.3.2 RX DES (De-serializer)

The RX DES converts the received serial bit stream from the CDR into 5bit symbols, identifies and generates start-of-packet/end-of-packet/hard and cable reset signals, and decodes the symbols to 4b data.

Note:	Note that only valid D-codes are processed to be written to the RX Queue. An invalid/reserved code is simply dropped.

11.2.3.2.1 SOP Detection

The SOP is detected by comparing the four most recently received 5-bit symbols to the expected SOP pattern. Per the USB PD specification, detection of 3 of the 4 SOP symbols is sufficient. This is done by creating four unique groups each containing three received symbols ({a,b,c}, {a,b,d}, {a,c,d} and {b,c,d}) with each group independently compared to its respective symbols. Any successful match indicates an SOP.

There is a separate compare for each SOP type, enabled by the RX_SOP_ENABLE field in the RX Control Register B (RX_CTL_B). The results are OR'ed. The SOP type is saved and used in the packet processing (auto reply, auto response and queue status). Software is allowed to change the RX_SOP_ENABLE field at any time. The internal value is held during active receive and updated from the register during idle time.

For generation of the RX_SOP interrupts in the RX SOP Interrupt Status Register (RX_SOP_IRQ_STAT), each SOP detection is decoded but is not AND'ed with its RX_SOP_ENABLE.

11.2.3.2.2 Reset Detection

Hard and Cable resets are detected by comparing the four most recently received 5-bit symbols to the expected patterns. Per the USB PD specification, detection of 3 of the 4 symbols is sufficient. This is done by creating four unique groups each containing three received symbols ({a,b,c}, {a,b,d}, {a,c,d} and {b,c,d}) with each group independently compared to its respective symbols. Any successful match indicates a Hard or Cable reset respectively.

Cable Reset must be enabled via the EN_CABLE_RESET bit in the RX Control Register A (RX_CTL_A).

11.2.3.3 RX CRC32

The receive data CRC32 is computed by the RX CRC32 module. The same logic used for TX CRC generation is utilized to calculate the RX CRC.

The coded nibble data is streamed into the RX CRC32 one nibble at a time for CRC calculation. When the packet ends, the output of this module represents the computed CRC for the received packet. A value of 0xC704_DD7B indicates a good CRC. Detection of a valid EOP symbol latches the CRC value. The latched value is compared with the expected good CRC residue value, which is used by the RX Queue to determine if the RX Queue status should be updated to reflect valid data or not.

11.3 PD MAC BIST

The PD MAC incorporates BIST functions as defined in the USB PD Specification. It is comprised of a TX and RX block.

The BIST TX block contains a PRBS (Pseudo Random Binary Sequence) generator, BIST pattern generation logic, and its own bit-timing logic. The SOP type used by TX BIST Test Frames is a 20-bit static vector which is created by multiplexing between the five SOP ordered sets based on a register setting. The resultant 20-bit vector is simply bit selected when the packet is transmitted.

BIST TX is entered when the BIST_EN bit in the BIST Control Register A (BIST_CTL_A) is set along with BIST_RX_EN being clear. The BIST TX mode (TX type) is set using the BIST_TX_MODE field of the BIST Control Register A (BIST_CTL_A) and is started by first using the BIST_TX_RST bit and then by using the BIST_TX_START/BIST_TX-_STATUS bit of the BIST Control Register B (BIST_CTL_B). Continuous BIST transmission (modes 0, 1, 2, 3, and 5) should be stopped by first using the BIST_TX_RST bit before clearing the BIST_EN bit. It may take up to 1 bit time for the TX to stop. BIST_EN should not be cleared until after this time.

The BIST RX block contains a PRBS generator and bit error detection logic. BIST RX is used only during the BIST Receiver Test. BIST RX is entered when the BIST_EN bit in the BIST Control Register A (BIST_CTL_A) is set along with BIST_RX_EN. The BIST error counter is reset using the BIST_CLR_ERR_CNT bit of the BIST Control Register B (BIST_CTL_B).

For information on the BIST registers, refer to Section 11.4.5, "PD MAC BIST Registers," on page 170.

11.4 Power Delivery MAC Registers

This section details the Power Delivery MAC registers. For an overview of the entire device register map, refer to Section 4.0, "Register Map," on page 18.

TABLE 11-5: POWER DELIVERY MAC REGISTER MAP

Address	Register Name (Symbol)
1800h – 1849h	PD MAC TX Queue/FIFO (74 bytes) (See Section 11.1.1, "TX Queue" for additional info)
184Ah – 18FFh	Reserved for future expansion
1900h – 197Fh	PD MAC RX FIFO (128 bytes) (See Section 11.2.1, "RX Queue" for additional info)
1980h – 19FFh	Reserved for future expansion
1A00h	TX Control Register A (TX_CTL_A)
1A01h	TX Status Register (TX_STAT)
1A02h	TX Parameters Register C (TX_PARAM_C)
1A03h	TX Packet Length Register (TX_PKT_LEN)
1A04h	TX Parameters Register A (TX_PARAM_A)
1A05h	TX Control Register B (TX_CTL_B)
1A06h	TX Parameters Register B (TX_PARAM_B)
1A07h	TX Bit-Time Count Register (TX_BITTIME_CNT)
1A08h	TX Turnaround Time Register (TX_TA_TIME)
1A09h	TX Abort Status Register (TX_ABORT_STAT)
1A0Ah	TX Auto-Response Abort Status Register (TX_AR_ABORT_STAT)
1A0Bh	TX Power Up Time Register (TX_POWER_UP_TIME)
1A0Ch	TX Power Down Time Register (TX_POWER_DOWN_TIME)
1A0Dh – 1A3Fh	Reserved for future expansion
1A40h	RX Control Register A (RX_CTL_A)
1A41h	RX Control Register B (RX_CTL_B)
1A42h	RX Maximum Bit-Rate Bit Period Count Register (RX_BIT_PER_CNT_MAX_BR)
1A43h	RX Minimum Bit-Rate Bit Period Count Register (RX_BIT_PER_CNT_MIN_BR)
1A44h	RX Status Register (RX_STAT)
1A45h	RX Packet Status Register (RX_PKT_STAT)
1A46h	RX tReceive Time Register (RX_TRECEIVE_TIME)
1A47h	RX BadCRC Packet Count Register (RX_BADCRC_PKT_CNT)
1A48h	RX Duplicate Packet Count Register (RX_DUP_PKT_CNT)
1A49h	RX Hard Reset Detection Window Register (RX_HR_DET_WINDOW)
1A4Ah	RX Last GoodCRC Packet High Byte Register (RX_LAST_GCRC_PKT_HI)
1A4Bh	RX Last GoodCRC Packet Low Byte Register (RX_LAST_GCRC_PKT_LO)
1A4Ch	RX Message ID Stored Register (RX_MSG_ID_STORED)
1A4Dh	RX Maximum Packet Size Register (RX_MAX_SIZE)
1A4Eh – 1A4Fh	Reserved for future expansion
1A50h	RX FIFO Read Pointer Register (RX_FIFO_RD_PTR)
1A51h	RX FIFO Read Pointer Control Bits Register (RX_FIFO_RD_PTR_CTL_BITS)
1A52h	RX FIFO Write Current Pointer Register (RX_FIFO_WR_CURRENT_PTR)
1A53h	RX FIFO Write Current Pointer Control Bits Register (RX_FIFO_WR_CURRENT_PTR_CTL_BITS)
1A54h	RX FIFO Write Head Pointer Register (RX_FIFO_WR_HEAD_PTR)
1A55h	RX FIFO Write Head Pointer Control Bits Register (RX_FIFO_WR_HEAD_PTR_CTL_BITS)
1A56h	RX FIFO Write Head Pointer Plus One Register (RX_FIFO_WR_HEAD_PTR_PLUS_ONE)

TABLE 11-5: POWER DELIVERY MAC REGISTER MAP (CONTINUED)

Address	Register Name (Symbol)
1A57h	RX FIFO Write Head Pointer Plus One Control Bits Register (RX_FIFO_WR_HEAD_PTR_PLUS_ONE_CTL_BITS)
1A58h – 1A7Fh	Reserved for future expansion
1A80h	MAC Interrupt Status Register (MAC_IRQ_STAT)
1A81h	TX Interrupt Status Register (TX_IRQ_STAT)
1A82h	RX Interrupt Status Register (RX_IRQ_STAT)
1A83h	RX Error Interrupt Status Register (RX_ERR_IRQ_STAT)
1A84h	Power Management Interrupt Status Register (PM_IRQ_STAT)
1A85h	RX SOP Interrupt Status Register (RX_SOP_IRQ_STAT)
1A86h	TX Interrupt Enable Register (TX_IRQ_EN)
1A87h	RX Interrupt Enable Register (RX_IRQ_EN)
1A88h	RX Error Interrupt Enable Register (RX_ERR_IRQ_EN)
1A89h	Power Management Interrupt Enable and Control Register (PM_IRQ_EN)
1A8Ah	RX SOP Interrupt Enable Register (RX_SOP_IRQ_EN)
1A8Bh	Reset Control Register (RESET_CTL)
1A8Ch – 1A9Fh	Reserved for future expansion
1AA0h	BMC RX High Level Full Bit Maximum Time Register (BMC_RX_HI_FB_MAX_TIME)
1AA1h	BMC RX High Level Full Bit Minimum Time Register (BMC_RX_HI_FB_MIN_TIME)
1AA2h	BMC RX Low Level Full Bit Maximum Time Register (BMC_RX_LO_FB_MAX_TIME)
1AA3h	BMC RX Low Level Full Bit Minimum Time Register (BMC_RX_LO_FB_MIN_TIME)
1AA4h	BMC RX High Level Half Bit Maximum Time Register (BMC_RX_HI_HB_MAX_TIME)
1AA5h	BMC RX High Level Half Bit Minimum Time Register (BMC_RX_HI_HB_MIN_TIME)
1AA6h	BMC RX Low Level Half Bit Maximum Time Register (BMC_RX_LO_HB_MAX_TIME)
1AA7h	BMC RX Low Level Half Bit Minimum Time Register (BMC_RX_LO_HB_MIN_TIME)
1AA8h	BMC RX Squelch Assert Time Register (BMC_RX_SQL_ASSERT_TIME)
1AA9h	BMC RX Squelch Hold Time Register (BMC_RX_SQL_HOLD_TIME)
1AAAh – 1AAFh	Reserved for future expansion
1AB0h	BMC TX Bit-Time Count Register (BMC_TX_BITTIME_CNT)
1AB1h	BMC Transition Window Time Register (BMC_TRANSITION_WINDOW_TIME)
1AB2h – 1ABFh	Reserved for future expansion
1AC0h	BIST Control Register A (BIST_CTL_A)
1AC1h	BIST Control Register B (BIST_CTL_B)
1AC2h	BIST Error Count High Register (BIST_ERR_CNT_HI)
1AC3h	BIST Error Count Low Register (BIST_ERR_CNT_LO)
1AC4h – 1AC6h	Reserved for future expansion
1AC7h	BIST RX Status Register (BIST_RX_STAT)
1AC8h – 1BFFh	Reserved for future expansion

Note: RESERVED address space must not be written under any circumstances. Failure to heed this warning may result in untoward operation and unexpected results.

11.4.1 PD MAC TX REGISTERS

The following sub-sections describe the various registers associated with the PD MAC TX logic.

11.4.1.1 TX Control Register A (TX_CTL_A)

Address: 1A00h Size: 8 bits

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6	**RETRY_ON_LINE_BUSY** Normally hardware will abort a TX packet (including all retry attempts) if the line is Busy when hardware attempts to transmit the packet. (Note that transmission of GoodCRC ACK in response for Soft Reset is an exception to this, see the DIS_SPCL_SR_GCRC_ACK bit.) Setting this bit will force hardware to not completely abort the transmission, but rather it will retry the packet up to N_RETRY_CNT times. Failure after N_RETRY_CNT times results in TX_FAILED status in the TX Interrupt Status Register (TX_IRQ_STAT) rather than a TX_ABORTED status. The following conditions will still abort the TX packet: • Hard reset received • Cable reset received (if enabled and the SOP type of the pending TX is SOP', or SOP'', SOP'_Debug or SOP''_Debug) • Good, non-duplicate, non-Ping, non-GoodCRC packet (including soft-reset) received with RX and TX SOP types matching or RX SOP type equaling SOP with TX SOP type not equaling SOP (can be disabled) • Software sets the ABORT bit in the TX Control Register B (TX_CTL_B).	R/W	0b
5	**DIS_SOP_ABRTS_NON_SOP** Normally, when a SOP packet is received, a pending non-SOP packet (including wait for a GoodCRC) will be aborted. When this bit is set, a non-SOP packet will only be aborted by a packet with the same non-SOP type.	R/W	0b
4	**DIS_SPCL_SR_GCRC_ACK** Disable special treatment of GoodCRC ACK in response to Soft Reset. By default, in auto-response mode a line busy condition will not abort GoodCRC ACK transmission in response to Soft Reset reception. Setting this bit will disable this behavior and instead GoodCRC ACK transmission for Soft Reset will be processed the same as GoodCRC ACK for other packets. **Note:** When this bit is set, the WAIT4LINE_IDLE bit will affect GoodCRC ACK for Soft Reset as well.	R/W	0b

Bits	Description	Type	Default
3	**WAIT4LINE_IDLE** Normally hardware will abort a TX packet (including all retry attempt) if the line is Busy when hardware attempts to transmit the packet. (Note that transmission of GoodCRC ACK in response for Soft Reset is an exception to this, see the DIS_SPCL_SR_GCRC_ACK bit.) Setting this bit will force hardware to not abort the transmission, but rather wait until line becomes idle and then transmit the message. The RETRY_ON_LINE_BUSY option has lower priority then this function. The following conditions will still abort the TX packet: • Hard reset received • Cable reset received (if enabled and the SOP type of the pending TX is SOP', or SOP'', SOP'_Debug or SOP''_Debug) • Good, non-duplicate packet, non-Ping, non-GoodCRC (including soft-reset) received with RX and TX SOP types matching or RX SOP type equaling SOP with TX SOP type not equaling SOP (can be disabled) • Software sets the ABORT bit in the TX Control Register B (TX_CTL_B).	R/W	0b
2	**EN_AUTO_RSP_MODE** Enable automatic sending of GoodCRC message and BIST Error Count message by hardware. 0: Auto GoodCRC Message disabled (default) 1: Auto GoodCRC Message enabled **Note:** Disabling Auto Response Mode will also disable automatic retry by hardware, i.e., hardware will ignore the N_RETRY_CNT field of the TX Parameters Register C (TX_PARAM_C).	R/W	0b
1	**EN_FMQ** Enable FIFO Mode queuing on the TX Queue's write interface. 0: Buffer Mode - TX Queue is accessed like a buffer/CSR (Default) 1: FIFO Mode	R/W	0b
0	**EN_RMDP** Enable Raw Mode Data Processing. 0: Automatic Mode Data Processing (default) 1: Enable Raw Mode Data Processing	R/W	0b

11.4.1.2 TX Status Register (TX_STAT)

Address: 1A01h Size: 8 bits

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6:4	**N_HW_RETRIES** Number of attempts by hardware to send a message successfully. This field is automatically cleared when the GO bit in the TX Control Register B (TX_CTL_B) is set. This field is meaningful only if a TX was successful. It is informational only. 0: Initial message was successful. This value must be correlated with the TX_ABORTED bit. If a TX message was aborted, this bit is meaningless. >0: Number of retries performed by the device. The maximum value depends on the N_RETRY_CNT field in the TX Parameters Register C (TX_PARAM_C) register.	RO	000b
3:1	**RESERVED**	RO	-
0	**TX_ACTIVE** Transmitter is active. 0: Transmitter is idle 1: Transmitter is active Software can monitor this bit to determine if transmission is in progress. De-assertion of this bit does not imply successful message transmission, only that the transmitter is no longer active and has stopped transmitting.	RO	0b

11.4.1.3 TX Parameters Register C (TX_PARAM_C)

Address: 1A02h Size: 8 bits

Bits	Description	Type	Default
7	**PORT_DATA_ROLE** This field is used as the Port Data Role field for hardware generated packet header for packets that use SOP signaling. 0: UFP 1: DFP **Note:** SOP' and SOP'' always send a 0 bit for the Port Data Role.	R/W	0b
6:4	**N_RETRY_CNT** Number of retries allowed. Set this to zero to disable automatic hardware retry (i.e., when software will take care of retries). If N_RETRY_CNT is set to zero, but hardware is expected to wait for GoodCRC, the EXPECT_GOODCRC bit should be set in the TX Parameters Register A (TX_PARAM_A), otherwise TX will be considered complete as soon as packet transmission is complete. **Note:** Hard and Cable Resets are not automatically retried since there is no feedback to indicate a failure.	R/W	000b
3	**CABLE_PLUG** This field is used as the Cable Plug field for hardware generated packet header for packets that use SOP' and SOP'' signaling. 0: DFP or UFP 1: Cable Plug	R/W	0b
2	**PORT_POWER_ROLE** This field is used as the Port Power Role field for hardware generated packet header for packets that use SOP signaling. 0: Sink 1: Source	R/W	0b
1:0	**RESERVED**	RO	-

11.4.1.4 TX Packet Length Register (TX_PKT_LEN)

Address: 1A03h Size: 8 bits

Bits	Description	Type	Default
7:6	**RESERVED**	RO	-
5:0	**PKT_LEN** Queued packet length in bytes. Hardware uses this value in Auto Mode to determine when to insert the CRC. PKT_LEN = bytes in header + bytes in data objects **Note:** This applies to Auto Mode queue only and not to Raw Mode queue. Raw Mode queue has special code to insert CRC and terminate the transmission. **Note:** A value of '0' is invalid for this field.	R/W	000000b

11.4.1.5 TX Parameters Register A (TX_PARAM_A)

Address: 1A04h Size: 8 bits

Bits	Description	Type	Default
7	**EXPECT_GOODCRC** This bit is used when the N_RETRY_CNT field in the TX Parameters Register C (TX_PARAM_C) is set to zero but hardware is expected to wait for GoodCRC in response to a TX message. If N_RETRY_CNT is non-zero and EN_AUTO_RSP_MODE in the TX Control Register A (TX_CTL_A) is set, then hardware automatically waits for GoodCRC and this bit has no effect.	R/W	0b
6:4	**TX_SOP_SELECT** This field selects the SOP used for normal TX packets, for Returned BIST Counters packets (in response to RX BIST Test Frames) and for TX BIST Test Frames. This value is also used to verify the expected SOP type in the GoodCRC response when Auto retry is enabled. 000: SOP 001: SOP' 010: SOP" 011: SOP'_Debug 100: SOP"_Debug 101 - 111: Reserved **Note:** For Auto retry mode, it is assumed that the expected SOP type is also enabled in the RX Control Register B (RX_CTL_B).	R/W	000b

Bits	Description	Type	Default
3	**EN_FWTX** Enable software issued transmission. This bit is effective only at the moment the GO bit is set by software and is intended to cover race conditions where it is not possible for software to back off from setting the GO bit in response to a valid RX. This bit is automatically held cleared by hardware if the RX Interrupt Status Register (RX_IRQ_STAT) indicates reception of a Hard Reset, or a Cable Reset, or if there is any data in the RX FIFO and, thus, software must set it prior to each transmission it wants to issue. When this bit is cleared, a software issued transmission will be aborted. **Note:** This is not an Abort bit. Clearing this bit will have no effect on a transmission that has already been issued to hardware. Such transmissions will be aborted automatically by hardware as appropriate. To abort a transmission in progress, use the ABORT bit in the TX Control Register B (TX_CTL_B). **Note:** Setting of this bit does not prevent hardware issued transmissions. **Note:** This bit is reset by assertion of PD_RESET.	R/W	0b
2:0	**MSG_ID** Message ID. This value is used to verify the expected message ID in the GoodCRC response. **Note:** Hardware will not insert this value into the TX message header. For proper normal operation, software must program the same value in bits[11:9] of message header. **Note:** Software must set this to zero when it processes a Hard Reset, Cable Reset or Soft Reset.	R/W	000b

11.4.1.6 TX Control Register B (TX_CTL_B)

Address: 1A05h Size: 8 bits

Bits	Description	Type	Default
7:6	**RESERVED**	RO	-
5	**TX_CABLE_RESET** Setting this bit to 1 together with the GO bit will cause hardware configured for Auto Mode data processing to transmit a Cable Reset. Any data in the TX Queue and related registers is ignored. This bit is not applicable to Raw Mode data processing since Cable Reset packets can be explicitly specified in the TX Queue. It is ignored for Raw Mode. This bit can only be set if the GO bit is not currently set. The TX_HARD_RESET bit has precedence and if set along with this bit, this bit will not set. This bit is cleared by hardware at end of packet transmission. This bit is reset by assertion of PD_RESET. Software can only write 1 to this bit and therefore can not clear it once set. **Note:** Cable Reset is not automatically retried (there is no feedback to indicate a failure). It is up to software to perform Cable Reset retries. **Note:** This bit will be cleared upon completion of any packet transmission, so it should be set only after GO bit is cleared after completion of the prior transmission.	W1S/SC	0b
4	**OK_TO_TX** Channel is idle and it is OK to transmit. Software should monitor this bit before setting the GO bit. 0: Software should not set the GO bit 1: Software can set the GO bit. The following conditions will cause hardware to set this bit to 0: • BMC line idle timer indicating line is busy. • Hardware may be processing an automatic response to a received packet.	RO	1b
3	**RST_TXQ_FIFO_WRI_PTR** Setting this bit will reset the TX Queue's write interface FIFO pointer. When using FIFO Mode Queuing, software must always write a '1' to this bit prior to putting data in the FIFO. Hardware does not latch this bit and it will read back as zero. Hardware may automatically reset the pointer when the GO bit is cleared (i.e., hardware is done with current transmission).	WO	0b

Bits	Description	Type	Default
2	**TX_HARD_RESET** Setting this bit to 1 together with the GO bit will cause hardware configured for Auto Mode data processing to transmit a Hard Reset. Any data in the TX Queue and related registers is ignored. This bit is not applicable to Raw Mode data processing since a Hard Reset packet can be explicitly specified in the TX Queue. It is ignored for Raw Mode. This bit can only be set if the GO bit is not currently set. This bit is cleared by hardware at end of packet transmission. This bit is reset by assertion of PD_RESET. Software can only write 1 to this bit and therefore can not clear it once set. **Note:** Hard Reset is not automatically retried (there is no feedback to indicate a failure). It is up to software to perform Hard Reset retries. **Note:** This bit will be cleared upon completion of any packet transmission, so it should be set only after the GO bit is cleared after completion of the prior transmission.	W1S/SC	0b
1	**ABORT** Setting this bit will abort the current transmission in progress as soon as possible. Writing a '0' has not effect. Hardware does not latch this bit and it will read back as zero. Hardware will finish the current nibble transmission, send EOP, and then terminate transmission. If this bit is set during preamble phase, then the current bit of the preamble will be completed prior to termination. Software should use the TX_EOP status bit in the TX Interrupt Status Register (TX_IRQ_STAT) to determine when the current packet has been terminated.	WO	0b
0	**GO** Writing a 1 to this bit will: • Start transmission of the packet in the TX queue on to the wire or: • Send Hard Reset if the TX_HARD_RESET bit of this register is set and data processing mode is set to Auto (i.e., the EN_RMDP bit in the TX Control Register A (TX_CTL_A) is not set). • Send Cable Reset if the TX_CABLE_RESET bit of this register is set and data processing mode is set to Auto (i.e., the EN_RMDP bit in the TX Control Register A (TX_CTL_A) is not set). Hardware will clear this bit when transmission is complete. This bit is reset by assertion of PD_RESET. Software can only write 1 to this bit and therefore can not clear it once set. Clearing of this bit by hardware does not indicate the message was sent successfully. Hardware may detect bus collision and abort transmission. In this case, the TX_ABORTED bit in the TX Interrupt Status Register (TX_IRQ_STAT) will be set and an interrupt to MCU will be generated.	W1S/SC	0b

11.4.1.7 TX Parameters Register B (TX_PARAM_B)

Address: 1A06h Size: 8 bits

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6:0	**PREAMBLE_LEN** Number of preamble bits to send. Range: 1-127, specification: 64d (default) A value of 0 is invalid and hardware does not check for validity. Software should use the default value for normal operation. Setting this field to an odd value will cause violation of the USB PD Specification which states “The preamble shall start with a “0” and shall end with a “1”.” An odd value will cause preamble to start with a value of “0” but it will also end in a value of “0.” This may or may not lead to functional issues. This field should be set to an even value to remain spec compliant.	R/W	1000000b

11.4.1.8 TX Bit-Time Count Register (TX_BITTIME_CNT)

Address: 1A07h Size: 8 bits

Bits	Description	Type	Default
7:0	**BIT_TIME_CNT** Bit-time counter value. The TX logic uses this value to determine the bit-time for transmission (nominal bit rate is 300Kbps). This register can be used to adjust bit-time for testing purpose (adjusting bitrate). Count value is based on clock frequency and is given by: ((Clock Freq KHz / Bit Rate Kbps) - 1) i.e., Clock Frequency = 48000KHz (48MHz) Nominal Bit Rate = 300 Kbps BIT_TIME_CNT = (48000 / 300) - 1 = 159	R/W	00h

11.4.1.9 TX Turnaround Time Register (TX_TA_TIME)

Address: 1A08h Size: 8 bits

This register controls the IFG timer.

Bits	Description	Type	Default
7:6	**RESERVED**	RO	-
5:0	**TA_TIME** Turnaround time in us. Range: 1-63, specification: 25us A value of 0 is invalid and hardware does not check for validity. Hardware uses a free running 1us pulse generator, thus, the value of this field should be 1 more than desired to ensure the minimum time (26d).	R/W	010101b

11.4.1.10 TX Abort Status Register (TX_ABORT_STAT)

Address: 1A09h Size: 8 bits

This field is used in conjunction with the TX_ABORTED interrupt and provides details for why the TX was aborted. Bits in this register are sticky, i.e., once any bit is set by hardware this field cannot be updated by hardware until software clears the field.

Note: Bits in this register are set by hardware and must be cleared by software. The bits are not mutually exclusive, i.e., more than one bit may be set depending on the circumstance.

Bits	Description	Type	Default
7	**TX_ABORT_STAT[7]** While waiting for GoodCRC, TX aborted due to Protocol Error (i.e., received other than GoodCRC or Ping message while waiting for GoodCRC or Receiving a Hard or Cable Reset.)	R/W1C	0b
6	**TX_ABORT_STAT[6]** During software issued TX, TX was aborted due to RX (including Hard or Cable Reset) while waiting for bus turnaround or waiting for line to go idle.	R/W1C	0b
5	**TX_ABORT_STAT[5]** Auto-retry aborted due to RX (including Hard or Cable Reset) while waiting for bus turnaround or while waiting for line to go idle.	R/W1C	0b
4	**TX_ABORT_STAT[4]** Auto-retry aborted due to line being busy after bus turnaround.	R/W1C	0b
3	**RESERVED**	RO	-
2	**TX_ABORT_STAT[2]** During software issued TX, line was busy after bus turnaround.	R/W1C	0b
1	**TX_ABORT_STAT[1]** Software issued abort while software issued TX was in process.	R/W1C	0b
0	**TX_ABORT_STAT[0]** Software issued GO with TX disabled.	R/W1C	0b

11.4.1.11 TX Auto-Response Abort Status Register (TX_AR_ABORT_STAT)

Address: 1A0Ah Size: 8 bits

This field is used in conjunction with the AUTO_RSP_ABORTED interrupt and provides details for why the TX was aborted. Bits in this register are sticky, i.e., once any bit is set by hardware this field cannot be updated by hardware until software clears the field.

Note: Bits in this register are set by hardware and must be cleared by software. The bits are not mutually exclusive, i.e., more than one bit may be set depending on the circumstance.

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6	**TX_AR_ABORT_STAT[6]** Auto-response aborted due to RX (including Hard or Cable Reset) while waiting for bus turnaround or while waiting for line to go idle.	R/W1C	0b
5:4	**RESERVED**	RO	-
3	**TX_AR_ABORT_STAT[3]** Line was busy after bus turnaround time during auto-response.	R/W1C	0b
2	**RESERVED**	RO	-
1	**TX_AR_ABORT_STAT[1]** Software issued abort while auto-response was in process.	R/W1C	0b
0	**RESERVED**	RO	-

11.4.1.12 TX Power Up Time Register (TX_POWER_UP_TIME)

Address: 1A0Bh Size: 8 bits

Bits	Description	Type	Default
7:0	**TX_POWER_UP_TIME** TX Power Up Time before transmitting (unit is 8 clock cycles). Range: 0-255 (Actual time is 5 clocks larger.) The analog requires a minimum time of 5us to power up. Count value is based on clock frequency and is given by: TX_POWER_UP_TIME = power up time uS x Clock Freq MHz / 8 i.e., Clock Frequency = 48MHz power up time = 5 uS TX_POWER_UP_TIME = 5 x 48 / 8 = 30	R/W	00h

11.4.1.13 TX Power Down Time Register (TX_POWER_DOWN_TIME)

Address: 1A0Ch Size: 8 bits

Bits	Description	Type	Default
7:0	**TX_POWER_DOWN_TIME** TX Power Down Time after transmitting (unit is 8 clock cycles). Range: 0-255 (Actual time is 2 clocks larger.) The analog requires a minimum time of 1us before powering down. Count value is based on clock frequency and is given by: TX_POWER_DOWN_TIME = power down time uS x Clock Freq MHz / 8 i.e., Clock Frequency = 48MHz power down time = 1 uS TX_POWER_DOWN_TIME = 1 x 48 / 8 = 6	R/W	00h

11.4.2 PD MAC RX REGISTERS

The following sub-sections describe the various registers associated with the PD MAC RX logic.

11.4.2.1 RX Control Register A (RX_CTL_A)

Address: 1A40h Size: 8 bits

Bits	Description	Type	Default
7:6	**RESERVED**	RO	-
5	**EN_LOOPBACK** Enable loopback. This bit allows the MAC to receive its own transmission. **Note:** Loopback happens at the pin where TX and RX are connected together.	R/W	0b
4	**EN_RCV** Enable Receiver. Software should set this bit when it is done configuring the receiver. Setting this bit will enable the receiver to start normal operation. If this bit is set while receive activity is in progress it will take effect after current receive activity is over. If this bit is cleared while a receive is being processed it will take effect after current receive is finished. This bit will be automatically cleared by hardware upon reception of Hard Reset or a Cable Reset to prevent normal packet reception. After POR or SW_RESET, Hard and Cable Reset reception is disabled until EN_RCV is set. Thereafter, clearing of EN_RCV (by software or hardware) has no impact to reception of Hard or Cable Reset, i.e., hardware will process Hard or Cable Reset even with EN_RCV cleared. If software wishes to re-initialize receive timing parameters after EN_RCV has been set once it must first reset the UPD_MAC_v2 using SW_RESET. This bit is not reset by assertion of PD_RESET.	R/W	0b
3	**RST_RECEIVER** Setting this bit will reset the receiver Queue. Software should set this bit if receiver hardware may be out of sync with software. Hardware latches this bit so it must be cleared by software to resume receiver operation. Receiver hardware is also reset when PD_RESET bit in RESET_CTL register is set.	R/W	0b

Bits	Description	Type	Default
2	**EN_CRC_RCV_TMR** Enable CRCReceiveTimer. 0: CRCReceiveTimer disabled (default) 1: CRCReceiveTimer Enabled This bit is used by software only when it is not using auto-retry mode. When auto-retry is enabled (EN_AUTO_RSP_MODE in the TX Control Register A (TX_CTL_A) is set and N_RETRY_CNT is non-zero) or when the transmitter is set to wait for the GoodCRC message (EXPECT_GOODCRC is set and N_RETRY_CNT is zero), the CRCReceiveTimer is automatically enabled in hardware. If GoodCRC message is not received prior to timer expiration, then an interrupt will be generated. CRC_RCV_TIMEOUT bit in the TX Interrupt Status Register (TX_IRQ_STAT) will be set. If auto-retry and wait for GoodCRC are disabled, the software can set this bit to enable CRCReceiveTimer prior to starting transmission to help time tReceive. Once enabled, the timer will start automatically at the end of transmission; however, software must disable the timer when a GoodCRC message is received. If not disabled, timer expiration will generate an interrupt. Once expired, the CRCReceiveTimer remains expired until start of another packet transmission when it reloads. This timer can also be used for BIST transmissions as the BISTReceiveErrorTimer.	R/W	0b
1	**EN_SMBUS_MODE** Enable SMBus block read FIFO format. 0: First location in packet is status, second location is packet length 1: First location in packet is packet length + 1, second location is status	R/W	0b
0	**EN_CABLE_RESET** Enable Cable Reset Detection. 0: Cable Reset Detection (default) 1: Cable Reset Detection Enabled	R/W	0b

11.4.2.2 RX Control Register B (RX_CTL_B)

Address: 1A41h Size: 8 bits

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6:2	**RX_SOP_ENABLE** SOP Types enabled to be received. Includes RX BIST Test Frames. Bit 2 - SOP Bit 3 - SOP' Bit 4 - SOP" Bit 5 - SOP'_Debug Bit 6 - SOP"_Debug Software is allowed to change the field at any time. An internal copy is held during active receive and updated from the register during idle time	R/W	00001b
1:0	**RESERVED**	RO	-

11.4.2.3 RX Maximum Bit-Rate Bit Period Count Register (RX_BIT_PER_CNT_MAX_BR)

Address: 1A42h Size: 8 bits

The value of this register, together with value of the RX Minimum Bit-Rate Bit Period Count Register (RX_BIT_PER_CNT_MIN_BR), is used by the CDR to determine valid bit-time (also referenced as bit-period) of the RX signal.

RX_BIT_PER_CNT_MIN_BR > valid bit time > RX_BIT_PER_CNT_MAX_BR.

Bits	Description	Type	Default
7:0	**RX_BIT_PER_CNT_MAX_BR** Bit period count at maximum bit-rate. = (((clock_freq_KHz / max_bit_rate_Kbps) * (100 - tolerance) / 100) - 1) WARNING: If macros are defined in C code, then math must be arranged so that integer overflow does not occur.	R/W	00h

11.4.2.4 RX Minimum Bit-Rate Bit Period Count Register (RX_BIT_PER_CNT_MIN_BR)

Address: 1A43h Size: 8 bits

The value of this register, together with value of the RX Maximum Bit-Rate Bit Period Count Register (RX_BIT_PER_CNT_MAX_BR), is used by the CDR to determine valid bit-time (also referenced as bit-period) of the RX signal.

RX_BIT_PER_CNT_MIN_BR > valid bit time > RX_BIT_PER_CNT_MAX_BR.

Bits	Description	Type	Default
7:0	**RX_BIT_PER_CNT_MIN_BR** Bit period count at minimum bit-rate. = (((clock_freq_KHz / min_bit_rate_Kbps) * (100 + tolerance) / 100) - 1) WARNING: If macros are defined in C code, then math must be arranged so that integer overflow does not occur.	R/W	00h

11.4.2.5 RX Status Register (RX_STAT)

Address: 1A44h Size: 8 bits

Bits	Description	Type	Default
7:2	**RESERVED**	RO	-
1	**RX_IN_PROCESS** Receiver is active. Hardware is receiving a packet, i.e., line is busy and transmitter is not active.	RO	0b
0	**LINE_STATE** 0: Line is idle 1: Line is busy (BMC line idle timer indicating line is busy) If this bit is “1”, a receive might be in progress and software should wait for the message reception to complete. If it is “0”, software may set the GO bit. Hardware will take further action until the very last possible moment to avoid bus contention. Bus contention is still possible and not 100% avoidable. In cases where hardware takes preventative measures to avoid bus contention, the TX packet will be aborted. Software will be informed of this via an interrupt and setting of the TX_ABORTED bit in the TX Interrupt Status Register (TX_IRQ_STAT).	RO	0b

11.4.2.6 RX Packet Status Register (RX_PKT_STAT)

Address: 1A45h Size: 8 bits

This register is applicable only when using auto-response/retry modes. This register is used in conjunction with the following bits to provide additional information regarding the status of a received packet:

- RX_DONE bit in the RX Interrupt Status Register (RX_IRQ_STAT)
- RX_PCOL_ERROR bit in the RX Error Interrupt Status Register (RX_ERR_IRQ_STAT)
- RX_PKT_DROPPED bit in the RX Error Interrupt Status Register (RX_ERR_IRQ_STAT)
- RX_BUF_OVR_RUN bit in the RX Error Interrupt Status Register (RX_ERR_IRQ_STAT)

Bits in this register are sticky, i.e., once any bit is set by hardware this field cannot be updated by hardware until software clears the field.

A value of 0x00 indicates no errors, while a value of 0xFF indicates that the packet status is currently unknown. Other values are interpreted as specified in the bit definitions for this register.

Note: There no status for having received a GoodCRC message with matching MSG_ID. Successful completion of a TX is indication that a valid GoodCRC ACK was received.

Note: Bits in this register are set by hardware and must be cleared by software. The bits are not mutually exclusive, i.e., more than one bit may be set depending on the circumstance.

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6	**RX_PKT_STAT[6]** ODD_NIBBLES. Received packet dropped because it had odd number of nibbles in data phase. Not set if RX_PKT_STAT[3] (RX_OVER_SIZE) is set regardless of bad CRC, symbol errors or odd nibbles. RX_BAD_PKT interrupt is asserted.	R/W1C	0b
5	**RX_PKT_STAT[5]** SYM_ERROR. There was a symbol error in packet but BAD_CRC was not reported. Not set if RX_PKT_STAT[3] (RX_OVER_SIZE) is set regardless of bad CRC, symbol errors or odd nibbles. RX_BAD_PKT interrupt is asserted.	R/W1C	0b
4	**RX_PKT_STAT[4]** RX_FIFO_FULL. Received packet had to be dropped because the FIFO was full either at SOP or during data. RX_BUF_OVR_RUN interrupt is asserted.	R/W1C	0b
3	**RX_PKT_STAT[3]** RX_OVER_SIZE. Received packet had to be dropped because it was larger than the maximum size set in the RX Maximum Packet Size Register. RX_BAD_PKT interrupt is asserted.	R/W1C	0b
2	**RX_PKT_STAT[2]** PCOL_ERROR. Received other than GoodCRC or Ping message when GoodCRC was expected. RX_PCOL_ERROR interrupt is asserted.	R/W1C	0b
1	**RX_PKT_STAT[1]** DUPLICATE_PACKET. Applicable to auto-response mode. RX_PKT_-DROPPED interrupt is asserted.	R/W1C	0b
0	**RX_PKT_STAT[0]** BAD_CRC. RX packet had bad CRC. RX_BAD_PKT interrupt is asserted. This bit is also forced set if symbol error or odd nibble count occurs. It is not set if RX_PKT_STAT[3] (RX_OVER_SIZE) is set, regardless of bad CRC, symbol errors or odd nibbles.	R/W1C	0b

11.4.2.7 RX tReceive Time Register (RX_TRECEIVE_TIME)

Address: 1A46h Size: 8 bits

Bits	Description	Type	Default
7:0	**TRECEIVE** CRCReceiveTimer timeout value (tReceive) in 10's of micro seconds. Timeout value = field value * 10 (us) Range: 0-255, specification: 900-1100us (90-110) **Note:** A value of 0 is invalid and hardware does not check for validity. **Note:** Hardware uses a free running 1us pulse generator, thus, this CRCReceiveTimer may be 1us less than that expected. **Note:** Reset value is programmed to be 1000us (+/- 2%).	R/W	64h

11.4.2.8 RX BadCRC Packet Count Register (RX_BADCRC_PKT_CNT)

Address: 1A47h Size: 8 bits

This register keeps track of the number of packets received with bad CRC (any packet with valid SOP but bad CRC or invalid EOP or symbol error). This register is for debug purpose only.

If enabled, an interrupt can be generated when the count of this register reaches 128.

Bits	Description	Type	Default
7:0	**BADCRC_PKT_CNT** Number of bad packets received, i.e., with Bad CRC or invalid EOP or symbol error. When the value of this register becomes greater than 127, the DBG_EVENT bit in the RX Interrupt Status Register (RX_IRQ_STAT) will be set. Any write to this register will clear all bits. **Note:** This register is not affected by packets received during BIST Receiver Test. All these packets will be corrupted.	R/WAC	00h

11.4.2.9 RX Duplicate Packet Count Register (RX_DUP_PKT_CNT)

Address: 1A48h Size: 8 bits

This register keeps track of the number of the number of duplicate packets received. It is applicable only when Automatic Response Mode is enabled. This register is for debug purpose only and hardware does not use the value for anything.

If enabled, an interrupt can be generated when the count of this register reaches 128.

Bits	Description	Type	Default
7:0	**DUP_PKT_CNT** Number of duplicate packets received. Applicable when Auto Response Mode is enabled. When the value of this register becomes greater than 127, the DBG_EVENT bit in the RX Interrupt Status Register (RX_IRQ_STAT) will be set. Any write to this register will clear all bits.	R/WAC	00h

11.4.2.10 RX Hard Reset Detection Window Register (RX_HR_DET_WINDOW)

Address: 1A49h Size: 8 bits

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6:0	**RX_HR_DET_WINDOW** Value of Hard Reset detection window in number of bits. This field determines when a detected Hard Reset or Cable Reset will be considered valid.	R/W	1011110b

11.4.2.11 RX Last GoodCRC Packet High Byte Register (RX_LAST_GCRC_PKT_HI)

Address: 1A4Ah Size: 8 bits

Bits	Description	Type	Default
7:0	**RX_LAST_GCRC_PKT_HI** Bits [15:8] of received GoodCRC message. This register is only valid when auto-retry is enabled. Since GoodCRC message in auto-retry mode is not stored in the RX queue, this register provides a means for software to examine the last GoodCRC message received.	RO	00h

11.4.2.12 RX Last GoodCRC Packet Low Byte Register (RX_LAST_GCRC_PKT_LO)

Address: 1A4Bh Size: 8 bits

Bits	Description	Type	Default
7:0	**RX_LAST_GCRC_PKT_LO** Bits [7:0] of received GoodCRC message. This register is only valid when auto-retry is enabled. Since GoodCRC message in auto-retry mode is not stored in the RX queue, this register provides a means for software to examine the last GoodCRC message received.	RO	00h

11.4.2.13 RX Message ID Stored Register (RX_MSG_ID_STORED)

Address: 1A4Ch Size: 8 bits

This register indicates if a last message ID was stored for each SOP type. A write to the appropriate bit will clear the last stored message ID for the SOP type indicated. When cleared, the next packet received for the SOP type will not be considered a duplicate. Software should clear the appropriate bit when it processes a Soft Reset.

Bits in this register are set by hardware and must be cleared by software.

This register is also cleared when software issues a SW_RESET or PD_RESET via the Reset Control Register (RESET_CTL).

Bits	Description	Type	Default
7:5	**RESERVED**	RO	-
4:0	**RX_MSG_ID_STORED** This field indicates that a last message ID was stored for the SOP type. Bit 0: SOP Bit 1: SOP' Bit 2: SOP'' Bit 3: SOP'_Debug Bit 4: SOP''_Debug	W1C	00000b

11.4.2.14 RX Maximum Packet Size Register (RX_MAX_SIZE)

Address: 1A4Dh Size: 8 bits

The value of this register sets the maximum receive packet size, including the 4 bytes of CRC.

Bits	Description	Type	Default
7:0	**RX_MAX_SIZE** Maximum receive packet size including the 4 byte CRC.	R/W	22h

11.4.2.15 RX FIFO Read Pointer Register (RX_FIFO_RD_PTR)

Address: 1A50h Size: 8 bits

This register contains the RX FIFO read pointer. Software may update this register to flush a packet based on the known packet length and starting point.

Software must keep the RX FIFO Read Pointer Control Bits Register coherent with this register if software were to update this register.

In order to maintain coherency, writes to this register are held in a temporary register and are transferred to the actual register when the RX FIFO Read Pointer Control Bits Register is written.

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6:0	**RX_FIFO_RD_PTR** This field contains the RX FIFO read pointer.	R/W	0000000b

11.4.2.16 RX FIFO Read Pointer Control Bits Register (RX_FIFO_RD_PTR_CTL_BITS)

Address: 1A51h Size: 8 bits

This register contains the auxiliary control bits of the RX FIFO read pointer.

Software must keep these bits coherent with the RX FIFO Read Pointer if software were to update the latter.

Bits	Description	Type	Default
7	**RX_FIFO_EMPTY** This field indicates that the RX FIFO is empty (RX FIFO Write Head Pointer equals the RX FIFO Read Pointer.)	RO	1b
6:2	**RESERVED**	RO	-
1	**RX_FIFO_RD_PTR_AT_TOP** This field indicates that the RX FIFO read pointer is at the top of the RAM (location 127).	RO	0b
0	**RX_FIFO_RX_PTR_WRAP** This field toggles each time the RX FIFO read pointer wraps past the top of the RAM (location 127). If software updates the RX FIFO read pointer such that it wraps, this bit must be toggled.	R/W	0b

11.4.2.17 RX FIFO Write Current Pointer Register (RX_FIFO_WR_CURRENT_PTR)

Address: 1A52h Size: 8 bits

This register contains the RX FIFO write current pointer. This pointer indicates where the next received byte of data will be stored. This register is for debug and test purpose only and should not normally be written.

Software must maintain coherency between this and the other Write Pointer Registers.

There is no hardware holding register to maintain write coherency between this and the other Write Pointer Registers. It is assumed that this register is only written while there is no active receive.

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6:0	**RX_FIFO_WR_CURRENT_PTR** This field contains the RX FIFO write current pointer.	R/W	0000000b

11.4.2.18 RX FIFO Write Current Pointer Control Bits Register (RX_FIFO_WR_CURRENT_PTR_CTL_BITS)

Address: 1A53h Size: 8 bits

This register contains the auxiliary control bits of the RX FIFO write current pointer. This register is for debug and test purpose only and should not normally be written.

Software must maintain coherency between this and the other Write Pointer Registers.

There is no hardware holding register to maintain write coherency between this and the other Write Pointer Registers. It is assumed that this register is only written while there is no active receive.

Bits	Description	Type	Default
7	**RX_FIFO_FULL** This field indicates that the RX FIFO is full (RX FIFO Write Current Pointer equals the RX FIFO Read Pointer.)	RO	0b
6:2	**RESERVED**	RO	-
1	**RX_FIFO_WR_CURRENT_PTR_AT_TOP** This field indicates that the RX FIFO write current pointer is at the top of the RAM (location 127).	RO	0b
0	**RX_FIFO_WR_CURRENT_PTR_WRAP** This field toggles each time the RX FIFO write current pointer wraps past the top of the RAM (location 127).	R/W	0b

11.4.2.19 RX FIFO Write Head Pointer Register (RX_FIFO_WR_HEAD_PTR)

Address: 1A54h Size: 8 bits

This register contains the RX FIFO write head pointer. This pointer indicates where the next received packet starts. This register is for debug and test purpose only and should not normally be written.

Software must maintain coherency between this and the other Write Pointer Registers.

There is no hardware holding register to maintain write coherency between this and the other Write Pointer Registers. It is assumed that this register is only written while there is no active receive.

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6:0	**RX_FIFO_WR_HEAD_PTR** This field contains the RX FIFO write head pointer.	R/W	0000000b

11.4.2.20 RX FIFO Write Head Pointer Control Bits Register (RX_FIFO_WR_HEAD_PTR_CTL_BITS)

Address: 1A55h Size: 8 bits

This register contains the auxiliary control bits of the RX FIFO write head pointer. This register is for debug and test purpose only and should not normally be written.

Software must maintain coherency between this and the other Write Pointer Registers.

There is no hardware holding register to maintain write coherency between this and the other Write Pointer Registers. It is assumed that this register is only written while there is no active receive.

Bits	Description	Type	Default
7	**RX_FIFO_NO_ROOM_FOR_HEADER** This field indicates that the RX FIFO has no space for the RX header (status and length) (RX FIFO Write Head Pointer equals the RX FIFO Read Pointer or RX FIFO Write Head Pointer Plus One equals the RX FIFO Read Pointer).	RO	0b
6:3	**RESERVED**	RO	-
2	**RX_FIFO_WR_HEAD_PTR_AT_2ND_FROM _TOP** This field indicates that the RX FIFO write head pointer is one below the top of the RAM (location 126).	RO	0b
1	**RX_FIFO_WR_HEAD_PTR_AT_TOP** This field indicates that the RX FIFO write head pointer is at the top of the RAM (location 127).	RO	0b
0	**RX_FIFO_WR_HEAD_PTR_WRAP** This field toggles each time the RX FIFO write head pointer wraps past the top of the RAM (location 127).	R/W	0b

11.4.2.21 RX FIFO Write Head Pointer Plus One Register (RX_FIFO_WR_HEAD_PTR_PLUS_ONE)

Address: 1A56h Size: 8 bits

This register contains the RX FIFO write head pointer plus one. This pointer indicates where the next received packet starts plus one. This register is for debug and test purpose only and should not normally be written.

Software must maintain coherency between this and the other Write Pointer Registers.

There is no hardware holding register to maintain write coherency between this and the other Write Pointer Registers. It is assumed that this register is only written while there is no active receive.

Bits	Description	Type	Default
7	**RESERVED**	RO	-
6:0	**RX_FIFO_WR_HEAD_PTR_PLUS_ONE** This field contains the RX FIFO write head pointer plus one.	R/W	0000001b

11.4.2.22 RX FIFO Write Head Pointer Plus One Control Bits Register (RX_FIFO_WR_HEAD_PTR_PLUS_ONE_CTL_BITS)

Address: 1A57h Size: 8 bits

This register contains the auxiliary control bits of the RX FIFO write head pointer plus one. This register is for debug and test purpose only and should not normally be written.

Software must maintain coherency between this and the other Write Pointer Registers.

There is no hardware holding register to maintain write coherency between this and the other Write Pointer Registers. It is assumed that this register is only written while there is no active receive.

Bits	Description	Type	Default
7:1	**RESERVED**	RO	-
0	**RX_FIFO_WR_HEAD_PTR_PLUS_ONE_WRAP** This field toggles each time the RX FIFO write head pointer plus one wraps past the top of the RAM (location 127).	R/W	0b

11.4.3 PD MAC INTERRUPT STATUS AND ENABLE REGISTERS

The following sub-sections describe the various registers associated with the PD MAC interrupts.

Refer to Section 8.1, "System Interrupts," on page 35 for additional information on system level interrupt control.

11.4.3.1 MAC Interrupt Status Register (MAC_IRQ_STAT)

Address: 1A80h Size: 8 bits

This register reflects the summary status of the various interrupt sources. If any of these bits are set, the MAC_INT interrupt bit in the Interrupt Status Register (INT_STS) will be active (if enabled).

Bits in this register are read only and reflect the combined result of the other PD MAC interrupt status registers.

There is no corresponding enable register, since all interrupts have enables at lower levels of the hierarchy.

Bits	Description	Type	Default
7:5	**RESERVED**	RO	-
4	**RX_SOP_IRQ_STAT** RX SOP status. This bit is set when any bit in the RX SOP Interrupt Status Register (RX_SOP_IRQ_STAT) is set and enabled via the RX SOP Interrupt Enable Register (RX_SOP_IRQ_EN). This bit can only be cleared by clearing or disabling the bit(s) in the RX SOP Interrupt Status Register (RX_SOP_IRQ_STAT).	RO	0b
3	**PM_IRQ_STAT** Power Management status. This bit is set when any bit in the Power Management Interrupt Status Register (PM_IRQ_STAT) is set and enabled via the Power Management Interrupt Enable and Control Register (PM_IRQ_EN). This bit can only be cleared by clearing or disabling the bit(s) in the Power Management Interrupt Status Register (PM_IRQ_STAT).	RO	0b
2	**RX_ERR_IRQ_STAT** RX Error status. This bit is set when any bit in the RX Error Interrupt Status Register (RX_ERR_IRQ_STAT) is set and enabled via the RX Error Interrupt Enable Register (RX_ERR_IRQ_EN). This bit can only be cleared by clearing or disabling the bit(s) in the RX Error Interrupt Status Register (RX_ERR_IRQ_STAT).	RO	0b
1	**RX_IRQ_STAT** RX status. This bit is set when any bit in the RX Interrupt Status Register (RX_IRQ_STAT) is set and enabled via the RX Interrupt Enable Register (RX_IRQ_EN). This bit can only be cleared by clearing or disabling the bit(s) in the RX Interrupt Status Register (RX_IRQ_STAT).	RO	0b
0	**TX_IRQ_STAT** TX status. This bit is set when any bit in the TX Interrupt Status Register (TX_IRQ_STAT) is set and enabled via the TX Interrupt Enable Register (TX_IRQ_EN). This bit can only be cleared by clearing or disabling the bit(s) in the TX Interrupt Status Register (TX_IRQ_STAT).	RO	0b

11.4.3.2 TX Interrupt Status Register (TX_IRQ_STAT)

Address: 1A81h Size: 8 bits

This register reflects the status of various TX_IRQ_STAT sources. Whether an interrupt is generated or not depends on the values in the TX Interrupt Enable Register (TX_IRQ_EN).

Bits in this register are set by hardware and must be cleared by software.

Bits	Description	Type	Default
7	**AUTO_RSP_ABORTED** A TX auto-response message was aborted. Reason for why the TX was aborted may be determined by reading the TX Auto-Response Abort Status Register (TX_AR_ABORT_STAT).	R/W1C	0b
6	**AUTO_RSP_SENT** Hardware has sent an auto-response. This bit is set when hardware is finished sending an auto-response.	R/W1C	0b
5	**OK_TO_TX** It is now OK to transmit. This bit is set when OK_TO_TX bit in TX Control Register B (TX_CTL_B) changes from 0 to 1 indicating that it is OK for software to initiate transmission.	R/W1C	1b
4	**CRC_RCV_TIMEOUT** CRC Receive Timer expired. This is a TX failure. This interrupt should be enabled only when Automatic Retry Mode is disabled. The EN_CRC_RCV_TMR bit in the RX Control Register A (RX_CTL_A) must be set to enable the timer for use by software. **Note:** CRC Receive Timer is also used in BIST Mode to time reception of BIST Error Count message.	R/W1C	0b
3	**TX_EOP** Hardware is done sending a packet. Note this does not mean that a valid EOP framing symbol was sent. It only means that packet transmission ended, i.e., transmitter was turned off. This bit set after each TX attempt. This can be used to signal when a Hard or Cable Reset is finished transmission since it does not have a GoodCRC response so the TX_DONE bit will not be set - or if Auto Retry Mode is disabled in which case the hardware is done when the transmitter turns off. This bit is used by software to trigger internal timers. Software must be carefully when it enables/disables this interrupt because this bit is set after each TX completion.	R/W1C	0b
2	**TX_ABORTED** A TX message was aborted. Reason for why the TX was aborted may be determined by reading the TX Abort Status Register (TX_ABORT_STAT).	R/W1C	0b

Bits	Description	Type	Default
1	**TX_FAILED** Applicable in Auto Retry Mode only (or when GoodCRC timer is enabled). This interrupt indicates a TX message request failed to complete for one of the following reasons: • GoodCRC timer expired • Maximum number of retries exceeded **Note:** Software may enable the GoodCRC timer without enabling Auto Retry. In this case this interrupt will be generated when GoodCRC timer expires. It is software's responsibility to disable the GoodCRC timer on reception of GoodCRC message to prevent generation of this interrupt.	R/W1C	0b
0	**TX_DONE** Applicable in Auto Retry Mode only. A TX message request is successfully completed, i.e., the message is acknowledged with GoodCRC by recipient. This bit is not set for messages that don't get a GoodCRC message, i.e., Hard or Cable Reset. TX_EOP should be used in such situations. When Auto Retry Mode is disabled, TX_EOP should be used to determine when packet is done transmitting.	R/W1C	0b

11.4.3.3 RX Interrupt Status Register (RX_IRQ_STAT)

Address: 1A82h Size: 8 bits

This register reflects the status of various RX_IRQ_STAT sources. Whether an interrupt is generated or not depends on the values in the RX Interrupt Enable Register (RX_IRQ_EN).

Bits in this register are set by hardware and must be cleared by software.

Bits	Description	Type	Default
7	**RX_FIFO_NOT_EMPTY** RX FIFO is not empty. This bit is set whenever the RX FIFO has data to be read. Typically set when a valid packet is finished being received and cleared once all data has been read. **Note:** This bit is read only and is cleared once the FIFO is empty.	RO	0b
6	**LINE_WENT_IDLE** Line is now idle. This bit is set when hardware detects no activity on the PD bus. **Note:** This bit is not line status, it is an event. If line is already idle when this interrupt is enabled, this bit will be set immediately. If line is busy when this interrupt is enabled, this bit will be set when line goes idle. Software can use this to be informed when either TX is over or RX is over. This will only work while clocks are enabled.	R/W1C	1b
5	**RX_CABLE_RST** Cable Reset Message received.	R/W1C	0b
4	**RESERVED**	RO	-
3	**RX_HARD_RST** Hard Reset Message received.	R/W1C	0b

Bits	Description	Type	Default
2	**RX_EOP** Hardware detected valid EOP framing. A bad EOP or abnormal termination will cause packet to be dropped. This bit is set for only of frames that started with an SOP type that was enabled via the RX Control Register B (RX_CTL_B).	R/W1C	0b
1	**RX_SOP** Hardware detected valid SOP framing sequence. This bit is set for any SOP type that is enabled via the RX Control Register B (RX_CTL_B).	R/W1C	0b
0	**RX_DONE** RX message is successfully received. The following conditions apply: • In Auto Retry Mode, expected GoodCRC messages are dropped and therefore not indicated. • In Auto Response Mode, duplicate packets are dropped and therefore not indicated **Note:** In Auto Response Mode, this bit is set before the GoodCRC response is sent and regardless of its successful transmission.	R/W1C	0b

11.4.3.4 RX Error Interrupt Status Register (RX_ERR_IRQ_STAT)

Address: 1A83h Size: 8 bits

This register reflects the status of various RX_ERR_IRQ_STAT sources. Whether an interrupt is generated or not depends on the values in the RX Error Interrupt Enable Register (RX_ERR_IRQ_EN).

Bits in this register are set by hardware and must be cleared by software.

Bits	Description	Type	Default
7	**DBG_EVENT** Debug event occurred: • Count of packets with bad CRC reached limit. Provides indication of link quality. • Count of duplicate packets reached limit. Provides indication dropped GoodCRC packets by receiver. **Note:** Software must read both count registers in order to determine which of the above two conditions caused the event. **Note:** Unlike other RX IRQ status bits, which are triggered by strobes, this bit is persistent, i.e., if the register causing the interrupt is not cleared prior to clearing the interrupt then this bit will set again.	R/W1C	0b
6:4	**RESERVED**	RO	-
3	**RX_BAD_PKT** Received corrupt packet (Bad CRC, Symbol Error, Oversize or ODD nibble count). RX_PKT_STAT[0] (BAD_CRC), RX_PKT_STAT[5] (SYM_ERROR), RX_PKT_STAT[6] (ODD_NIBBLES) or RX_PKT_STAT[3] (RX_OVER_SIZE) in the RX Packet Status Register (RX_PKT_STAT) will be set.	R/W1C	0b
2	**RX_BUF_OVR_RUN** Receiving packet but no buffer space available. In this case, the RX_PKT_STAT[4] (RX_FIFO_FULL) bit in the RX Packet Status Register (RX_PKT_STAT) will be set. Received packet will be dropped.	R/W1C	0b
1	**RX_PCOL_ERROR** Applicable to Auto Retry Mode only. Hardware detected protocol error. i.e., received other than GoodCRC or Ping message when a GoodCRC message was expected. Received message will be placed in RXQ and RX_DONE be will be set. Auto Retry will be aborted and TX_ABORTED bit will be set.	R/W1C	0b
0	**RX_PKT_DROPPED** RX packet is dropped due to: • Duplicate packet in Auto Response Mode • GoodCRC message in Auto Retry Mode	R/W1C	0b

11.4.3.5 Power Management Interrupt Status Register (PM_IRQ_STAT)

Address: 1A84h Size: 8 bits

This register reflects the status of various PM_IRQ_STAT sources. Whether an interrupt is generated or not depends on the values in the Power Management Interrupt Enable and Control Register (PM_IRQ_EN).

Bits in this register are set by hardware and must be cleared by software.

Bits	Description	Type	Default
7:1	**RESERVED**	RO	-
0	**UPD_ACT_WS** USB PD activity detected. Set asynchronously by assertion of the receive input signal at the beginning of any activity. To prevent metastability this interrupt should only be enabled when software plans to stop the clocks. Clearing the USB_PD_ACT_WS_EN bit will actually prevent this bit from being set unlike the other interrupt status bits. This bit is not affected by either UPD_CLK_STOP_EN or UPD_WU_EN. USB PD Activity status is available in other register bits during normal operation.	R/W1C	0b

11.4.3.6 RX SOP Interrupt Status Register (RX_SOP_IRQ_STAT)

Address: 1A85h Size: 8 bits

This register reflects the status of the occurrence of various received SOP types. The SOP type does not need to be enabled via the RX Control Register B (RX_CTL_B) in order for a status bit to be set. Whether an interrupt is generated or not depends on the value of the RX SOP Interrupt Enable Register (RX_SOP_IRQ_EN).

The bits in this register are not cascaded into the RX Interrupt Status Register but instead are enabled and OR'ed into their own interrupt signal output.

Bits in this register are set by hardware and must be cleared by software.

Bits	Description	Type	Default
7:5	**RESERVED**	RO	-
4:0	**RX_SOP** Hardware detected valid SOP framing sequence. Bit 0: SOP Bit 1: SOP' Bit 2: SOP" Bit 3: SOP'_Debug Bit 4: SOP"_Debug	R/W1C	00000b

11.4.3.7 TX Interrupt Enable Register (TX_IRQ_EN)

Address: 1A86h Size: 8 bits

Setting a bit in this register will enable to corresponding source to generate an interrupt. Note that only interrupt generation is affected. The TX Interrupt Status Register (TX_IRQ_STAT) will still reflect the status of the event.

All interrupts are disabled after POR.

Bits	Description	Type	Default
7	**AUTO_RSP_ABORTED_EN** Setting this bit enables the AUTO_RSP_ABORTED interrupt.	R/W	0b
6	**AUTO_RSP_SENT_EN** Setting this bit enables the AUTO_RSP_SENT interrupt.	R/W	0b
5	**OK_TO_TX_EN** Setting this bit enables the OK_TO_TX interrupt.	R/W	0b
4	**CRC_RCV_TIMEOUT_EN** Setting this bit enables the CRC_RCV_TIMEOUT interrupt.	R/W	0b
3	**TX_EOP_EN** Setting this bit enables the TX_EOP interrupt.	R/W	0b
2	**TX_ABORTED_EN** Setting this bit enables the TX_ABORTED interrupt.	R/W	0b
1	**TX_FAILED_EN** Setting this bit enables the TX_FAILED interrupt.	R/W	0b
0	**TX_DONE_EN** Setting this bit enables the TX_DONE interrupt.	R/W	0b

11.4.3.8 RX Interrupt Enable Register (RX_IRQ_EN)

Address: 1A87h Size: 8 bits

Setting a bit in this register will enable to corresponding source to generate an interrupt. Note that only interrupt generation is affected. The RX Interrupt Status Register (RX_IRQ_STAT) will still reflect the status of the event.

All interrupts are disabled after POR.

Bits	Description	Type	Default
7	**RX_FIFO_NOT_EMPTY_EN** Setting this bit enables the RX_FIFO_NOT_EMPTY interrupt.	R/W	0b
6	**LINE_WENT_IDLE_EN** Setting this bit enables the LINE_WENT_IDLE interrupt.	R/W	0b
5	**RX_CABLE_RST_EN** Setting this bit enables the RX_CABLE_RST interrupt.	R/W	0b
4	**RESERVED**	RO	-
3	**RX_HARD_RST_EN** Setting this bit enables the RX_HARD_RST interrupt.	R/W	0b
2	**RX_EOP_EN** Setting this bit enables the RX_EOP interrupt.	R/W	0b
1	**RX_SOP_EN** Setting this bit enables the RX_SOP interrupt.	R/W	0b
0	**RX_DONE_EN** Setting this bit enables the RX_DONE interrupt.	R/W	0b

11.4.3.9 RX Error Interrupt Enable Register (RX_ERR_IRQ_EN)

Address: 1A88h Size: 8 bits

Setting a bit in this register will enable to corresponding source to generate an interrupt. Note that only interrupt generation is affected. The RX Error Interrupt Status Register (RX_ERR_IRQ_STAT) will still reflect the status of the event.

All interrupts are disabled after POR.

Bits	Description	Type	Default
7	**DBG_EVENT_EN** Setting this bit enables the DBG_EVENT interrupt.	R/W	0b
6:4	**RESERVED**	RO	-
3	**RX_BAD_PKT_EN** Setting this bit enables the RX_BAD_PKT interrupt.	R/W	0b
2	**RX_BUF_OVR_RUN_EN** Setting this bit enables the RX_BUF_OVR_RUN interrupt.	R/W	0b
1	**RX_PCOL_ERROR_EN** Setting this bit enables the RX_PCOL_ERROR interrupt.	R/W	0b
0	**RX_PKT_DROPPED_EN** Setting this bit enables the RX_PKT_DROPPED interrupt.	R/W	0b

11.4.3.10 Power Management Interrupt Enable and Control Register (PM_IRQ_EN)

Address: 1A89h Size: 8 bits

Setting a bit in this register will enable to corresponding source to generate an interrupt. Note that only interrupt generation is affected. The Power Management Interrupt Status Register (PM_IRQ_STAT) will still reflect the status of the event.

All interrupts are disabled after POR. This register also enables clock start and stop requests.

Bits	Description	Type	Default
7:4	**RESERVED**	RO	-
3	**UPD_CLK_STOP_EN** If this bit is set, the MAC, upon going idle, will request that the system clock be stopped. If this bit is cleared, software is responsible to return the device to sleep mode. This bit does not affect the wake up interrupt (UPD_ACT_WS) generation. **Note:** This bit does not affect the MAC's indication that it requires the clock. Even if this bit is cleared, the MAC, once it is idle, will indicate that it does not need the clock.	R/W	0b

Bits	Description	Type	Default
2	**UPD_WU_EN** If this bit is set, the MAC, upon detecting receive activity, will request that the system clock be started. This bit does not affect the wake up interrupt (UPD_ACT_WS) generation. **Note:** Once clocks have been started, this bit does not affect the MAC's indication that it requires the clock. Even if this bit is cleared, the MAC, once it is non-idle (i.e. squelch is active), will indicate that it requires the clock. Only the period between the asynchronous wake up and the start of squelch is affected.	R/W	0b
1	**RESERVED**	RO	-
0	**USB_PD_ACT_WS_EN** Setting this bit enables the UPD_ACT_WS interrupt.	R/W	0b

11.4.3.11 RX SOP Interrupt Enable Register (RX_SOP_IRQ_EN)

Address: 1A8Ah Size: 8 bits

Setting a bit in this register will enable to corresponding source to generate an interrupt. Note that only interrupt generation is affected. The RX SOP Interrupt Status Register (RX_SOP_IRQ_STAT) will still reflect the status of the event.

All interrupts are disabled after POR.

Bits	Description	Type	Default
7:5	**RESERVED**	RO	-
4:0	**RX_SOP_EN** Setting a bit enables the corresponding RX_SOP interrupt. Bit 0: SOP Bit 1: SOP' Bit 2: SOP" Bit 3: SOP'_Debug Bit 4: SOP"_Debug	R/W	00000b

11.4.3.12 Reset Control Register (RESET_CTL)

Address: 1A8Bh Size: 8 bits

Bits	Description	Type	Default
7	**SW_RESET** Software generated PD MAC reset. Software can use this bit to reset the PD MAC to its POR state. Software should toggle this bit, i.e., set it and clear it. After clearing this bit, software will have to re initialize the PD MAC, including the BMC registers. This bit is reset by **RESET_N**.	R/W	0b
6	**BMC_SW_RESET** Software generated BMC reset. Software can use this bit to reset the BMC encoder/decoder to its POR state. Software should toggle this bit, i.e., set it and clear it. The BMC registers are NOT affected by this reset.	R/W	0b
5:1	**RESERVED**	RO	-
0	**PD_RESET** Software should set this bit when it is done processing a PD Hard Reset, Cable Reset or Cable Reset so hardware can return to PD Reset state. Software should toggle this bit, i.e., set it and clear it. The following are reset by PD_RESET: • RX_CTL FSM goes to POR state except Hard and Cable Reset processing remains enabled if it was previously enabled by enabling the receiver. • TX_CTL FSM goes to POR state. • TX parallel to serial and 4b5b encoder • TX Control Register B (TX_CTL_B) GO, TX_HARD_RESET, TX_CABLE_RESET bits • TX Parameters Register A (TX_PARAM_A) EN_FWTX bit The following are not reset: • All PD MAC registers, except as listed above. • Various TX packet counters are NOT reset (NUM_PD_PACKETS_SENT, NUM_MISSING_GCRC_MSG, NUM_MISSING_GCRC_MSG_SENT). • Various RX packet counters are NOT reset (RX_BADCRC_PKT_CNT, RX_DUP_PKT_CNT, NUM_PD_PACKETS_RECVD). • RX FIFO is NOT reset. • RX CDR, 4b5b decoder, and CRC are not reset. • TX IFG timer and CRC are not reset. • BIST and Power Management functions are not reset. • BMC encoder/decoder.	R/W	0b

11.4.4 PD MAC BMC ENCODER/DECODER REGISTERS

The following sub-sections describe the various registers associated with the PD MAC BMC Encoder/Decoder.

11.4.4.1 BMC RX High Level Full Bit Maximum Time Register (BMC_RX_HI_FB_MAX_TIME)

Address: 1AA0h Size: 8 bits

The value of this register together with the value of the BMC RX High Level Full Bit Minimum Time Register (BMC_RX_HI_FB_MIN_TIME) is used by the BMC decoder to determine the valid full bit time for a high level RX signal. This time should take into account both the bit rate range and duty cycle distortion of the received signal.

BMC_RX_HI_FB_MAX_TIME > valid high time > BMC_RX_HI_FB_MIN_TIME

Bits	Description	Type	Default
7:0	**BMC_RX_HI_FB_MAX_TIME** Time in clock cycles at minimum bit-rate and worst duty cycle distortion. = (((clock_freq_KHz / min_bit_rate_Kbps) * (100 + tolerance) / 100) - 1) **Note:** A value of 202 (decimal) is recommended for this field.	R/W	00h

11.4.4.2 BMC RX High Level Full Bit Minimum Time Register (BMC_RX_HI_FB_MIN_TIME)

Address: 1AA1h Size: 8 bits

The value of this register together with the value of the BMC RX High Level Full Bit Maximum Time Register (BMC_RX_HI_FB_MAX_TIME) is used by the BMC decoder to determine the valid full bit time for a high level RX signal. This time should take into account both the bit rate range and duty cycle distortion of the received signal.

BMC_RX_HI_FB_MAX_TIME > valid high time > BMC_RX_HI_FB_MIN_TIME

BMC_RX_HI_FB_MIN_TIME > BMC_RX_HI_HB_MAX_TIME

Bits	Description	Type	Default
7:0	**BMC_RX_HI_FB_MIN_TIME** Time in clock cycles at maximum bit-rate and worst duty cycle distortion. = (((clock_freq_KHz / max_bit_rate_Kbps) * (100 - tolerance) / 100) - 1) **Note:** A value of 116 (decimal) is recommended for this field.	R/W	00h

11.4.4.3 BMC RX Low Level Full Bit Maximum Time Register (BMC_RX_LO_FB_MAX_TIME)

Address: 1AA2h Size: 8 bits

The value of this register together with the value of the BMC RX Low Level Full Bit Minimum Time Register (BMC_RX_LO_FB_MIN_TIME) is used by the BMC decoder to determine the valid full bit time for a low level RX signal. This time should take into account both the bit rate range and duty cycle distortion of the received signal.

BMC_RX_LO_FB_MAX_TIME > valid low time > BMC_RX_LO_FB_MIN_TIME

Bits	Description	Type	Default
7:0	**BMC_RX_LO_FB_MAX_TIME** Time in clock cycles at minimum bit-rate and worst duty cycle distortion. = (((clock_freq_KHz / min_bit_rate_Kbps) * (100 + tolerance) / 100) - 1) **Note:** A value of 202 (decimal) is recommended for this field.	R/W	00h

11.4.4.4 BMC RX Low Level Full Bit Minimum Time Register (BMC_RX_LO_FB_MIN_TIME)

Address: 1AA3h Size: 8 bits

The value of this register together with the value of the BMC RX Low Level Full Bit Maximum Time Register (BMC_RX-_LO_FB_MAX_TIME) is used by the BMC decoder to determine the valid full bit time for a low level RX signal. This time should take into account both the bit rate range and duty cycle distortion of the received signal.

BMC_RX_LO_FB_MAX_TIME > valid low time > BMC_RX_LO_FB_MIN_TIME

BMC_RX_LO_FB_MIN_TIME > BMC_RX_LO_HB_MAX_TIME

Bits	Description	Type	Default
7:0	**BMC_RX_LO_FB_MIN_TIME** Time in clock cycles at maximum bit-rate and worst duty cycle distortion. = (((clock_freq_KHz / max_bit_rate_Kbps) * (100 - tolerance) / 100) - 1) **Note:** A value of 116 (decimal) is recommended for this field.	R/W	00h

11.4.4.5 BMC RX High Level Half Bit Maximum Time Register (BMC_RX_HI_HB_MAX_TIME)

Address: 1AA4h Size: 8 bits

The value of this register together with the value of the BMC RX High Level Half Bit Minimum Time Register (BMC_RX-_HI_HB_MIN_TIME) is used by the BMC decoder to determine the valid half bit time for a high level RX signal. This time should take into account both the bit rate range and duty cycle distortion of the received signal.

BMC_RX_HI_HB_MAX_TIME > valid high time > BMC_RX_HI_HB_MIN_TIME

Bits	Description	Type	Default
7:0	**BMC_RX_HI_HB_MAX_TIME** Time in clock cycles at minimum bit-rate and worst duty cycle distortion. =((1 / 2) * ((clock_freq_KHz / min_bit_rate_Kbps)*(100 + tolerance) / 100) - 1) **Note:** A value of 110 (decimal) is recommended for this field.	R/W	00h

11.4.4.6 BMC RX High Level Half Bit Minimum Time Register (BMC_RX_HI_HB_MIN_TIME)

Address: 1AA5h Size: 8 bits

The value of this register together with the value of the BMC RX High Level Half Bit Maximum Time Register (BMC_RX-_HI_HB_MAX_TIME) is used by the BMC decoder to determine the valid half bit-time for a high level RX signal. This time should take into account both the bit rate range and duty cycle distortion of the received signal.

BMC_RX_HI_HB_MAX_TIME > valid high time > BMC_RX_HI_HB_MIN_TIME

BMC_RX_HI_FB_MIN_TIME > BMC_RX_HI_HB_MAX_TIME

Bits	Description	Type	Default
7:0	**BMC_RX_HI_HB_MIN_TIME** Time in clock cycles at maximum bit-rate and worst duty cycle distortion. =((1 / 2) * ((clock_freq_KHz / max_bit_rate_Kbps)*(100 - tolerance) / 100) - 1) **Note:** A value of 48 (decimal) is recommended for this field.	R/W	00h

11.4.4.7 BMC RX Low Level Half Bit Maximum Time Register (BMC_RX_LO_HB_MAX_TIME)

Address: 1AA6h Size: 8 bits

The value of this register together with the value of the BMC RX Low Level Half Bit Minimum Time Register (BMC_RX_LO_HB_MIN_TIME) is used by the BMC decoder to determine the valid half bit-time for a low level RX signal. This time should take into account both the bit rate range and duty cycle distortion of the received signal.

BMC_RX_LO_HB_MAX_TIME > valid low time > BMC_RX_LO_HB_MIN_TIME

Bits	Description	Type	Default
7:0	**BMC_RX_LO_HB_MAX_TIME** Time in clock cycles at minimum bit-rate and worst duty cycle distortion. =((1 / 2) * ((clock_freq_KHz / min_bit_rate_Kbps)*(100 + tolerance) / 100) - 1) **Note:** A value of 110 (decimal) is recommended for this field.	R/W	00h

11.4.4.8 BMC RX Low Level Half Bit Minimum Time Register (BMC_RX_LO_HB_MIN_TIME)

Address: 1AA7h Size: 8 bits

The value of this register together with the value of the BMC RX Low Level Half Bit Maximum Time Register (BMC_RX_LO_HB_MAX_TIME) is used by the BMC decoder to determine the valid half bit-time for a low level RX signal. This time should take into account both the bit rate range and duty cycle distortion of the received signal.

BMC_RX_LO_HB_MAX_TIME > valid low time > BMC_RX_LO_HB_MIN_TIME

BMC_RX_LO_FB_MIN_TIME > BMC_RX_LO_HB_MAX_TIME

Bits	Description	Type	Default
7:0	**BMC_RX_LO_HB_MIN_TIME** Time in clock cycles at maximum bit-rate and worst duty cycle distortion. =((1 / 2) * ((clock_freq_KHz / max_bit_rate_Kbps)*(100 - tolerance) / 100) - 1) **Note:** A value of 48 (decimal) is recommended for this field.	R/W	00h

11.4.4.9 BMC RX Squelch Assert Time Register (BMC_RX_SQL_ASSERT_TIME)

Address: 1AA8h Size: 8 bits

The value of this register is used by the BMC decoder to determine when there is a valid signal. If three edges of the receive signal are seen within this time, then the “squelch” goes active. This time should take into account the minimum bit rate but does not need to account for the duty cycle distortion of the received signal. A typical assert time would be 3 maximum half bit times pulse desired tolerance.

Bits	Description	Type	Default
7:0	**BMC_RX_SQL_ASSERT_TIME** Time in clock cycles x 4 at minimum bit-rate. assert time uS = (BMC_RX_SQL_ASSERT_TIME x 4 + 1) / clock_freq MHz A typical assert time would be 3 maximum half bit times plus desired tolerance. **Note:** A value of 70 (decimal) is recommended for this field.	R/W	00h

11.4.4.10 BMC RX Squelch Hold Time Register (BMC_RX_SQL_HOLD_TIME)

Address: 1AA9h Size: 8 bits

The value of this register is used by the BMC decoder to provide time for the CDR circuit to output the final bit of data. A typical hold time would be 1 maximum half bit time pulse desired tolerance.

Note: The Squelch Drop time is based on the BMC RX Maximum High and Low Time registers.

Bits	Description	Type	Default
7:0	**BMC_RX_SQL_HOLD_TIME** Time in clock cycles x 4 at minimum bit-rate. hold time uS = (BMC_RX_SQL_HOLD_TIME x 4 + 1) / clock_freq MHz typical value = ((1 / 2) * ((clock_freq_KHz / min_bit_rate_Kbps) * (100 + tolerance) / 100) - 1) / 4 **Note:** A value of 23 (decimal) is recommended for this field.	R/W	00h

11.4.4.11 BMC TX Bit-Time Count Register (BMC_TX_BITTIME_CNT)

Address: 1AB0h Size: 8 bits

Bits	Description	Type	Default
7:0	**BMC_TX_BIT_TIME_CNT** Bit-time counter value. The BMC encoder uses this value to determine the bit-timing for the last bit of data. This register can be used to adjust bit-time for testing purpose (adjusting bitrate). Count value is based on clock frequency and is given by: ((Clock Freq KHz / Bit Rate Kbps) - 1) i.e., Clock Frequency = 48000KHz (48MHz) Nominal Bit Rate = 300 Kbps bit_time_cnt = (48000 / 300) - 1 = 159 **Note:** A value of 159 (decimal) is recommended for this field.	R/W	00h

11.4.4.12 BMC Transition Window Time Register (BMC_TRANSITION_WINDOW_TIME)

Address: 1AB1h Size: 8 bits

The value of this register is used by the BMC decoder to determine when the line is IDLE. Detection is active when 3 transitions occur at the receiver within a time window of BMC_TRANSITION_WINDOW_TIME. After waiting BMC_TRANSITION_WINDOW_TIME without detecting 3 transitions, the bus is considered idle. The USB PD Specification calls for 12uS minimum and 20uS maximum.

Bits	Description	Type	Default
7:0	**BMC_TRANSITION_WINDOW_TIME** Time in clock cycles x 4 transition time uS = (BMC_TRANSITION WINDOW_TIME x 4) / clock_freq MHz Per the USB PD Specification, this value should be set between 12uS and 20uS. This would be a value between 144 and 240 at 48MHz. **Note:** Unlike the TX Turnaround Time Register (TX_TA_TIME), this hardware does not use a pre-scaler, therefore the time specified is accurate. **Note:** A value of 200 (decimal) is recommended for this field.	R/W	00h

11.4.5 PD MAC BIST REGISTERS

The following sub-sections describe the various registers associated with the PD MAC Built-In Self Test (BIST).

11.4.5.1 BIST Control Register A (BIST_CTL_A)

Address: 1AC0h Size: 8 bits

Bits	Description	Type	Default
7:5	**RESERVED**	RO	-
4	**BIST_RX_EN** Enable BIST Receiver. Setting this bit will: • Enable the BIST receiver. • Reset the PRBS generator in the BIST RX logic so it is ready to compare incoming BIST PRBS pattern. • Reset the BIST error count registers (BIST Error Count High Register (BIST_ERR_CNT_HI) and BIST Error Count Low Register (BIST_ERR_CNT_LO). Normally this bit is set only when BIST receiver mode is required; however, this bit can be set to enable loopback BIST testing by disabling Automatic Response Mode and enabling the RX loopback (setting the EN_LOOPBACK bit in the RX Control Register A (RX_CTL_A)). This bit must be cleared and set by software every time a new BIST RX request comes in. **Note:** In BIST loopback test mode, BIST TX must be manually triggered for each TX packet since functional receiver path is disabled.	R/W	0b
3	**BIST_EN** Enable BIST. Set this bit when BIST logic is activated. This bit together with the EN_LOOPBACK bit in the RX Control Register A (RX_CTL_A) is used by hardware to block received data from going into functional path. This means that when loopback is enabled in BIST mode hardware cannot detect hard reset. This bit also powers up the BMC transmitter for BIST mode operation. Software should set this bit for a sufficient amount of time (5us) before trigger transmission of BIST frame via BIST_TX_START/BIST_TX_STATUS.	R/W	0b
2:0	**BIST_TX_MODE** Set BIST Test Mode. Only the following mode is supported: 010b: BIST Carrier Test Mode-2 (alternating '1's and '0's) All other modes are reserved and should not be used.	R/W	000b

11.4.5.2 BIST Control Register B (BIST_CTL_B)

Address: 1AC1h Size: 8 bits

Bits	Description	Type	Default
7:3	**RESERVED**	RO	-
2	**BIST_CLR_ERR_CNT** Clears the BIST error count registers BIST Error Count High Register (BIST_ERR_CNT_HI) and BIST Error Count Low Register (BIST_ERR_CNT_LO).	WO	0b
1	**BIST_TX_RST** Reset the TX logic and PRBS generator. Software must set this bit once before starting the BIST test, i.e., setting BIST_TX_START/BIST_TX_STATUS bit for the first time. Since this bit resets the PRBS generator it should not be set again until the current BIST test is ended. This bit is not latched and will always read '0' The PRBS generator cannot be reset by BIST_TX_START/BIST_TX_STATUS because that would reset it for each TX BIST frame. This bit should be used to stop the continuous BIST TX mode (2). It may take up to 1 bit time for the TX to stop. BIST_EN should not be cleared until after this time. **Note:** If this bit is set simultaneously with BIST_TX_START/BIST_TX_STATUS, it takes precedence and will void BIST_TX_START/BIST_TX_STATUS.	WO	0b
0	**BIST_TX_START/BIST_TX_STATUS** Start BIST transmission. Setting this bit will trigger transmission of BIST frame. This bit is not latched and reading it will return the status of current BIST transmission. Depending on BIST Mode, transmission will either continue indefinitely (unless BIST_TX_RST is asserted) or stop after sending a BIST PRBS frame. BIST does not support sending of next BIST frame automatically. It must be triggered by software by first clearing and then re-setting this bit. **Note:** This bit cannot be set simultaneously with BIST_TX_RST. BIST_TX_RST has precedence and will void BIST_TX_START/BIST_TX_STATUS. In BIST RX mode, hardware will automatically send the BIST Error Count message if the EN_AUTO_RSP_MODE bit in the TX Control Register A (TX_CTL_A) is set.	WO/RO	0b

11.4.5.3 BIST Error Count High Register (BIST_ERR_CNT_HI)

Address: 1AC2h Size: 8 bits

Bits	Description	Type	Default
7:0	**BIST_ERR_CNT_HI[7:0]** Upper byte of BIST error count (bits 15-8) (BIST_ERR_CNT[15:8])	RO	00h

11.4.5.4 BIST Error Count Low Register (BIST_ERR_CNT_LO)

Address: 1AC3h Size: 8 bits

Bits	Description	Type	Default
7:0	**BIST_ERR_CNT_LO[7:0]** Lower byte of BIST error count (bits 7-0) (BIST_ERR_CNT[7:0])	RO	00h

11.4.5.5 BIST RX Status Register (BIST_RX_STAT)

Address: 1AC7h Size: 8 bits

This register is used to help with functional testing. of PD AFE on ATE. Hardware is configured to transmit BIST PRBS frames. At the same time, loopback mode is enabled and BIST receive mode is enabled. Now when a PRBS frame is transmitted it is also received and compared. This register provides information when the BIST PRBS frame is done receiving and whether there were any errors detected.

Bits	Description	Type	Default
7:2	**RESERVED**	RO	-
1	**BIST_RX_ERROR** This bit is valid only when BIST_RX_DONE bit is set. A value of “0” indicates that BIST PRBS frame was received without errors. A value of “1” indicates that BIST PRBS frame was received with errors.	RO	0b
0	**BIST_RX_DONE** Successfully received a BIST RX FRAME. Writing a “1” to this bit will clear it.	R/W1C	0b

12.0 POWER SWITCH

To enable the device to efficiently support dead battery use cases, an integrated power switch is provided to select between two external +3.3V supplies:

- **3V3_ALW**: +3.3V main power supply input to integrated power switch.
- **3V3_VBUS**: +3.3V power supply input derived from VBUS to the integrated power switch.

The power switch allows the core to be powered from **3V3_ALW** normally, and from **3V3_VBUS** when **3V3_ALW** is not present. This effectively allows connection detection and system wakeup without external processor intervention (external processor in sleep mode).

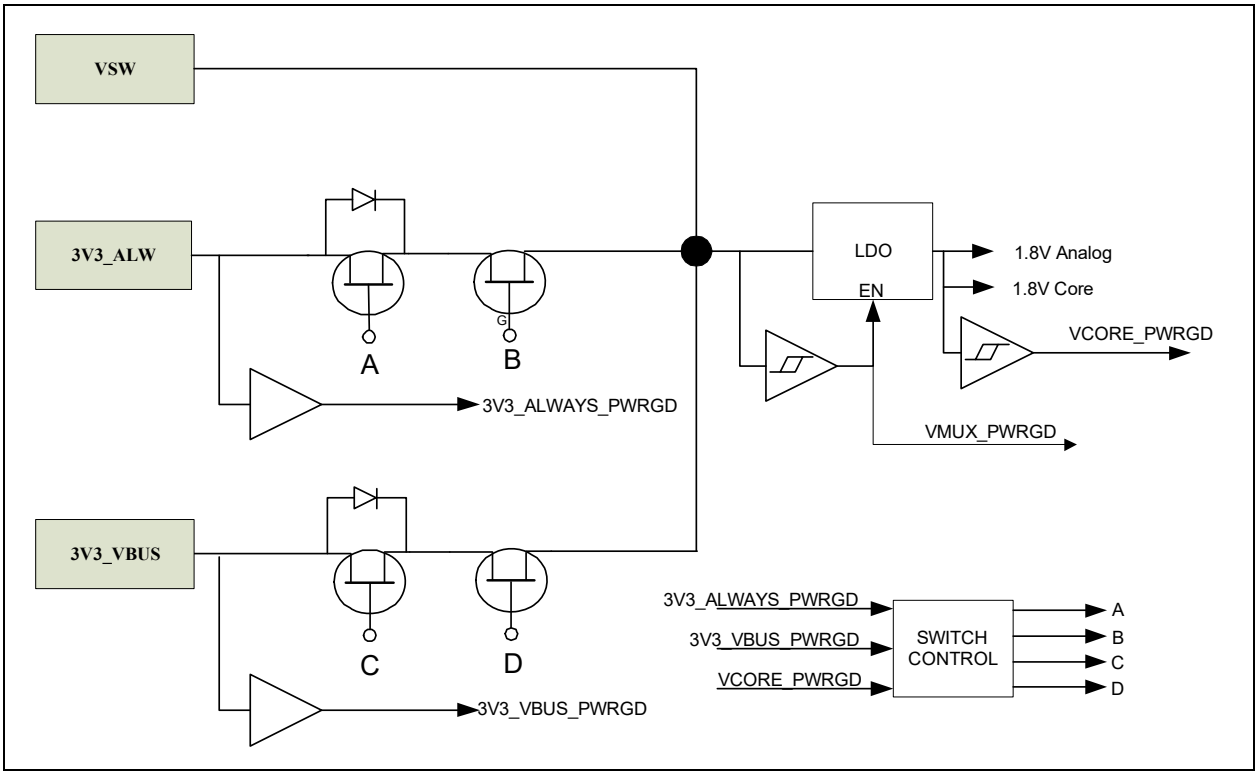
Attached to **3V3_ALW** and **3V3_VBUS** are two FET switches. The first FET switches have a diode across the output.

There are three voltage comparators. VBUS_PWRGD is on when **3V3_VBUS** exceeds 2.7V. 3V3_ALW_PWRGD is on when **3V3_ALW** exceeds 2.7V. VCORE_PWRGD is on when the core voltage reaches an operational level.

If VCORE_PWRGD is not asserted, the part is held in reset. If both VBUS_PWRGD and 3V3_ALW_PWRGD are not asserted, the part is held in reset, regardless of the state of the VCORE_PWRGD.

A block diagram of the internal power switch can be seen in Figure 12-1.

FIGURE 12-1: POWER SWITCH BLOCK DIAGRAM

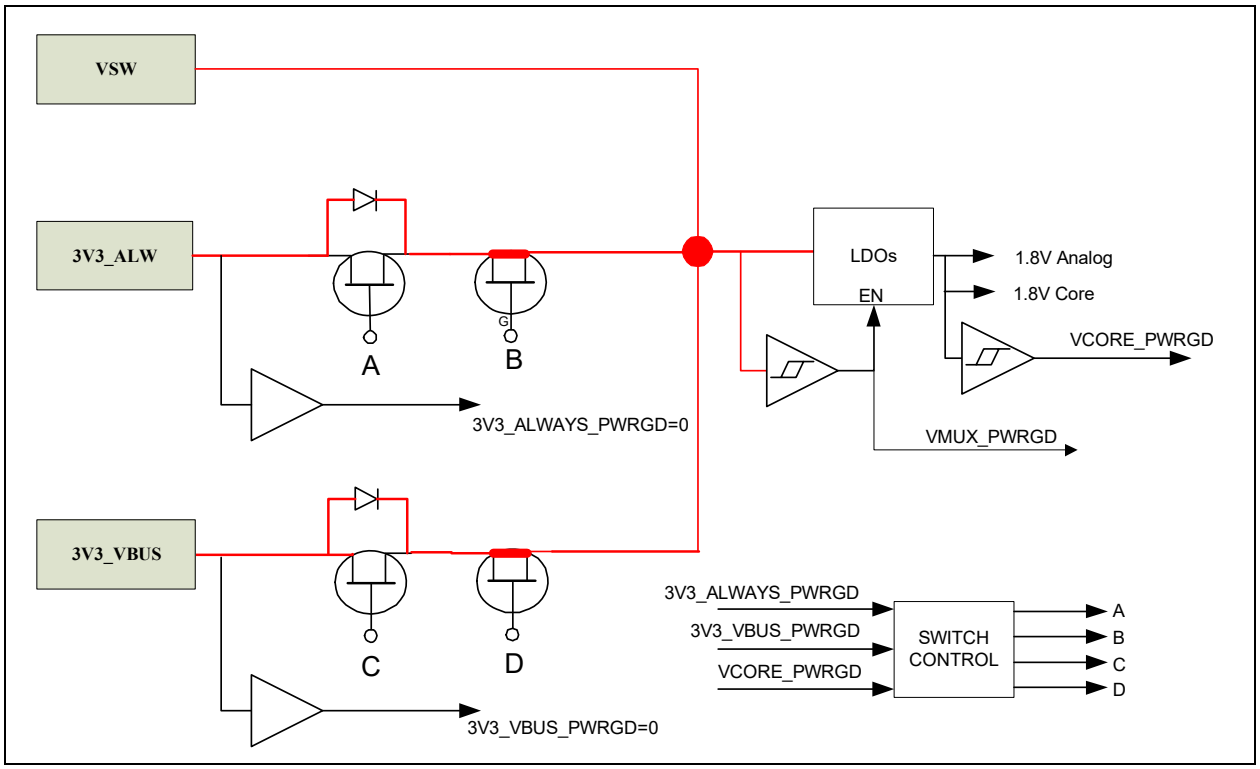


12.1 Operation

12.1.1 POWER UP

When the device is powering up, both VBUS_PWRGD and 3V3_ALW_PWRGD are low. Under these conditions, switches A and C are open, B and D are closed. The voltage going to the input of the core regulator is the higher of **3V3_ALW** and **3V3_VBUS**. The diodes prevent back drive.

FIGURE 12-2: POWER SWITCH: POWERING UP



12.1.2 3V3_ALW ONLY POWERS UP

In this case, only **3V3_ALW** powers up (VBUS_PWRGD = 0 and 3V3_ALW_PWRGD = 1). When this occurs, FET switch D is opened. After 50us, FET switch A is shorted. This sequence eliminates the possibility of back-drive when the diode is shorted to eliminate the voltage drop.

FIGURE 12-3: POWER SWITCH: POWERING UP STEP ONE

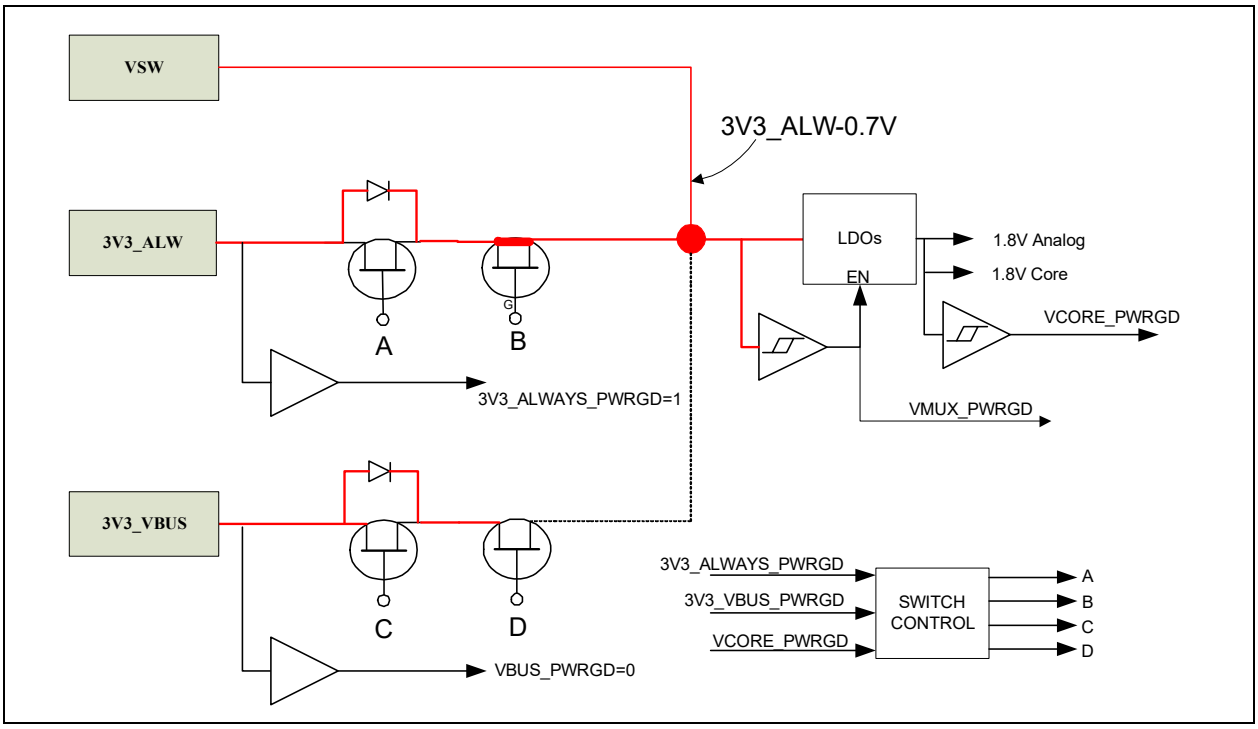
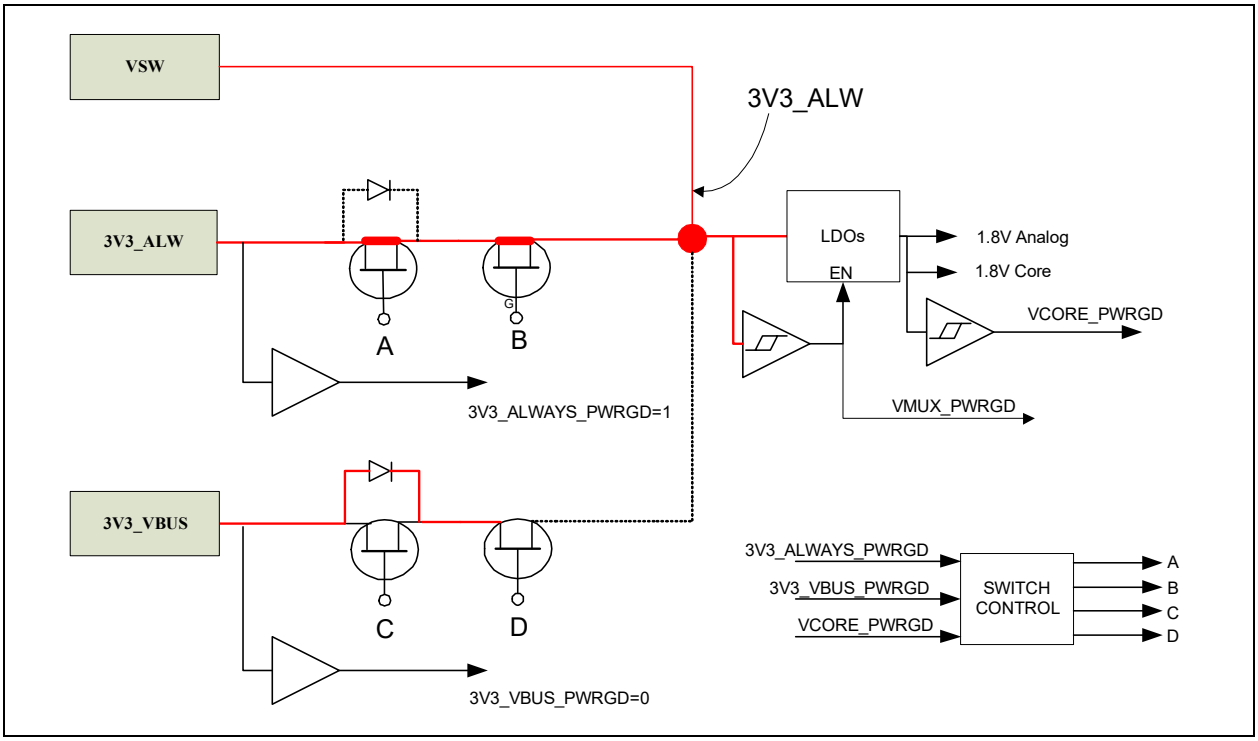


FIGURE 12-4: POWER SWITCH: POWERING UP STEP TWO



12.1.3 VBUS ONLY POWERS UP

This is the opposite example from the previous case. In this case, only VBUS powers up (3V3_ALW_PWRGD = 0 throughout, and VBUS_PWRGD = 1 when the threshold is hit). Once VBUS_PWRGD is valid, the internal regulator is turned on and normal operation begins. At no time during this process is the **3V3_ALW** rail back driven.When VBUS is available and **3V3_ALW** is not, VBUS can supply 500mA or more. be in one of two modes. The first is a normal supply mode that can supply 500mA or more. VBUS can also be in a high impedance mode where the output voltage drop linearly with current supplied (known as vSafeDB in the USB PD specification). The voltage and current relationship is show in the Figure 12-5.

FIGURE 12-5: POWER SWITCH: VBUS OUTPUT IN HIGH IMPEDANCE MODE

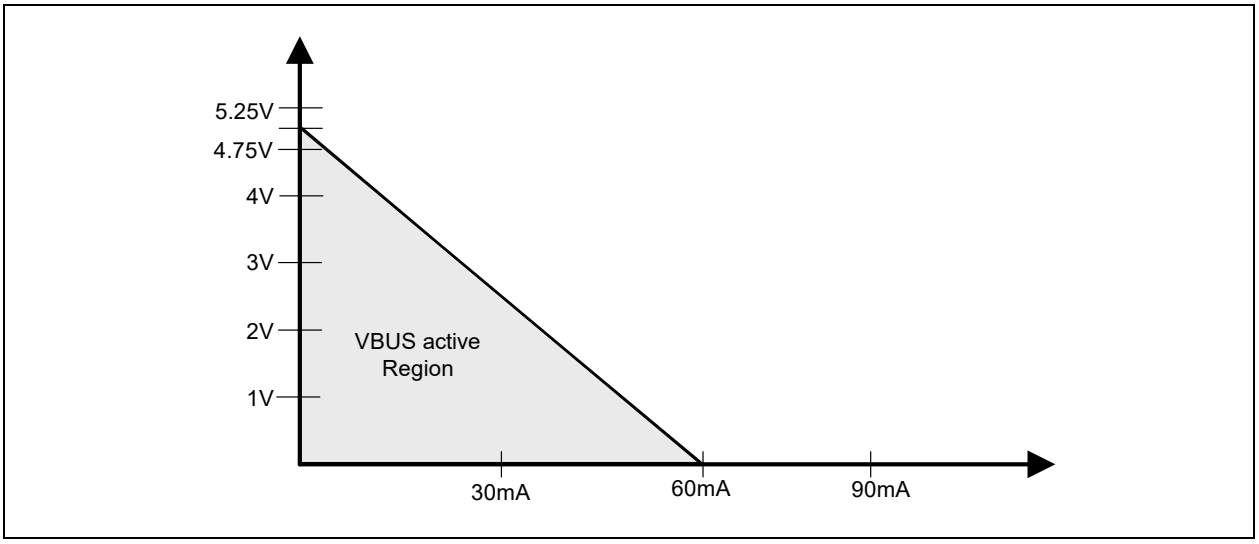


FIGURE 12-6: POWER SWITCH: VBUS ONLY POWER UP STEP ONE

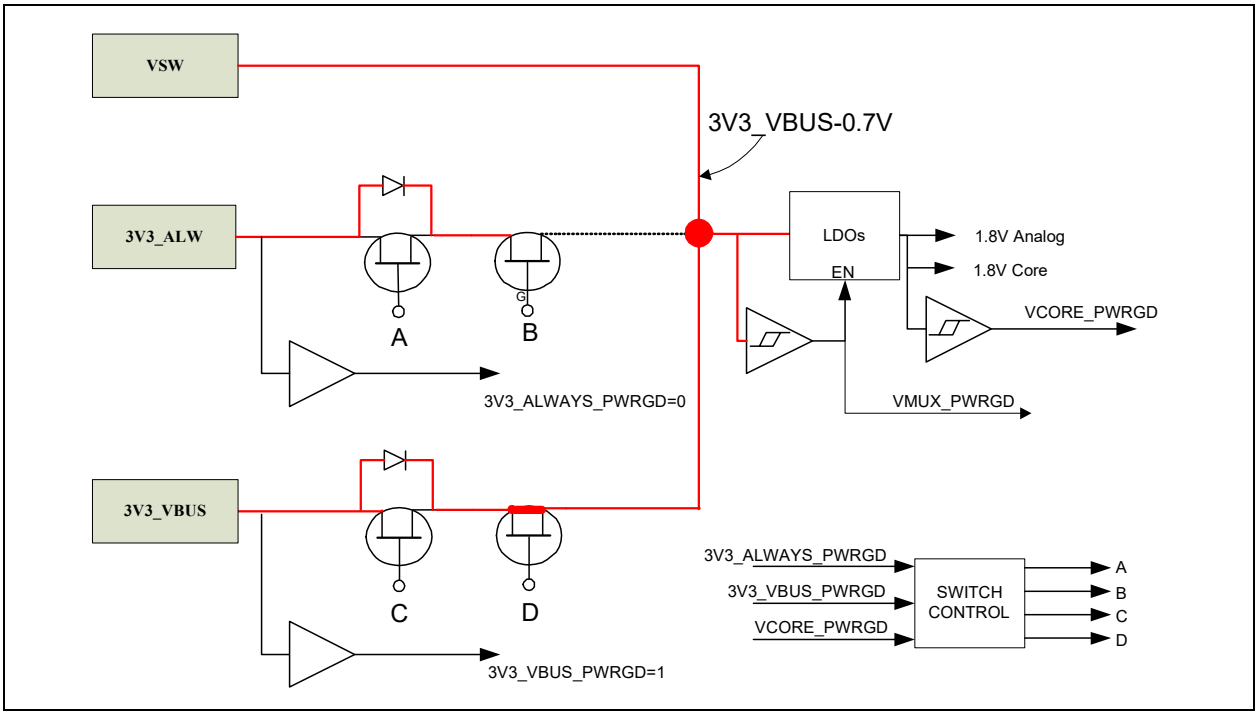
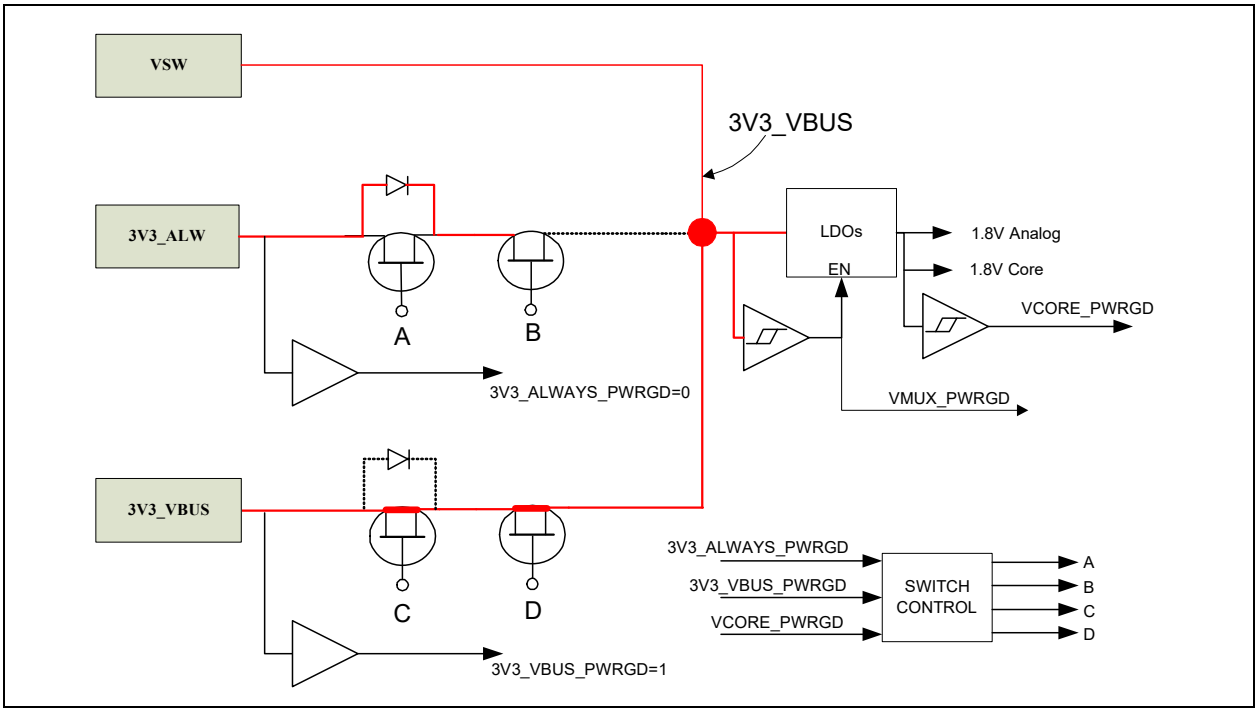


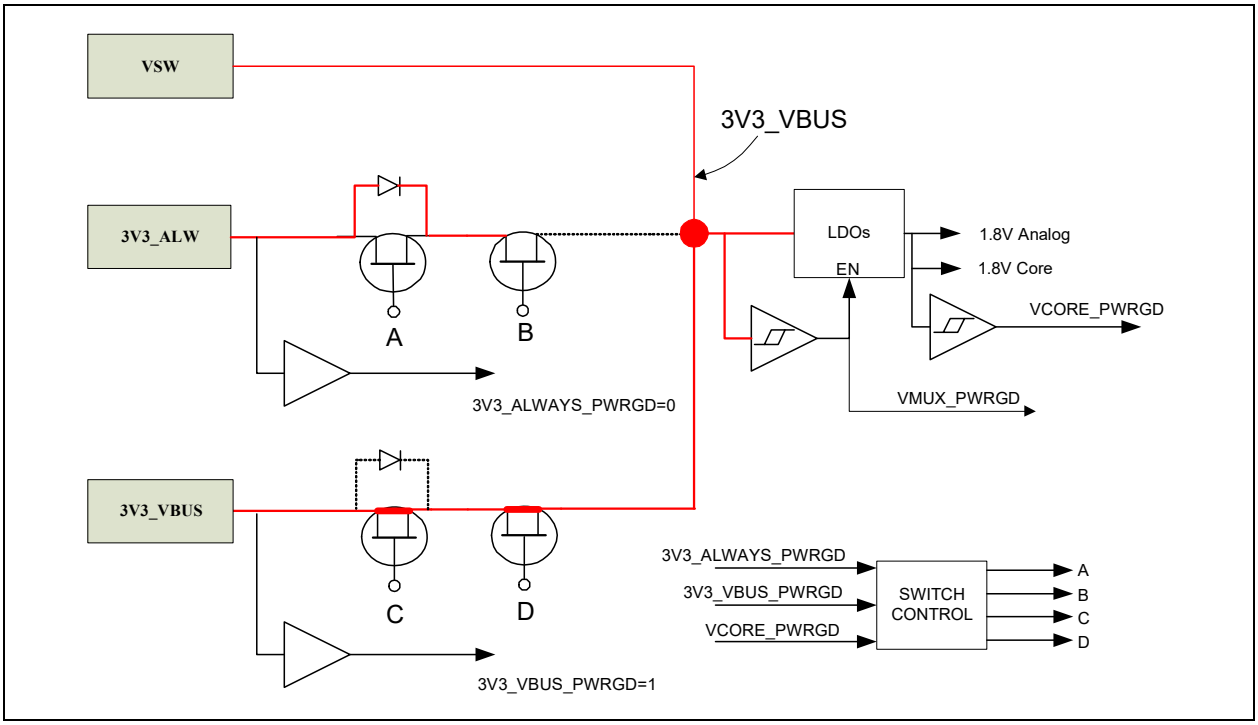
FIGURE 12-7: POWER SWITCH: VBUS ONLY POWER UP STEP TWO



12.1.4 VBUS POWERED UP FOLLOWED BY 3V3_ALW COMING UP

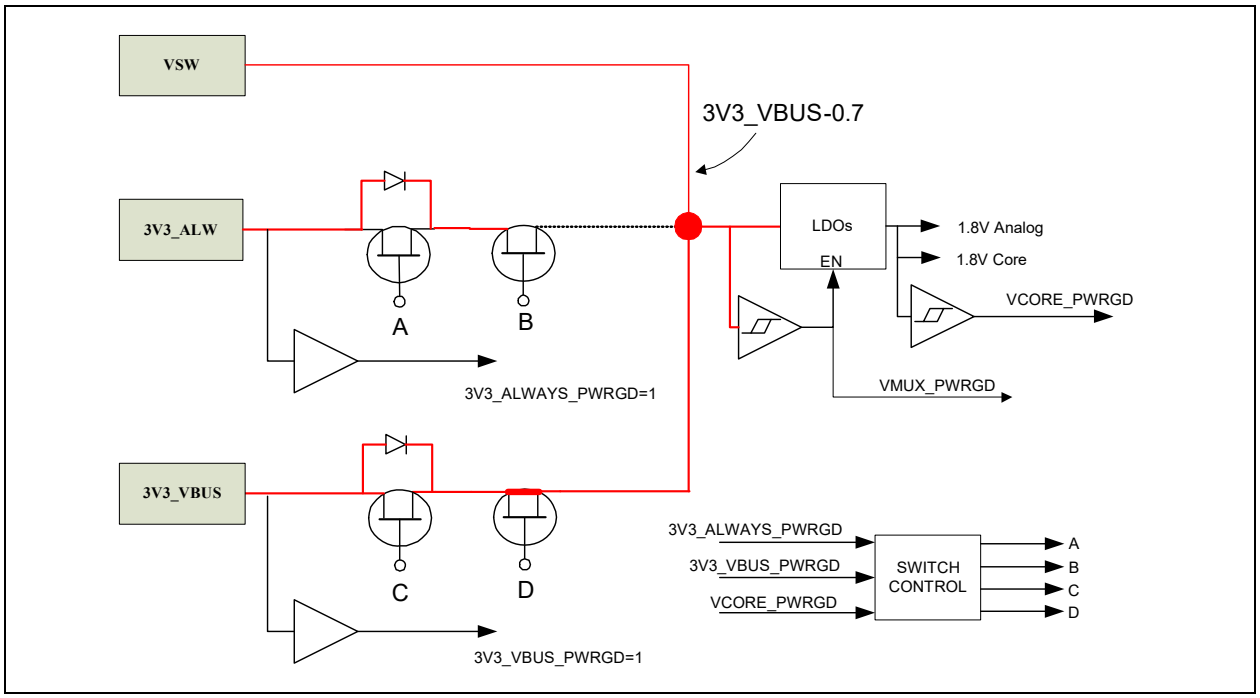
In this case, VBUS is already powered up and **3V3_ALW** starts to come up (and will take over and power the core). Figure 12-8 shows the VBUS only state.

FIGURE 12-8: POWER SWITCH: VBUS FOLLOWED BY 3V3_ALW STEP ONE



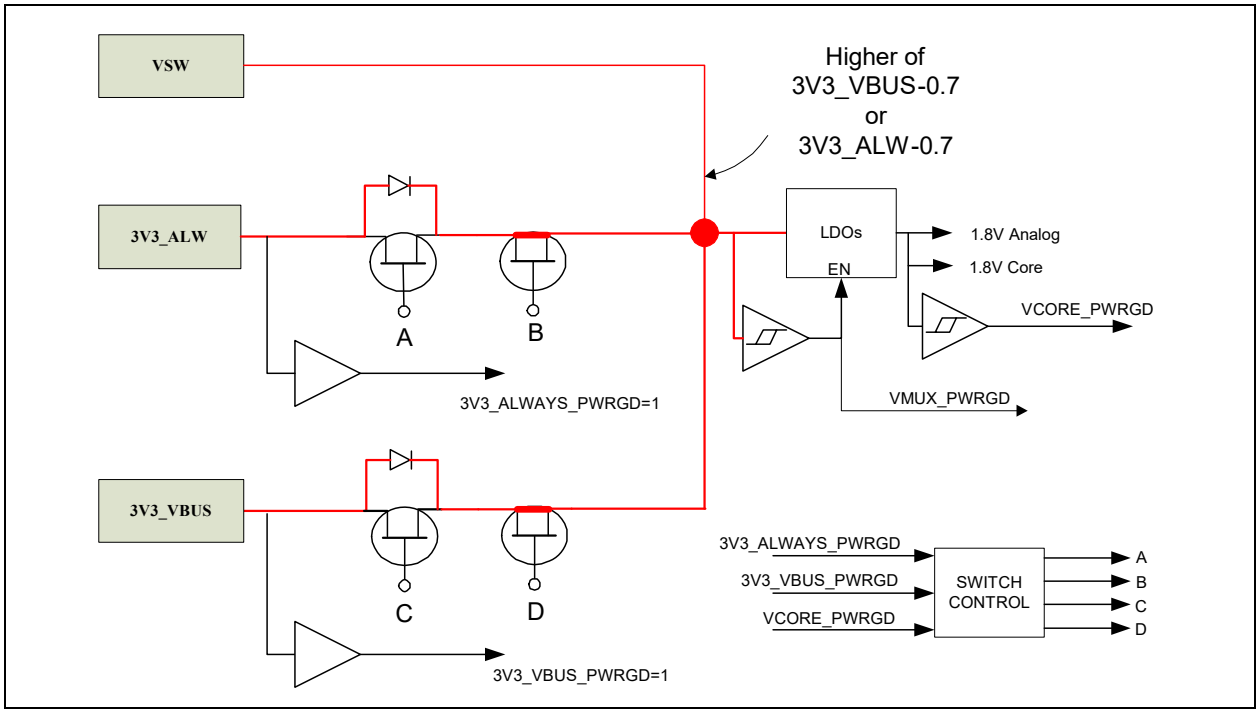
Once 3V3_ALW_PWRGD is detected, FET switch C is opened. The diode is required because **3V3_ALW** could be greater than or less than VBUS. The voltage at the input to the LDO drops to 3V3_VBUS-0.7V.

FIGURE 12-9: POWER SWITCH: VBUS FOLLOWED BY 3V3_ALW STEP TWO



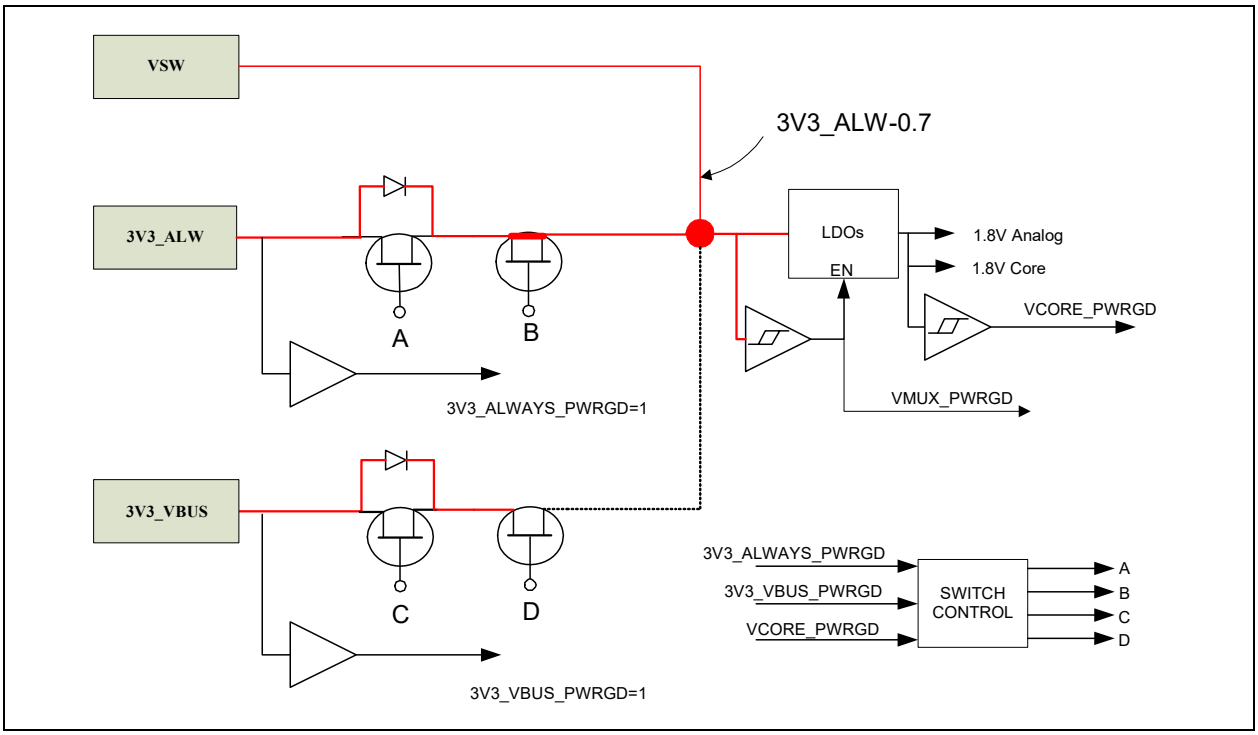
50us after FET C is opened, FET B is shorted. The two diodes ensure there is no back drive. The input to the core regulator will be at the higher voltage of VBUS - 0.7V or 3V3_ALW - 0.7V.

FIGURE 12-10: POWER SWITCH: VBUS FOLLOWED BY 3V3_ALW STEP THREE



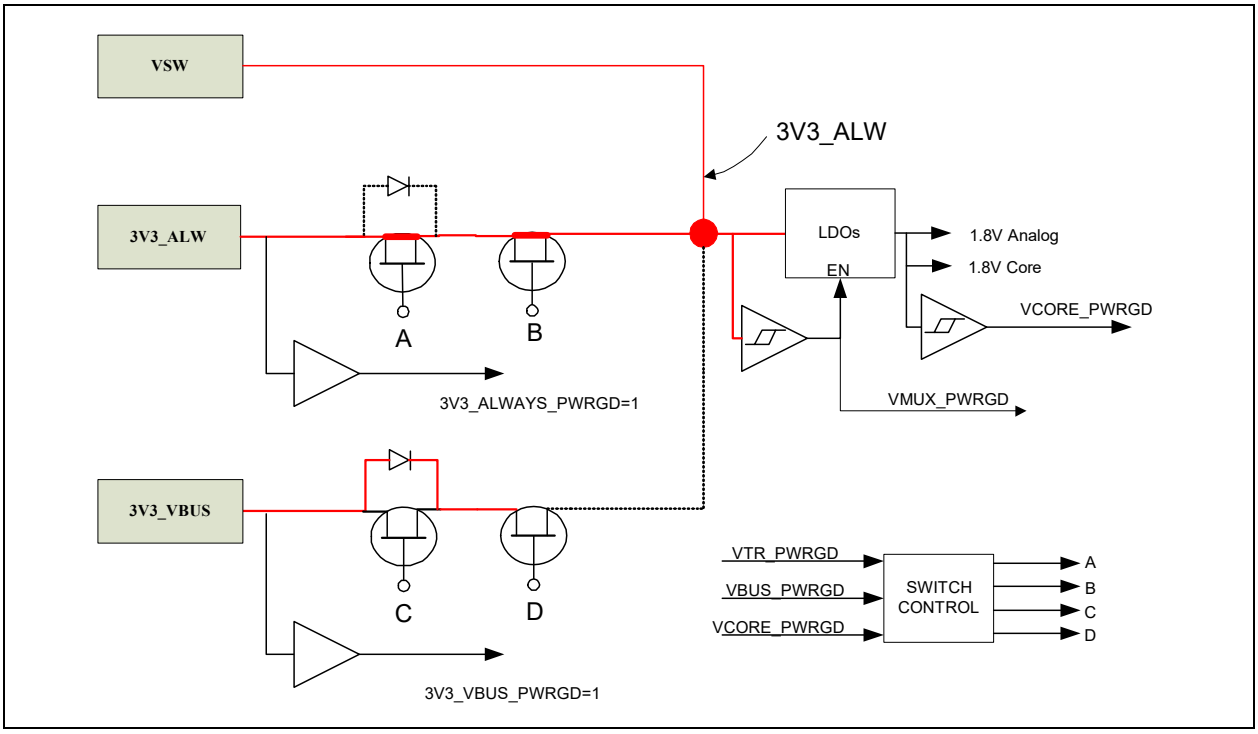
50us after FET B is shorted, FET D is opened. The core regulator input is now 3V3_ALW - 0.7V.

FIGURE 12-11: POWER SWITCH: VBUS FOLLOWED BY 3V3_ALW STEP FOUR



In the final step, 50us after FET D is opened, FET A is shorted. The input to the core regulator now becomes **3V3_ALW**.

FIGURE 12-12: POWER SWITCH: VBUS FOLLOWED BY 3V3_ALW STEP FIVE



12.1.5 3V3_ALW POWERED UP FOLLOWED BY VBUS COMING UP

In this case, **3V3_ALW** is already powered up and VBUS starts to come up. **3V3_ALW** will remain the power source for the core and no changes will be made to the FET switches.

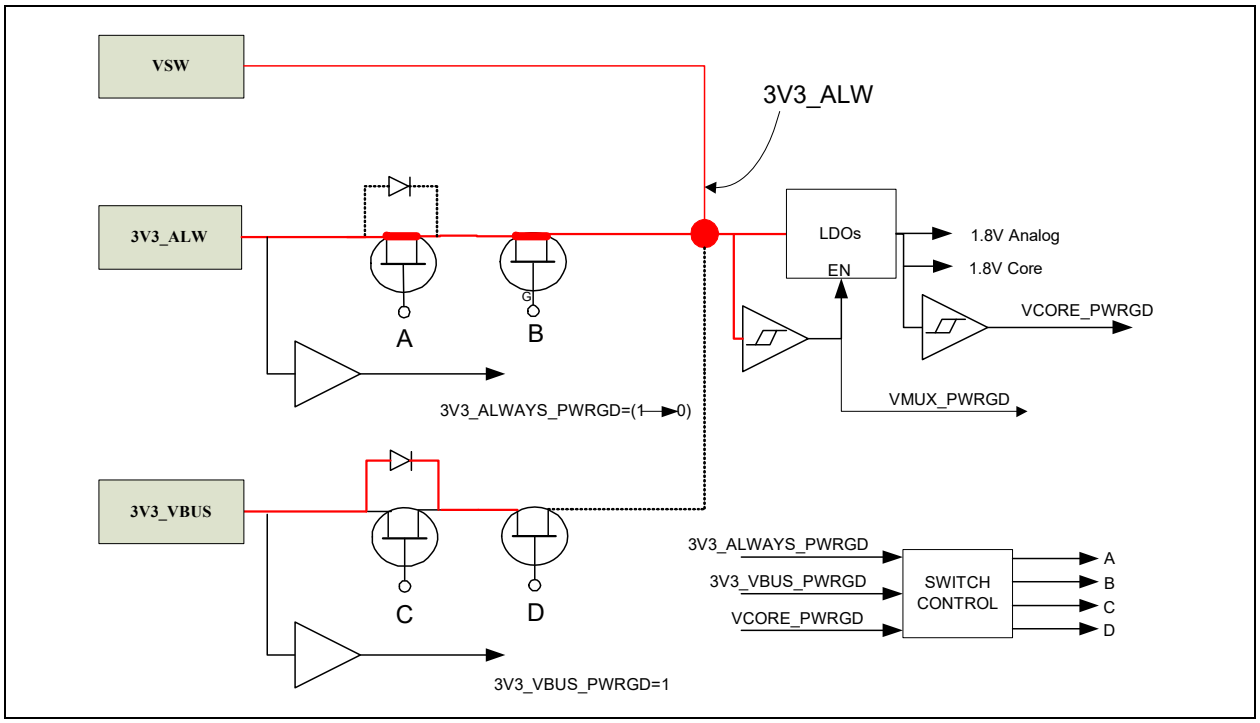
12.1.6 POWERED UP WITH VBUS AND 3V3_ALW, VBUS GOING AWAY

In this case, **3V3_ALW** and VBUS are both present. The core is always powered from **3V3_ALW**. If VBUS goes away, nothing changes in the voltage to the core. An interrupt will be generated to the processor to let it know that VBUS status changed.

12.1.7 POWERED UP WITH VBUS AND 3V3_ALW, 3V3_ALW GOING AWAY

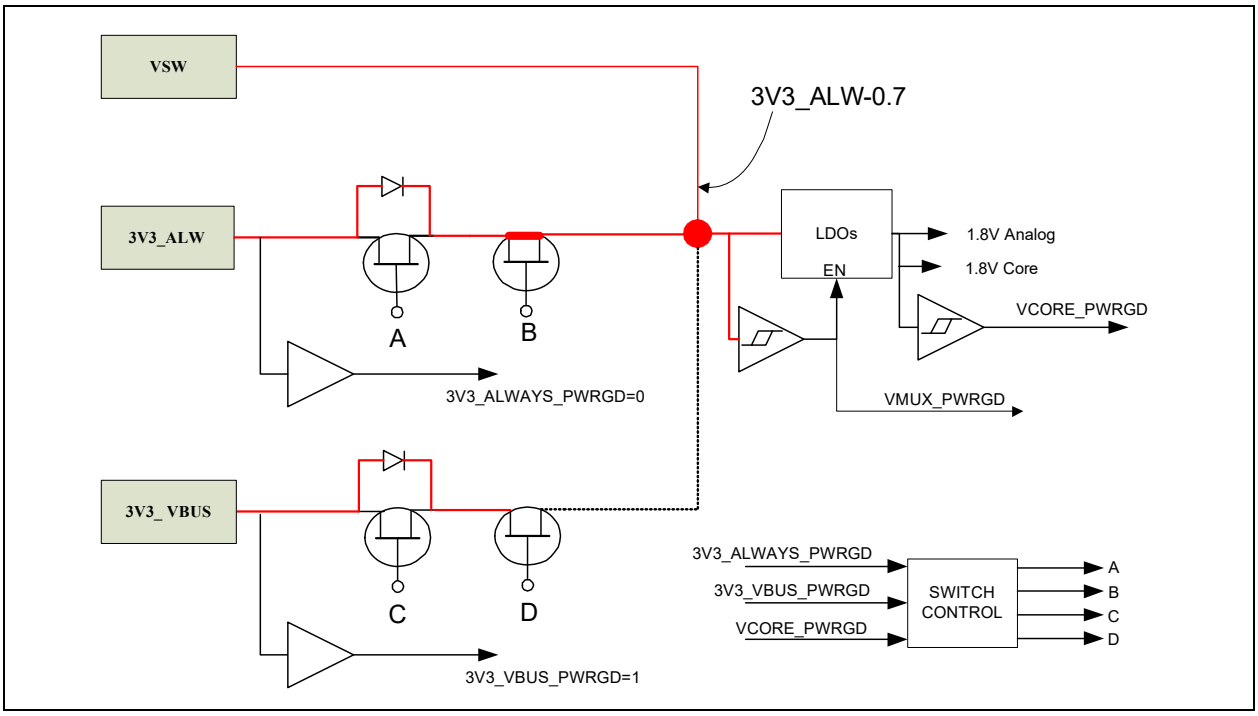
In this case, **3V3_ALW** and VBUS are both present. The core is always powered from **3V3_ALW**. If **3V3_ALW** goes away, the switch will first transition to the initial power on state, then transition to the VBUS only state.

FIGURE 12-13: POWER SWITCH: VBUS PRESENT, 3V3_ALW GOING AWAY STEP ONE



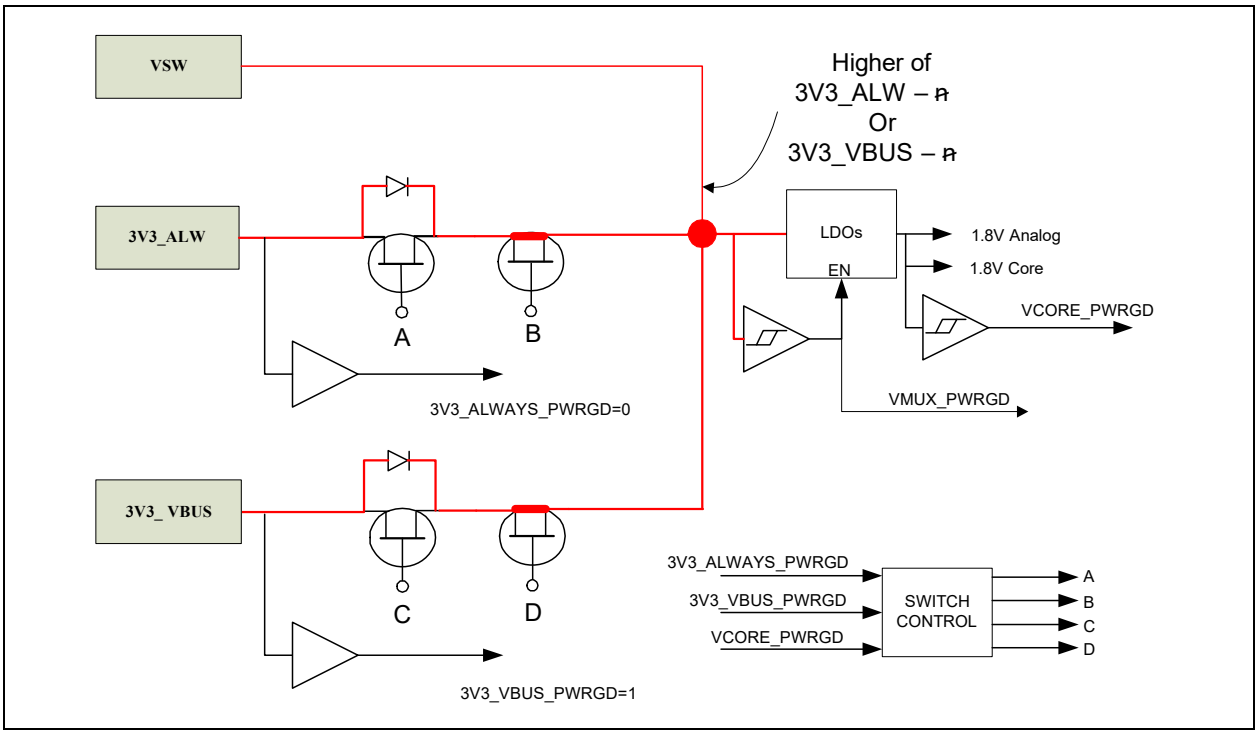
First FET A is opened. The voltage at the core regulator input goes to 3V3_ALW-0.7

FIGURE 12-14: POWER SWITCH: VBUS PRESENT, 3V3_ALW GOING AWAY STEP TWO



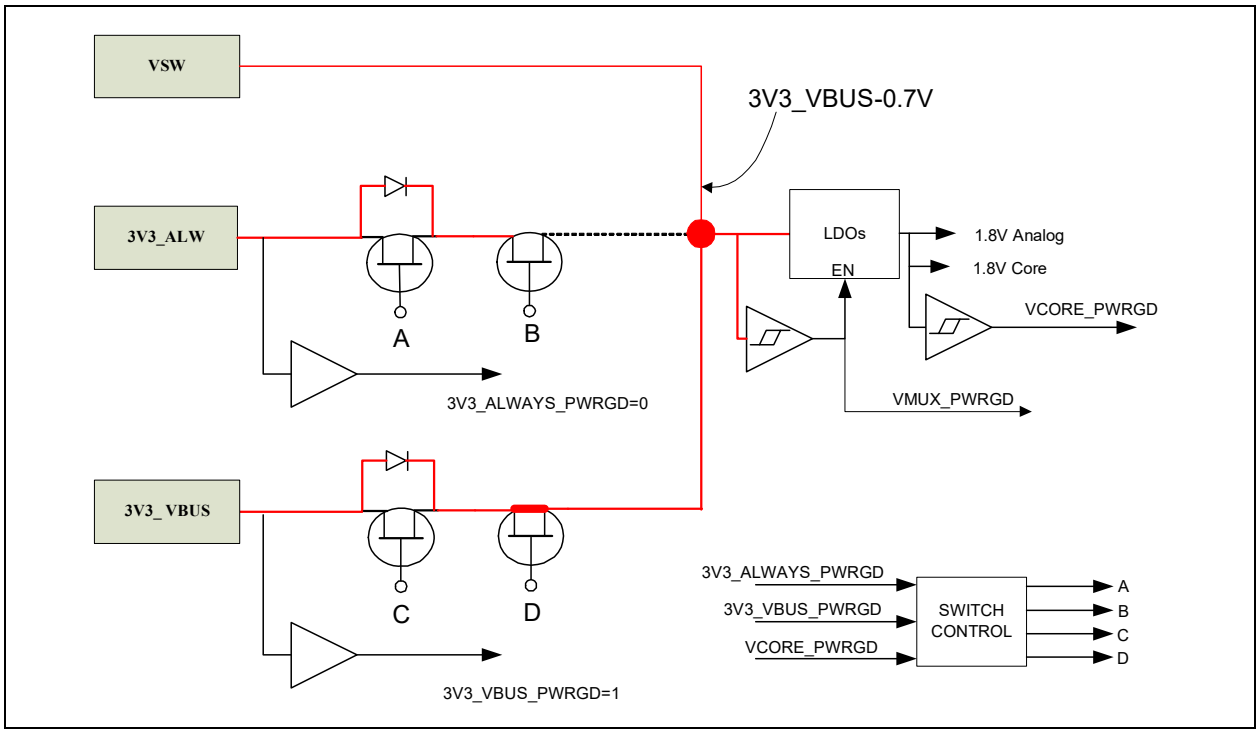
50uS later, the FET is closed. The voltage at the input of the core regulator will be the higher of 3V3_ALW-0.7 or 3V3_V-BUS -0.7. Core regulator may fall out of regulation.

FIGURE 12-15: POWER SWITCH: VBUS PRESENT, 3V3_ALW GOING AWAY STEP THREE



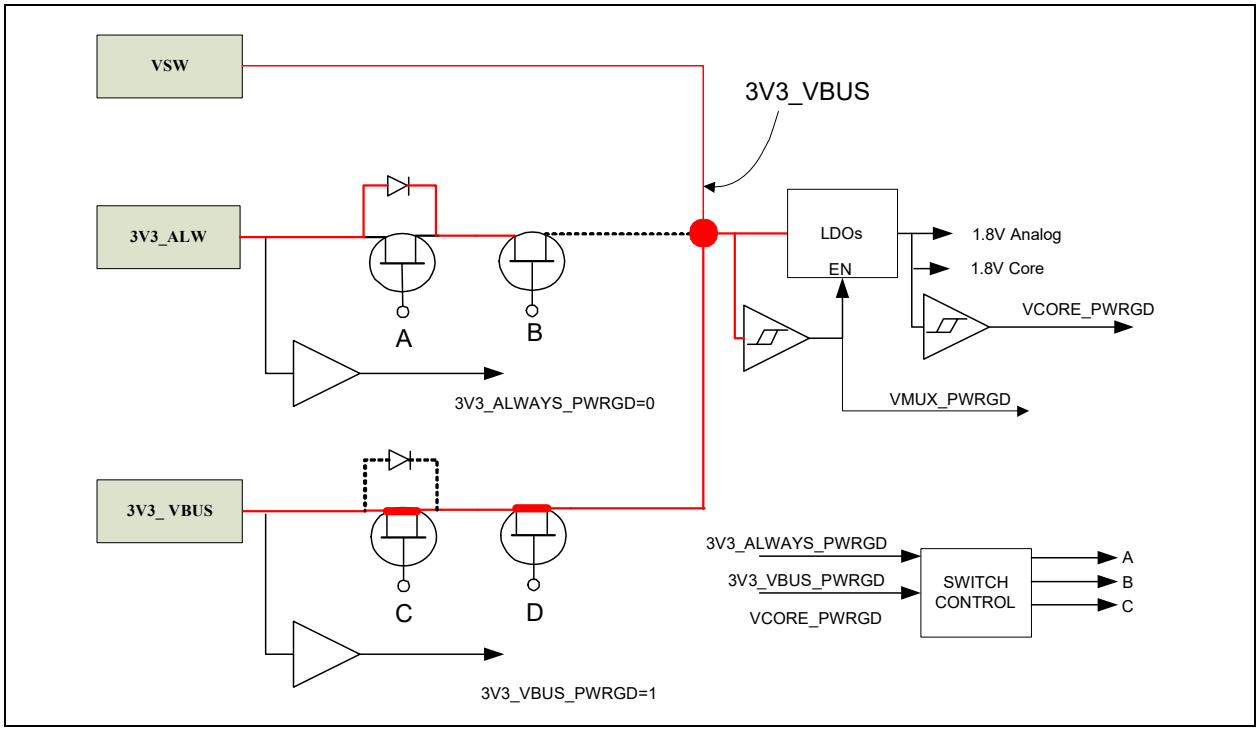
50uS later, FET B is opened. The voltage at the input of the core regulator is VBUS-0.7.

FIGURE 12-16: POWER SWITCH: VBUS PRESENT, 3V3_ALW GOING AWAY STEP FOUR



50uS later, FET C is shorted. The voltage at the input of the core regulator is 3V3_VBUS.

FIGURE 12-17: POWER SWITCH: VBUS PRESENT, 3V3_ALW GOING AWAY STEP FIVE



12.2 Software Override

In the event that both **3V3_VBUS** and **3V3_ALW** are available, the Power Switch automatically selects **3V3_ALW** for operation. This can be overridden by software the VBUS Switch Enable Override (VBUS_SW_EN_OVR) bit of the Power Switch Control Register (PWR_SW_CTL), which forces the switch to operate off of VBUS. When this bit is set, the auto-switch mechanism of the switch is disabled.

12.3 Power Switch Interrupts

The power switch interrupt alerts software to changes in the state of the Power Switch. These events are listed in the Power Switch Interrupt Status Register (PWR_INT_STS). The interrupt persists until the asserted bits in the Power Switch Interrupt Status Register (PWR_INT_STS) are cleared. Individual interrupt events can be a enabled via the Power Switch Interrupt Enable Register (PWR_INT_EN). Power switch interrupt status sources are capable of triggering asynchronous wakes.

12.4 Power Switch Registers

This section details the power switch registers. For an overview of the entire device register map, refer to Section 4.0, "Register Map," on page 18.

TABLE 12-1: POWER SWITCH REGISTER MAP

Address	Register Name (Symbol)
1C00h	Power Switch Control Register (PWR_SW_CTL)
1C01h	Power Switch Status Register (PWR_SW_STS)
1C02h	Power Switch Interrupt Status Register (PWR_INT_STS)
1C03h	Power Switch Interrupt Enable Register (PWR_INT_EN)
1C04h	Power Switch State Status Register (PWR_SW_STATE_STS)
1C05h – 1FFFh	Reserved for future expansion

12.4.1 POWER SWITCH CONTROL REGISTER (PWR_SW_CTL)

Address: 1C00h Size: 8 bits

Bits	Description	Type	Default
7:2	**RESERVED**	RO	-
1	**Power Switch Enable (PM_SW_ENB)** This active low signal enables the power switch.	R/W	0b
0	**VBUS Switch Enable Override (VBUS_SW_EN_OVR)** When this bit is asserted, the power switch utilizes the **3V3_VBUS** supply. This bit only has meaning if the device is currently operating on **3V3_ALW** and the VBUS supply is present. This bit disables the auto-switch mechanism of the power switch.	R/W	0b

12.4.2 POWER SWITCH STATUS REGISTER (PWR_SW_STS)

Address: 1C01h Size: 8 bits

Bits	Description	Type	Default
7:6	**RESERVED**	RO	-
5	**3V3_ALW POR (3V3_ALW_RDY)** POR output from the main VBAT external supply. Active high when the supply is above 2.85V. Active low when the supply drops below 2.7V.	RO	0b
4	**3V3_VBUS POR (VBUS_RDY)** POR output from the VBUS external supply. Active high when the supply is above 4V. Active low when the supply drops below 2.7V.	RO	0b
3	**RESERVED**	RO	-
2	**PWR_RDY** Indicates power ready on VSW and VDD18 supplies.	RO	1b
1	**3V3_ALW Switch Gate Off (3V3_ALW_SW_OKB)** Indicates state of VSW with respect to 3V3_ALW. 0: VSW < 3V3_ALW 1: VSW > 3V3_ALW	RO	0b
0	**VBUS Switch Gate Off (VBUS_SW_OKB)** Indicates state of VSW with respect to VBUS. 0: VSW < VBUS 1: VSW > VBUS	RO	0b

12.4.3 POWER SWITCH INTERRUPT STATUS REGISTER (PWR_INT_STS)

Address: 1C02h Size: 8 bits

Bits	Description	Type	Default
7:6	**RESERVED**	RO	-
5	**3V3_ALW to VBUS Switch Interrupt (3V3_ALW2VBUS_INTR)** This interrupt asserts after the power switch automatically transitions from selecting **3V3_ALW** to selecting VBUS as the power source. Write a '1' to clear this bit. Writes of '0' have no effect. **Note:** Assertion of the VBUS Switch Enable Override (VBUS_SW_EN_OVR) bit causes this interrupt to assert.	R/WC	0b
4	**VBUS to 3V3_ALW Switch Interrupt POR (VBUS23V3_ALW_INTR)** This interrupt asserts after the power switch automatically transitions from selecting VBUS to selecting **3V3_ALW** as the power source. Write a '1' to clear this bit. Writes of '0' have no effect. **Note:** Assertion of the VBUS Switch Enable Override (VBUS_SW_EN_OVR) bit prevents assertion of this interrupt when VBUS is lost and **3V3_ALW** is present.	R/WC	0b
3	**3V3_ALW Lost Interrupt (3V3_ALW_LOST_INTR)** This interrupt asserts when 3V3_ALW POR (3V3_ALW_RDY) transitions from 1 to 0. It indicates that the **3V3_ALW** supply has been lost. Write a '1' to clear this bit. Writes of '0' have no effect.	R/WC	0b
2	**3V3_ALW Ready Interrupt (3V3_ALW_RDY_INTR)** This interrupt asserts when 3V3_ALW POR (3V3_ALW_RDY) transitions from 0 to 1. It indicates that a **3V3_ALW** supply has been detected. Write a '1' to clear this bit. Writes of '0' have no effect.	R/WC	0b
1	**VBUS Lost Interrupt (VBUS_LOST_INTR)** This interrupt asserts when 3V3_VBUS POR (VBUS_RDY) transitions from 1to 0. It indicates that the VBUS supply has been lost. Write a '1' to clear this bit. Writes of '0' have no effect.	R/WC	0b
0	**VBUS Ready Interrupt (VBUS_RDY_INTR)** This interrupt asserts when 3V3_VBUS POR (VBUS_RDY) transitions from 0 to 1. It indicates that a VBUS supply has been detected. Write a '1' to clear this bit. Writes of '0' have no effect.	R/WC	0b

12.4.4 POWER SWITCH INTERRUPT ENABLE REGISTER (PWR_INT_EN)

Address: 1C03h Size: 8 bits

Bits	Description	Type	Default
7:6	**RESERVED**	RO	-
5	**3V3_ALW2VBUS_EN** When '0', prevents the generation of this interrupt.	R/W	0b
4	**VBUS23V3_ALW_EN** When '0', prevents the generation of this interrupt.	R/W	0b
3	**3V3_ALW_LOST_EN** When '0', prevents the generation of this interrupt.	R/W	0b
2	**3V3_ALW_RDY_EN** When '0', prevents the generation of this interrupt.	R/W	0b
1	**VBUS_LOST_EN** When '0', prevents the generation of this interrupt.	R/W	0b
0	**VBUS_RDY_EN** When '0', prevents the generation of this interrupt.	R/W	0b

12.4.5 POWER SWITCH STATE STATUS REGISTER (PWR_SW_STATE_STS)

This register indicates the current state of the **3V3_ALW** and **3V3_VBUS** switches.

Address: 1C04h Size: 8 bits

Bits	Description	Type	Default
7:2	**RESERVED**	RO	-
1	**VBUS Switch State (3V3_VBUS_STATE)** 0 = **3V3_VBUS** switch is disabled. 1 = **3V3_VBUS** switch is enabled.	RO	0b
0	**3V3_ALW Switch State (3V3_ALW_STATE)** 0 = **3V3_ALW** switch is disabled. 1 = **3V3_ALW** switch is enabled.	RO	0b

13.0 HDMI/DISPLAYPORT HOT PLUG DETECT (HPD)

13.1 Overview

The device provides hardware offload support for detecting the state of a Display Port compliant **HPD** input/output pin to the device. This pin may also be used to implement HDMI alternate mode.

The USB Type-C® DisplayPort Alternate Mode Specification defines the **HPD** state in terms of two status flags:

- **HPD_STATE**: Indicates whether the HPD's logical state is high or low (denoted as HPD_HIGH or HPD_LOW, respectively). For the purposes of communicating the HPD state over USB, the logical state of HPD is considered as remaining high while receiving an IRQ_HPD, and is low during the time that HPD is being de-bounced on a new mechanical connection. The logical state of HPD transitions from high to low when a low level on the HPD link has been detected for 2ms (i.e., longer than the maximum IRQ_HPD pulse detection time). The logical state of HPD is unchanged during glitches (as specified in DP v1.3) on the HPD link.
- **IRQ_HPD**: Indicates an IRQ_HPD (i.e., a high-to-low transition on HPD followed by a low-to-high transition was detected between 250us and 2ms later, as specified in DP v1.3).

Note:	The **HPD** pin is configured as an input in DisplayPort Source applications, and configured as a push/pull driver in DisplayPort Sink applications.

Note:	HDMI alternate mode requires a 5V level shifter to generate the HPD output. Since the **HPD** pin is 5V tolerant, it can safely be connected directly to HPD when operating as a receiver.

13.2 HPD Receiver

The device provides HPD detection (input) via the **HPD** pin. HPD detection is enabled by setting HPD Enable bit and configuring the **HPD** pin to be an input via HPD Configuration in HPD Control Register (HPD_CTL). Upon setting this bit, the device continuously monitors the state of the **HPD** pin as follows:

- When an IRQ_HPD event occurs, the IRQ_HPD interrupt bit in HPD Interrupt Status Register (HPD_INT_STS) will assert. IRQ_HPD is detected when **HPD** is initially high and de-asserts for a time greater than or equal to the value in the IRQ_HPD Minimum Time Register (IRQ_HPD_MIN_TIME) and less than or equal to the value in the IRQ_HPD Maximum Time Register (IRQ_HPD_MAX_TIME).
- When a Low to High transition is debounced and detected, the HPD_HIGH interrupt bit in HPD Interrupt Status Register (HPD_INT_STS) will assert. HPD_HIGH is detected when the **HPD** pin is initially low and asserts for a time greater than or equal to the value in the HPD High Detect Time Register (HPD_HIGH_DET_TIME).
- When a High to Low transition is debounced and detected, the HPD_LOW interrupt bit in HPD Interrupt Status Register (HPD_INT_STS) will assert. HPD_LOW is detected when **HPD** pin is initially low and asserts for a time greater than or equal to the value in the HPD Low Detect Time Register (HPD_LOW_DET_TIME).

APPLICATION NOTE: The HPD Interrupt Status Register (HPD_INT_STS) contents are reset to 00h when HPD Enable is cleared.

APPLICATION NOTE: Enabled and asserted interrupts trigger the HPD_INT interrupt in the Interrupt Status Register (INT_STS).

A mechanism has been added via the HPD Event Queue Register (HPD_QUEUE) to record up to four **HPD** events. This function is enabled when HPD Enable is set and HPD Configuration indicates the **HPD** pin is operating as an input. See Section 13.4.4, "HPD Event Queue Register (HPD_QUEUE)," on page 191 for additional details.

APPLICATION NOTE: The state of the **HPD** pin is available by reading HPD State in HPD Control Register (HPD_CTL).

This feature is used as follows:

1. Software configures IRQ_HPD Minimum Time Register (IRQ_HPD_MIN_TIME), IRQ_HPD Maximum Time Register (IRQ_HPD_MAX_TIME), HPD High Detect Time Register (HPD_HIGH_DET_TIME) and HPD Low Detect Time Register (HPD_LOW_DET_TIME). To comply with the DisplayPort specification for IRQ_HPD, the IRQ_HPD Minimum Time Register (IRQ_HPD_MIN_TIME) and IRQ_HPD Maximum Time Register (IRQ_HPD_-

MAX_TIME) should be configured to 250 us and 2 ms, respectively. Likewise, HPD Low Detect Time Register (HPD_LOW_DET_TIME) should be configured to greater than 2 ms.

2. If desired, interrupts may be enabled by asserting the respective bits in the HPD Interrupt Enable Register (HPD_INT_EN).
3. HPD Enable is set and HPD Configuration is cleared in the HPD Control Register (HPD_CTL).
4. When an Hot Plug event occurs, the HPD Event Queue Register (HPD_QUEUE) is updated and an interrupt will assert if configured.
5. HPD state changes are reflected by assertion of QUEUE_NOT_EMPTY, HPD_HIGH, HPD_LOW and IRQ_HPD bits in HPD Interrupt Status Register (HPD_INT_STS). They may cause assertion of HPD_INT in the Interrupt Status Register (INT_STS), if enabled.
6. Software reads the HPD Event Queue Register (HPD_QUEUE) to determine the Hot Plug events received.
7. Software writes to the HPD Event Queue Register (HPD_QUEUE) and clears the valid entries.

13.3 HPD Transmission

The device provides HPD transmission (output) via the **HPD** pin. This is done via the HPD Control Register (HPD_CTL) by setting both HPD Configuration and HPD Enable to “1”. The value driven on the HPD pin is controlled by HPD Output Value.

Support is also provided for generating an IRQ_HPD. This is accomplished by programming the deassertion time in HPD IRQ Generation Time Register (HPD_IRQ_GEN) and then setting Generate IRQ in HPD Control Register (HPD_CTL). Generate IRQ self clears after the interrupt pulse is issued.

This feature is used as follows:

1. Configure the **HPD** pin as an output by setting HPD Configuration and HPD Enable in HPD Control Register (HPD_CTL). The value for **HPD** is defined by the value programmed in HPD Output Value.
2. If it is desired to generate an IRQ_HPD, the HPD IRQ Generation Time Register (HPD_IRQ_GEN) must be programmed with the desired HPD deassertion time.
3. If it is desired to receive an interrupt upon generation of IRQ_HPD, the respective bit in the HPD Interrupt Enable Register (HPD_INT_EN) should be enabled.
4. Software requests IRQ_HPD generation by setting Generate IRQ in the HPD Control Register (HPD_CTL). HPD Output Value must also be set to 1b. or the request will have no meeting.
5. Upon generating the IRQ_HPD interrupt, the Generate IRQ bit will self clear.
6. The IRQ_HPD bit asserts in the HPD Interrupt Status Register (HPD_INT_STS) and shall cause assertion of the HPD_INT interrupt in the Interrupt Status Register (INT_STS), if enabled.

13.4 DisplayPort HPD Registers

This section details the DisplayPort HPD registers. For an overview of the entire device register map, refer to Section 4.0, "Register Map," on page 18.

TABLE 13-1: DISPLAYPORT HPD REGISTER MAP

Address	Register Name (Symbol)
0C00h	HPD Control Register (HPD_CTL)
0C01h	HPD Interrupt Status Register (HPD_INT_STS)
0C02h	HPD Interrupt Enable Register (HPD_INT_EN)
0C03h	HPD Event Queue Register (HPD_QUEUE)
0C04h	IRQ_HPD Minimum Time Register (IRQ_HPD_MIN_TIME)
0C05h	IRQ_HPD Maximum Time Register (IRQ_HPD_MAX_TIME)
0C06h	HPD High Detect Time Register (HPD_HIGH_DET_TIME)
0C08h	HPD Low Detect Time Register (HPD_LOW_DET_TIME)
0C0Ah	HPD IRQ Generation Time Register (HPD_IRQ_GEN)
0C0Bh – 0FFFh	Reserved for future expansion

Note:	**RESERVED address space must not be written under any circumstances. Failure to heed this warning may result in untoward operation and unexpected results.**

13.4.1 HPD CONTROL REGISTER (HPD_CTL)

Address: 0C00h Size: 8 bits

Bits	Description	Type	Default
7	**HPD State** Provides the state of the **HPD** pin. This bit resets to 0b when HPD Enable is cleared. **Note:** An IRQ_HPD event is recorded as an HPD_HIGH state. **Note:** This bit is valid when HP is configured as an input or output.	RO	-
6:4	**RESERVED**	RO	-
3	**HPD Output Value** This bit only has meaning when the **HPD** pin is configured as an output per HPD Configuration. Otherwise, it will always read 0b.	R/W	0b
2	**Generate IRQ** When set, the **HPD** pin generates an HPD IRQ by deasserting **HPD** for the amount of time defined in the HPD IRQ Generation Time Register (HPD_IRQ_GEN). This bit only has meaning when the **HPD** pin is configured as an output per HPD Configuration. Otherwise, it will always read 0b. **Note:** HPD Output Value must be set to 1b before using this feature. **Note:** Software may not change the state of this bit while HPD Enable is set to 1b.	R/SC	0b
1	**HPD Configuration** 0: **HPD** is configured as an input 1: **HPD** is configured as an output **Note:** Software may not change the state of this bit while HPD Enable is set to 1b.	R/W	0b
0	**HPD Enable** When set, the **HPD** pin is enabled.	R/W	0b

13.4.2 HPD INTERRUPT STATUS REGISTER (HPD_INT_STS)

Address: 0C01h Size: 8 bits

The contents of this register reset to 00h when the HPD Enable bit in the HPD Control Register (HPD_CTL) is cleared.

BITS	DESCRIPTION	TYPE	DEFAULT
7:4	**RESERVED**	RO	-
3	**QUEUE_NOT_EMPTY** Indicates that the HPD Event Queue Register (HPD_QUEUE) is not empty. **Note:** This interrupt is cleared after HPD Event Queue Register (HPD_-QUEUE) is read. **Note:** This bit only has meaning when the **HPD** pin is configured as an input.	RO	0b
2	**HPD_HIGH** When set, indicates that an **HPD** Low to High transition has been detected. **Note:** The source of this input is a pulse and does not persist after being cleared. **Note:** This bit only has meaning when the **HPD** pin is configured as an input.	R/WC	0b
1	**HPD_LOW** When set, indicates that an **HPD** High to Low transition has been detected. **Note:** The source of this input is a pulse and does not persist after being cleared. **Note:** This bit only has meaning when the **HPD** pin is configured as an input.	R/WC	0b
0	**IRQ_HPD** When set, indicates that an HPD IRQ has been detected. **Note:** The source of this input is a pulse and does not persist after being cleared. **Note:** When the **HPD** pin is configured as an input, this bit indicates that an IRQ_HPD has been detected. When the **HPD** pin is configured as an output, this pin indicates that the request to generate an IRQ_HPD has completed.	R/WC	0b

13.4.3 HPD INTERRUPT ENABLE REGISTER (HPD_INT_EN)

Address: 0C02h Size: 8 bits

BITS	DESCRIPTION	TYPE	DEFAULT
7:4	**RESERVED**	RO	0h
3:0	**HPD Interrupt Enable** When “0”, prevents generation of the respective interrupt.	R/W	0h

13.4.4 HPD EVENT QUEUE REGISTER (HPD_QUEUE)

Address: 0C03h Size: 8 bits

This register implements a circular queue that records HPD events detected by the device.

As an event is detected, it is written into the next HPD Event status field that is available and initially starts at HPD Event 0 out of reset.

After this register is read, any entries with a valid HPD event field are cleared when software writes 01b into the respective locations. The device will keep track of the last valid entry and store the next received event into the next entry. For example, if entries 0 and 1 are valid, the device will always place the next event into location 2, regardless of whether or not entries 0 and 1 have been cleared by software.

The device will only add entries to empty fields. After writing to HPD Event 3, the device will write the next event into HPD Event 0, if available.

Each field is encoded as follows.

- 00b: HPD event field is empty
- 01b: HPD_HIGH Detected
- 10b: HPD_LOW Detected
- 11b: HPD_IRQ Detected

When HPD Enable is set to 0b and/or HPD Configuration is set to 1b, this register will always read 00h.

After four HPD events are recorded, any future HPD events shall be ignored until HPD event locations are freed up by software.

BITS	DESCRIPTION	TYPE	DEFAULT
7:6	**HPD Event 3**	R/WC	00b
5:4	**HPD Event 2**	R/WC	00b
3:2	**HPD Event 1**	R/WC	00b
1:0	**HPD Event 0**	R/WC	00b

13.4.5 IRQ_HPD MINIMUM TIME REGISTER (IRQ_HPD_MIN_TIME)

Address: 0C04h Size: 8 bits

BITS	DESCRIPTION	TYPE	DEFAULT
7:0	**IRQ HPD Minimum Time** This field defines the minimum amount of time that **HPD** must be deasserted for an IRQ event to be recognized. De-assertions for less than this amount of time shall be ignored. The programmed time is (50 us * IRQ_HPD_MIN_TIME) + 100 us. **Note:** Software may not change the state of this register while HPD Enable is set to 1b.	R/W	0h

13.4.6 IRQ_HPD MAXIMUM TIME REGISTER (IRQ_HPD_MAX_TIME)

Address: 0C05h Size: 8 bits

BITS	DESCRIPTION	TYPE	DEFAULT
7:0	**IRQ HPD Maximum Time** This field defines the maximum amount of time that **HPD** shall be deasserted for an IRQ event to be recognized. Assertions for more than this amount of time shall not be determined to be an HPD interrupt event. The programmed time is (50 us * IRQ_HPD_MAX_TIME) + 100 us. **Note:** Software may not change the state of this register while HPD Enable is set to 1b.	R/W	0h

13.4.7 HPD HIGH DETECT TIME REGISTER (HPD_HIGH_DET_TIME)

Address: 0C06h Size: 16 bits

BITS	DESCRIPTION	TYPE	DEFAULT
15:0	**HPD High Detect Time** This field defines the amount of time that **HPD** must be asserted, after initially being low, in order to detect an HPD_HIGH event. The programmed time is (50 us * HPD_HIGH_DET_TIME) + 100 us. **Note:** Software may not change the state of this register while HPD Enable is set to 1b.	R/W	0h

13.4.8 HPD LOW DETECT TIME REGISTER (HPD_LOW_DET_TIME)

Address: 0C08h Size: 16 bits

BITS	DESCRIPTION	TYPE	DEFAULT
15:0	**HPD Low Detect Time** This field defines the amount of time that **HPD** must be deasserted, after initially being high, in order to detect an HPD_LOW event. The programmed time is (50 us * HPD_LOW_DET_TIME) + 100 us. **Note:** Software may not change the state of this register while HPD Enable is set to 1b.	R/W	0h

13.4.9 HPD IRQ GENERATION TIME REGISTER (HPD_IRQ_GEN)

Address: 0C0Ah Size: 8 bits

BITS	DESCRIPTION	TYPE	DEFAULT
7:0	**HPD IRQ Generation Time** This field defines the amount of time, in units of 50 us, that **HPD** shall be deasserted after the Generate IRQ bit in the HPD Control Register (HPD_CTL) is set. After this amount of time, the **HPD** pin will be returned to the state defined in HPD Output Value. **Note:** Software may not change the state of this register while HPD Enable is set to 1b.	R/W	0h

14.0 WATCHDOG TIMER (WDT)

14.1 General Description

The function of the Watchdog Timer (WDT) is to provide a mechanism to detect if the device has failed.

When enabled, the Watchdog Timer circuit will generate a WDT initiated system reset if the user program fails to reload the WDT within a specified length of time known as the WDT Interval.

The Watchdog timer operates off of the 20 KHz Keep Alive Oscillator or 48 MHz Relaxation Oscillator depending on the resolution selected by WDT_UNITS bit in Watchdog Control Register (WDT_CTL).

A watchdog timer initiated system reset is indicated by assertion of the WDT_INT bit in the Interrupt Enable Register (INT_EN). In order to clear the WDT_INT bit, the WDT_STS bit in Watchdog Control Register (WDT_CTL) must be cleared.

14.2 WDT Operation

14.2.1 WDT ACTIVATION MECHANISM

The WDT is activated by the following sequence of operations during normal operation:

1. Clear the WDT_EN bit of Watchdog Control Register (WDT_CTL).
2. Load the Watchdog Load Register (WDT_LOAD) with the count value.
3. Set the WDT_EN bit of Watchdog Control Register (WDT_CTL).

The WDT Activation Mechanism starts the WDT decrementing counter.

14.2.2 WDT DEACTIVATION MECHANISM

The WDT is deactivated by clearing the WDT_EN bit in the Watchdog Control Register (WDT_CTL). The WDT Reload Mechanism places the WDT in a low power state in which clocks are gated and the counter stops decrementing.

14.2.3 WDT RELOAD MECHANISM

If the WDT is not reloaded within periods that are shorter than the programmed watchdog interval, the WDT will underflow, a WDT reset will be generated, and the WDT_STS bit will be set in the Watchdog Control Register (WDT_CTL).

It is the responsibility of software to continually execute sections of code which reload the watchdog timer (WDT), causing the counter to be reloaded via the WDT Activation Mechanism or writing to the Watchdog Kick Register (WDT_KICK).

14.2.4 WDT INTERVAL

The WDT Interval is the time it takes for the WDT to decrements from the Watchdog Load Register (WDT_LOAD) value to 0h.

14.3 I2C/SPI Writes

When a watchdog interrupt is pending, all write operations are blocked until the interrupt WDT_STS bit in the Watchdog Control Register (WDT_CTL) is cleared.

14.4 Watchdog Timer Registers

This section details the watchdog timer registers. For an overview of the entire device register map, refer to Section 4.0, "Register Map," on page 18.

TABLE 14-1: WATCHDOG TIMER REGISTER MAP

Address	Register Name (Symbol)
3000h	Watchdog Control Register (WDT_CTL)
3002h	Watchdog Kick Register (WDT_KICK)
3004h	Watchdog Count Register (WDT_COUNT)
3008h	Watchdog Load Register (WDT_LOAD)
300Ch – 33FFh	Reserved for future expansion

Note: RESERVED address space must not be written under any circumstances. Failure to heed this warning may result in untoward operation and unexpected results.

14.4.1 WATCHDOG CONTROL REGISTER (WDT_CTL)

Address: 3000h Size: 16 bits

Bits	Description	Type	Default
15:4	**RESERVED**	RO	0h
3	**WDT_UNITS** 0: Units of 1 ms are used. 1: Units of 1 us are used. **Note:** When units of 1 us are used, the 48 MHz Relaxation Oscillator must be enabled and device current consumption will increase by over 400 uA. Otherwise, when units are 1 ms, the 20 KHz Keep Alive Oscillator is used.	R/W	0b
2	**WDT_SIM** Used for speeding up the watchdog timer for simulation. Watchdog operates off of a 1 MHz clock when set. When this bit is set, WDT_UNITS shall be set to 1b.	R/W	0b
1	**WDT_STS** WDT Status is set by hardware if the last reset of the device was caused by an underflow of the WDT. This bit must be cleared by software writing a '1' to this bit. Writing a '0' to this bit has no effect.	R/WC	Note 14-1
0	**WDT_EN** When set, the watchdog timer is enabled. Clearing this bit disables the watchdog timer. **Note:** Due to clock domain synchronization, several clock cycles may elapse before this takes affect and the WDT counter begins to decrement.	R/W	0b

Note 14-1 Default value depends on whether a WDT initiated reset occurred, in which the case the value is 1b. Otherwise the value is 0b.

14.4.2 WATCHDOG KICK REGISTER (WDT_KICK)

Address: 3002h Size: 8 bits

Bits	Description	Type	Default
7:1	**RESERVED**	RO	0h
0	**KICK** Writes of any value to this register shall cause the watchdog timer to be immediately reloaded with the value defined in Watchdog Load Register (WDT_LOAD) and start decrementing when the WDT_EN bit in the Watchdog Control Register (WDT_CTL) is set to 1b. Writes to this register when WDT_EN is 0b have no affect. **Note:** Reads of this register always return 0h. **Note:** Due to clock domain synchronization, several clock cycles may elapse before this takes affect and the counter reloads.	R/W	0h

14.4.3 WATCHDOG COUNT REGISTER (WDT_COUNT)

Address: 3004h Size: 32 bits

Bits	Description	Type	Default
31:0	**WDT_COUNT** This read only register provides a snapshot of the current state of the timer. The lower byte shall always be read first. Reading of the lower order byte results in a snapshot of the counter taken. This ensures that when the upper bytes are read the state of the counter is remains in sync.	RO	FFFF_FFFFh

WDT_COUNT[7:0] = 0x4h

WDT_COUNT[15:8] = 0x5h

WDT_COUNT[23:16] = 0x6h

WDT_COUNT[31:24] = 0x7h

14.4.4 WATCHDOG LOAD REGISTER (WDT_LOAD)

Address: 3008h Size: 32 bits

Bits	Description	Type	Default
31:0	**WDT_LOAD** Writing this register reloads the Watch Dog Timer counter. This register can be programmed only when WDT_EN = 0. Zero is not a valid load value.	R/W	FFFF_FFFFh

WDT_LOAD[7:0] = 0x8h

WDT_LOAD[15:8] = 0x9h

WDT_LOAD[23:16] = 0xAh

WDT_LOAD[31:24] = 0xBh

15.0 OPERATIONAL CHARACTERISTICS

15.1 Absolute Maximum Ratings*

Supply Voltage (**VS**) (Note 15-1) . . . . . -0.3 V to +6.0 V

Supply Voltage (**VDD33IO**, **3V3_VBUS**, **3V3_ALW**, **VDD_12C**, **VDD18**) (Note 15-1) . . . . . 0 V to +4.0 V

Positive voltage on input signal pins, with respect to ground . . . . . +6.0 V

Negative voltage on input signal pins, with respect to ground . . . . . -0.5 V

Storage Temperature . . . . . -55°C to +150°C

Lead Temperature Range . . . . . Refer to JEDEC Spec. J-STD-020

HBM ESD Performance . . . . . +/-8 kV

Note 15-1 When powering this device from laboratory or system power supplies, it is important that the absolute maximum ratings not be exceeded or device failure can result. Some power supplies exhibit voltage spikes on their outputs when AC power is switched on or off. In addition, voltage transients on the AC power line may appear on the DC output. If this possibility exists, it is suggested to use a clamp circuit.

*Stresses exceeding those listed in this section could cause permanent damage to the device. This is a stress rating only. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. Functional operation of the device at any condition exceeding those indicated in Section 15.2, "Operating Conditions**", Section 15.5, "DC Characteristics", or any other applicable section of this specification is not implied.

15.2 Operating Conditions**

Supply Voltage (**VS**) . . . . . +4.75 V to +5.25 V

Supply Voltage (**VDD33IO**, **3V3_VBUS**, **3V3_ALW**) . . . . . +3.1 V to +3.47 V

Supply Voltage (**VDD_I2C**) . . . . . Note 15-2

Supply Voltage (**VDD18**) . . . . . +1.62 V to +1.98 V

Positive voltage on input signal pins, with respect to ground . . . . . +3.3 V

Negative voltage on input signal pins, with respect to ground . . . . . -0.3 V

Power Supply Rise Time Max (T_{RT}) (Figure 15-1) *100ms*

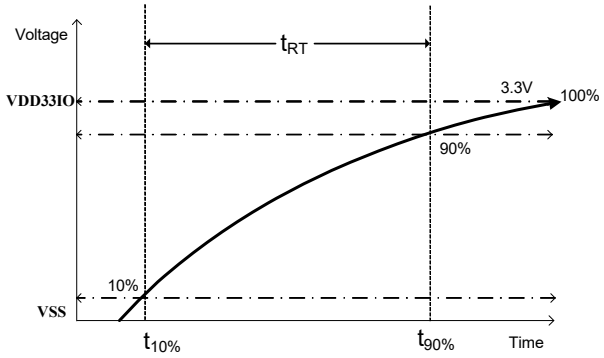
Commercial Ambient Operating Temperature in Still Air (T_A) . . . . . 0°C to +70°C

Industrial and Grade 3 Automotive Ambient Operating Temperature in Still Air (T_A) . . . . . -40°C to +85°C

Note 15-2 To operate the I^2C interface at 1.8 V, connect to 1.8 V (+/-10%), to operate the I^2C interface at 3.3 V, connect to 3.3 V (+/-10%)

**Proper operation of the device is guaranteed only within the ranges specified in this section.

FIGURE 15-1: SUPPLY RISE TIME MODEL



15.3 Package Thermal Specifications

TABLE 15-1: PACKAGE THERMAL PARAMETERS

Parameter	Symbol	°C/W
Thermal Resistance Junction to Ambient	Θ_{JA}	49
Thermal Resistance Junction to Top of Case	Θ_{JC}	6
Thermal Resistance Junction to Board	Θ_{JB}	29
Thermal Resistance Junction to Bottom of Case	Ψ_{JT}	0.7

Note: Thermal parameters are measured or estimated for devices in a multi-layer 2S2P PCB per JESDN51.

TABLE 15-2: POWER DISSIPATION

Parameter	Symbol	Max	Units
Power Dissipation	P_{dis}	55	mW

Note: This is the worst-case power dissipation as a consequence of maximum loading (before current-limiting protections take effect) upon the internal VBUS power switch, VCONN power switch, 3.3V power-ORing switch, analog blocks, and core digital logic.

15.4 Current Consumption

TABLE 15-3: DEVICE CURRENT CONSUMPTION

Power State	Supply Current		
	Typical	Max	Units
RESET	***110***	**-**	µA
SLEEP	***15***	**-**	µA
HIBERNATE	***80***	**-**	µA
STANDBY	***1.20***	**-**	mA
ATTACHED IDLE (FRS Enabled)	***1.40***	**-**	mA
ATTACHED IDLE (FRS Disabled)	***1.25***	**-**	mA
ACTIVE (with PD packet transmitting)	**-**	***15.0***	mA

Note 1: This table details the power consumption of the UPD350 device as measured during various modes of operation. Refer to Section 7.2, "Power States" for additional information. Power dissipation is determined by temperature, supply voltage, and external source/sin k requirements. Maximum values represent very short bursts of activity over a small amount of time. Typical values represent averaged current consumption over time.

2: **SLEEP** power state is achieved with **PWR_DN** pin asserted

3: **STANDBY** is equivalent to USB Type-C® specification's Unattached.SRC/Unattached.SNK

4: Currents measured with all 3.3V rails tied together.

15.5 DC Characteristics

TABLE 15-4: DC ELECTRICAL CHARACTERISTICS

Parameter	Symbol	Min	Typ	Max	Units	Notes
IS Type Input Buffer						
Low Input Level	V_{ILI}	-0.3		0.8	V	
High Input Level	V_{IHI}	2.0		3.6	V	
Negative-Going Threshold	V_{ILT}	1.21	1.33	1.8	V	Schmitt trigger
Positive-Going Threshold	V_{IHT}	1.31	1.58	1.8	V	Schmitt trigger
Schmitt Trigger Hysteresis (V_{IHT} - V_{ILT})	V_{HYS}	100	133	200	mV	
Input Leakage (V_{IN} = **VSS** or **VDDIO**)	I_{IH}	-10		10	µA	Note 15-3
Input Capacitance	C_{IN}			3	pF	
O8 Type Output Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = -8 mA
High Output Level	V_{OH}	**VDD33IO** - 0.4			V	I_{OH} = 8 mA

TABLE 15-4: DC ELECTRICAL CHARACTERISTICS (CONTINUED)

Parameter	Symbol	Min	Typ	Max	Units	Notes
OD8 Type Output Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = -8 mA
I2C Type Buffer						Note 15-4

Note 15-3 This specification applies to all inputs and tri-stated bi-directional pins. Internal pull-down and pull-up resistors add +/- 50 µA per-pin (typical).

Note 15-4 The I2C type buffer conforms to the NXP *I²C-Bus Specification* (UM10204, Rev. 6). Refer to the *I²C-Bus Specification* for additional information.

TABLE 15-5: VCONN SOURCE DC PARAMETERS

Parameter	Symbol	Min	Typ	Max	Units	Notes
ILIM	I_{LIM_VCONN}		600		mA	VS=5V
On Resistance	R_{ON_VCONN}		270		mΩ	

TABLE 15-6: POWER SWITCH DC PARAMETERS

Parameter	Symbol	Min	Typ	Max	Units	Notes
VSW Load	VSW_Load			100	mA	3V3_ALW/3V3_VBUS = 3.3V
VSW Resistance	R_VSW		500		Ω	

15.6 AC Characteristics and Timing

This section details the various AC timing specifications of the device.

15.6.1 RESET_N TIMING

Figure 15-2 illustrates the RESET_N timing requirements. Assertion of RESET_N is not a requirement. However, if used, it must be asserted for the minimum period specified

FIGURE 15-2: RESET_N TIMING

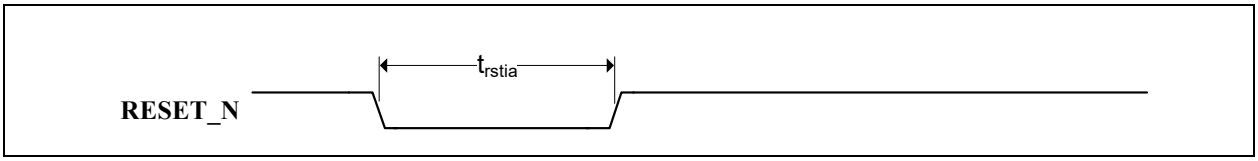


TABLE 15-7: RESET_N TIMING VALUES

Symbol	Description	Min	Typ	Max	Units
t_{rstia}	RESET_N input assertion time	1			µs

15.6.2 I²C SLAVE INTERFACE *(UPD350-A/UPD350-C ONLY)*

Figure 15-3 illustrates the I²C slave interface timing requirements. The I²C slave interface can operate in Standard Mode, Fast Mode, or Fast Mode Plus. Refer to Section 5.0, "I2C Slave Controller (UPD350-A/UPD350-C Only)," on page 19 for additional information.

FIGURE 15-3: I²C SLAVE TIMING

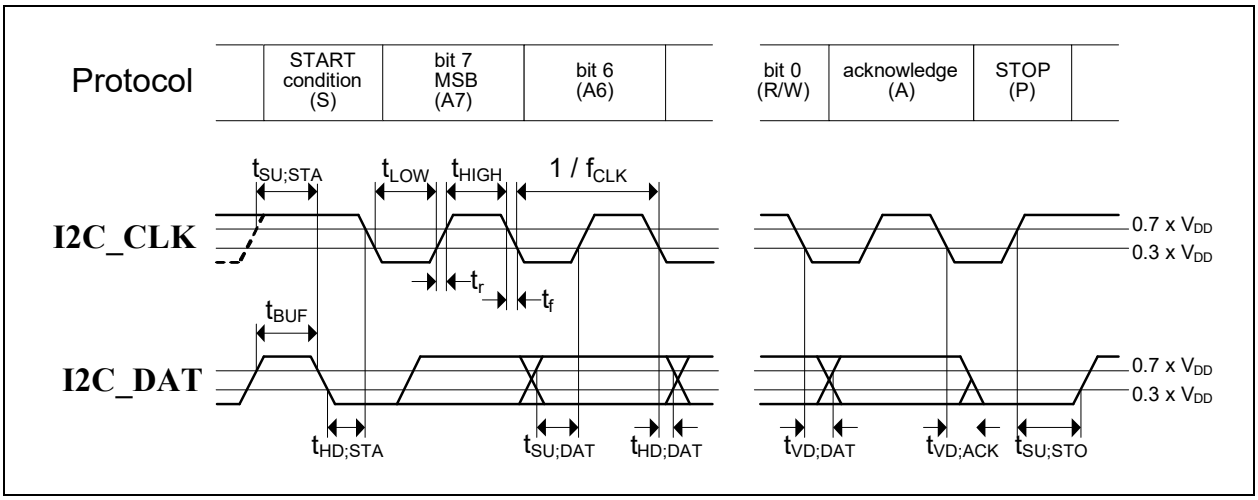


TABLE 15-8: I²C SLAVE TIMING VALUES

Symbol	Description	Min	Max	Units
f_{CLK}	**I2C_CLK** clock frequency	0	1000	kHz
t_{BUF}	Bus free time between a STOP and START condition	0.5		µs
$t_{HD;STA}$	Hold time (repeated) START condition	0.26		µs
$t_{SU;STA}$	Setup time for repeated START condition	0.26		µs
$t_{SU;STO}$	Setup time for STOP condition	0.26		µs
$t_{HD;DAT}$	Data hold time	0		ns
$t_{VD;ACK}$	Data valid acknowledge time (Note 15-5)	0.05	0.45	µs
$t_{VD;DAT}$	Data valid time (Note 15-6)	50	450	ns
$t_{SU;DAT}$	Data setup time	50		ns
t_{LOW}	LOW period of the **I2C_CLK** clock	0.5		µs
t_{HIGH}	HIGH period of the **I2C_CLK** clock	0.26		µs
t_f	Fall time of **I2C_CLK** and **I2C_DAT** (Note 15-7)(Note 15-8)		120	ns
t_r	Rise time of **I2C_CLK** and **I2C_DAT** (Note 15-7)(Note 15-8)		120	ns
t_{SP}	Pulse width of spikes that must be suppressed by the input filter (Note 15-9)		50	ns

Note 15-5 $t_{VD;ACK}$ = time for Acknowledgment signal from **I2C_CLK** LOW to **I2C_DAT** (out) LOW.

Note 15-6 $t_{VD;DAT}$ = minimum time for **I2C_DAT** data out to be valid following **I2C_CLK** LOW.

Note 15-7 A master device must internally provide a hold time of at least 300 ns for the **I2C_DAT** signal (refer to the V_{IL} of the **I2C_CLK** signal) in order to bridge the undefined region **I2C_CLK**'s falling edge.

Note 15-8 The maximum t_f for the **I2C_DAT** and **I2C_CLK** bus lines is specified at 300 ns. The maximum fall time for the **I2C_DAT** output stage t_f is specified at 250 ns. This allows series protection resistors to be connected between the **I2C_DAT** and **I2C_CLK** pins and the respective bus lines without exceeding the maximum specified t_f.

Note 15-9 Input filters on the **I2C_DAT** and **I2C_CLK** inputs suppress noise spikes less than 50 ns.

15.6.3 SPI SLAVE INTERFACE *(UPD350-B/UPD350-D ONLY)*

Figure 15-4 and Figure 15-5 illustrate the SPI slave interface input and output timing requirements, respectively. Refer to Section 6.0, "SPI Slave Controller (UPD350-B/UPD350-D Only)," on page 25 for additional information.

FIGURE 15-4: SPI SLAVE INPUT TIMING

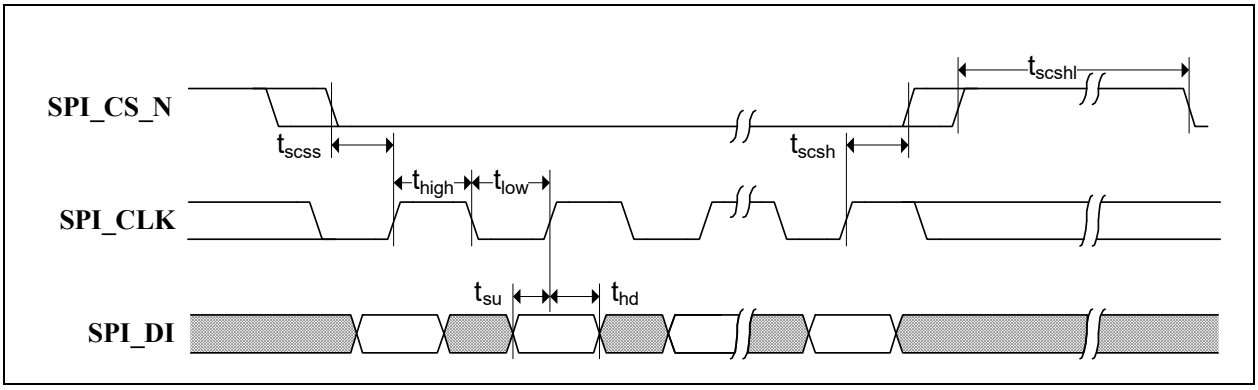


FIGURE 15-5: SPI SLAVE OUTPUT TIMING

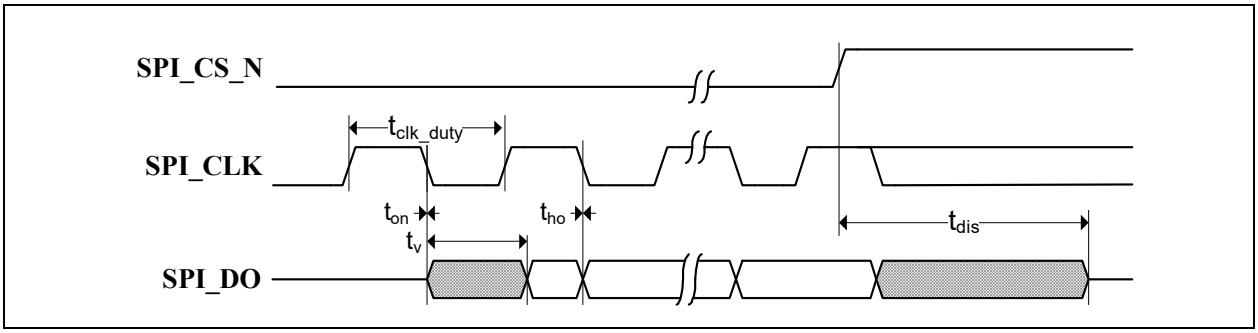


TABLE 15-9: SPI TIMING VALUES

Symbol	Description	Min	Typ	Max	Units
f_{sck}	**SPI_CLK** clock frequency			25	MHz
t_{clk_duty}	**SPI_CLK** high/low duty cycle	40		60	%
t_{scss}	**SPI_CS_N** setup time to **SPI_CLK**	5			ns
t_{scsh}	**SPI_CS_N** hold time from **SPI_CLK**	5			ns
t_{scshl}	**SPI_CS_N** inactive time	100			ns
t_{su}	Data input setup time to **SPI_CLK**	10			ns
t_{hd}	Data input hold time from **SPI_CLK**	4			ns
t_{on}	Data output turn on time from **SPI_CLK**	0			ns
t_v	Data output valid time from **SPI_CLK**			Note 15-10	ns
t_{ho}	Data output hold time from **SPI_CLK**	0			ns
t_{dis}	Data output disable time from **SPI_CS_N** inactive			20	ns

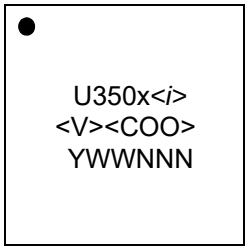
Note 15-10 8.5 or 8.0, depending on loading of 30pF or 10pF, respectively.

16.0 PACKAGE INFORMATION

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging

FIGURE 16-1: PACKAGE MARKING INFORMATION

28-SQFN

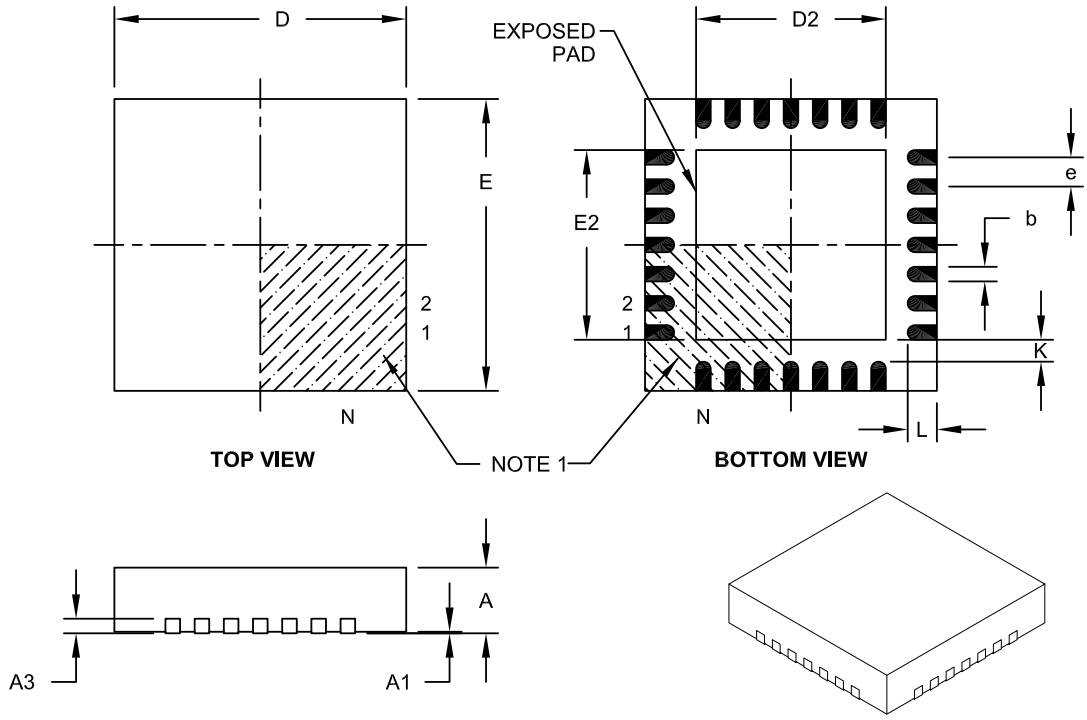


Legend:		
	x	UPD350 version (“A” or “C” = I^2C interface, “B” or “D” = SPI interface)
	<*i*>	Industrial temperature range indicator
	<V>	Assembly vendor code
	<COO>	Country code
	Y	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging.

FIGURE 16-2: 28-QFN PACKAGE (DRAWING & DIMENSIONS)

28-Lead Plastic Quad Flat, No Lead Package (MK) – 4x4x0.9 mm Body [QFN]



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		0.40 BSC	
Overall Height	A	0.80	0.85	0.90
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		4.00 BSC	
Exposed Pad Width	E2	2.50	2.60	2.70
Overall Length	D		4.00 BSC	
Exposed Pad Length	D2	2.50	2.60	2.70
Contact Width	b	0.17	0.20	0.25
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

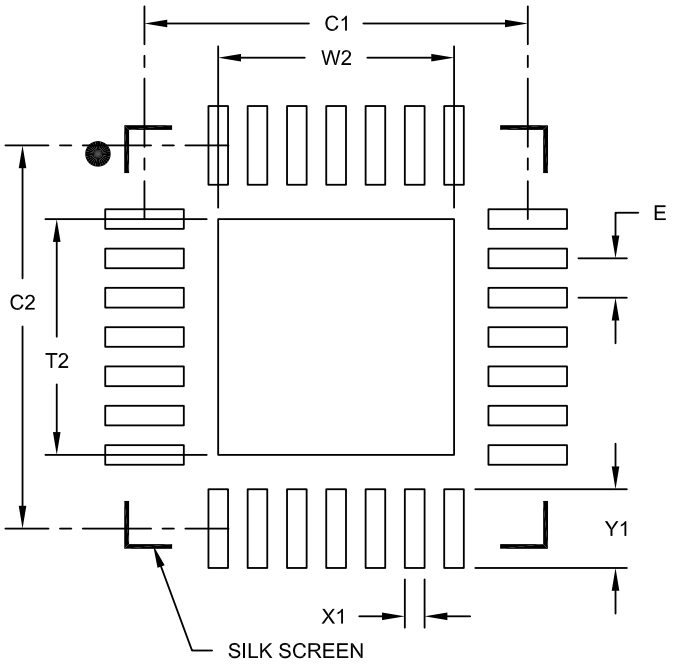
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-144A

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging.

FIGURE 16-3: 28-QFN PACKAGE (LAND PATTERN)

28-Lead Plastic Quad Flat, No Lead Package (MK) – 4x4x0.9 mm Body [QFN]



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.40 BSC	
Optional Center Pad Width	W2			2.40
Optional Center Pad Length	T2			2.40
Contact Pad Spacing	C1		3.90	
Contact Pad Spacing	C2		3.90	
Contact Pad Width	X1			0.20
Contact Pad Length	Y1			0.80

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2144A

APPENDIX A: DATA SHEET REVISION HISTORY

TABLE A-1: REVISION HISTORY

Revision	Section/Figure/Entry	Correction
DS00002643C (01-27-20)	Cover	Added “Automotive” to list of Target Applications.
	All	Added hyperlinks to registers and register settings throughout document.
	Section 2.0, "Introduction," on page 7	Added information for Standalone DFP mode to UPD350 Family Differences.
	Figure 2-1	Diagram modified to include VBUS power circuitry. Wording of I^2C/SPI connected device modified to include USB Smart Hub.
	Section 3.0, "Pin Descriptions and Configuration," on page 9	Added information for Standalone DFP mode to UPD350-A / UPD350-C Alternate GPIO Functions in Standalone DFP/UFP Mode and Section 3.2, "Pin Descriptions," on page 13.
	Section 3.2, "Pin Descriptions," on page 13	• Corrected GPO4 note: “The GPO4 general purpose signal must be augmented with a weak pull-down in order to prevent the device from entering a test mode on start-up.” • Updated GPIO note: “In companion mode, these signals should be tied to ground when unused.”
	Section 4.0, "Register Map," on page 18	Added new chapter.
	Section 7.0, "Clocks, Resets, and Power Management," on page 30	Added Clocks and Power Management Registers section.
	Section 8.0, "System Control," on page 35	Added System Interrupts section.
		Expansion of General Purpose I/O, External Overcurrent Detection, General Purpose Timer, System Control Registers sections to include additional feature detail and register control information.
	Section 9.0, "Cable Plug Orientation and Detection," on page 67	Expansion of CC Comparator, DFP Operation, DRP Operation (Legacy), DRP Offload, Collision Avoidance,Fast Role Swap (FRS), VCONN Operation, VBUS Detection, and Back-Drive Detection-sections to include additional feature detail and register control information.
		Added Standalone DFP (UPD350-A/UPD350-C Only) section.
	Section 10.0, "Baseband CC Interface (BCI)," on page 108	Added Baseband CC Interface Registers section.

TABLE A-1: REVISION HISTORY (CONTINUED)

Revision	Section/Figure/Entry	Correction
	Section 10.3.1, "CC RX DAC Control Register (CC_RX_DAC_CTL)," on page 110, Section 10.3.4, "BB TX Risex Registers (BB_TX_RISEx)," on page 111, Section 10.3.5, "BB TX Fallx Registers (BB_TX_FALLx)," on page 112	Added recommended field values.
	Section 11.0, "Power Delivery MAC," on page 113	Expansion of PD MAC Transmitter, PD MAC Receiver, PD MAC BIST, sections to include additional feature detail and register control information.
		Added Power Delivery MAC Registers section.
	Section 12.0, "Power Switch," on page 173	Expansion of Power Switch to include additional feature details
		Added Software Override, Power Switch Interrupts, Power Switch Registers sections.
	Section 13.0, "HDMI/DisplayPort Hot Plug Detect (HPD)," on page 187	Added HPD Receiver, HPD Transmission, DisplayPort HPD Registers sections.
	Section 14.0, "Watchdog Timer (WDT)," on page 194	Expansion of General Description to include additional feature details
		Added WDT Operation, I2C/SPI Writes, Watchdog Timer Registers sections.
	Table 15-4, "DC Electrical Characteristics"	Max value for Schmitt Trigger Hysteresis changed from “0” to “200”.
DS00002643B (06-19-18)	Cover	Added new highlight bullet: “Companion PD Controller for Microchip USB Hub / SoC”.
	Section 2.0, "Introduction," on page 7	Edited first paragraph and added new content.
DS00002643A (03-26-18)	Initial Document Release	

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://www.microchip.com/support

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	X	[X](1)	[X]	/XXX	Vxx
Device	Version	Tape & Reel Option	Temp. Range	Package	Automotive Code

Device:	UPD350	
Version:	A	= +1.8V-3.3V I²C Interface, Dead battery support
	B	= SPI Interface, Dead battery support
	C	= +1.8V-3.3V I²C Interface, No dead battery support
	D	= SPI Interface, No dead battery support
Tape and Reel Option:	Blank	= Standard packaging (tray)
	T	= Tape and Reel (Note 1)
Temperature Range:	Blank	= 0°C to +70°C (Commercial)
	-I	= -40°C to +85°C (Industrial and Grade 3 Automotive)
Package:	Q8X	= 28-pin QFN
Automotive Code:	Vxx	= 3 character code with “V” prefix, specifying automotive part

Examples:

a) UPD350A/Q8X
I²C Interface, Dead battery support
Standard packaging,
Commercial temperature,
28-pin QFN package

b) UPD350BT/Q8X
SPI Interface, Dead battery support
Tape and Reel,
Commercial temperature,
28-pin QFN package

c) UPD350B-I/Q8XVAA
SPI Interface, Dead Battery support
Standard packaging,
Industrial Temperature,
28-pin QFN package, Automotive

d) UPD350C/Q8X
I²C Interface, No dead battery support
Standard packaging,
Commercial temperature,
28-pin QFN package

e) UPD350DT-I/Q8X
SPI Interface, No dead battery support
Tape and Reel,
Industrial temperature,
28-pin QFN package

Note 1: Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option.

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

ISBN: 9781522454076

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-72400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7288-4388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820