Technical Reference Manual

TPS25730A Technical Reference Manual

Table of Contents

List of Figures

List of Tables

1 Read This First

1.1 About This Manual

This manual covers the features and peripherals supported by the TPS25730A USB Type-C® and PD controller devices. The document covers the Host Interface (4CC) Command and register read/write descriptions

1.2 Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers may be shown with the suffix h or the prefix 0x. For example, the following number is 40 hexadecimal (decimal 64): 40h or 0x40.
- Registers in this document are shown in figures and described in tables.
  - Byte convention for N bytes is 0 through (N-1) bytes.
  - Bit convention for N bits is 0 through (N-1) bits.
  - Each register figure shows a rectangle divided into fields that represent the fields of the register. Each field is labeled with its bit name, its beginning and ending bit numbers above, and its read/write properties with default reset value below. A legend explains the notation used for the properties.
  - Reserved bits in a register figure can have one of multiple meanings:
    - Not implemented on the device
    - Reserved for future device expansion
    - Reserved for TI testing
    - Reserved configurations of the device that are not supported
  - Writing nondefault values to the Reserved bits could cause unexpected behavior and should be avoided.

1.3 Glossary

TI Glossary	This glossary lists and explains terms, acronyms, and definitions.

1.4 Related Documents

- USB Power Delivery Specification Revision 3.2 www.usb.org/developers/docs
- USB Type-C Cable and Connector Specification Revision 2.4. www.usb.org/developers/docs

1.5 Support Resources

TI E2E™ support forums are an engineer's go-to source for fast, verified answers and design help — straight from the experts. Search existing answers or ask your own question to get the quick design help you need.

Linked content is provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's Terms of Use.

1.6 Trademarks

TI E2E™ is a trademark of Texas Instruments.
USB Type-C® is a registered trademark of USB Implementers Forum.
All trademarks are the property of their respective owners.

2 Introduction

2.1 Purpose and Scope

This document describes the Host Interface for the TPS25730AS and TPS25730AD Type-C Port Switch / Power Delivery (PD) Controller device.

2.2 PD Controller Host Interface Description

2.2.1 Overview

The PD Controller provides one I2C target. The I2Ct target is meant to be connected to an Embedded Controller (EC).

The Host Interface defines how the registers are accessed from I2C target port and target addresses. The target address is selected by the customer using the ADCIN1 and ADCIN2 pins on the PD controller.

The Host Interface provides general status information to the controller of these I2C interfaces about the PD Controller, ability to control the PD Controller, status of USB Type-C® Port and communications to/from a connected device (Port Partner) and/or cable plug through USB PD messages. All Host Interface communication that uses the Unique I2C address is referred to as Unique Address Interface.

The PD Controller supports a register-based Unique Address Interface. Section 4.1 lists the Unique Address Interface registers and provides detailed Unique Address Interface register descriptions.

The key to the protocol diagrams is in the SMBus Specification, version 2.0 and is repeated here in part in Figure 2-1.

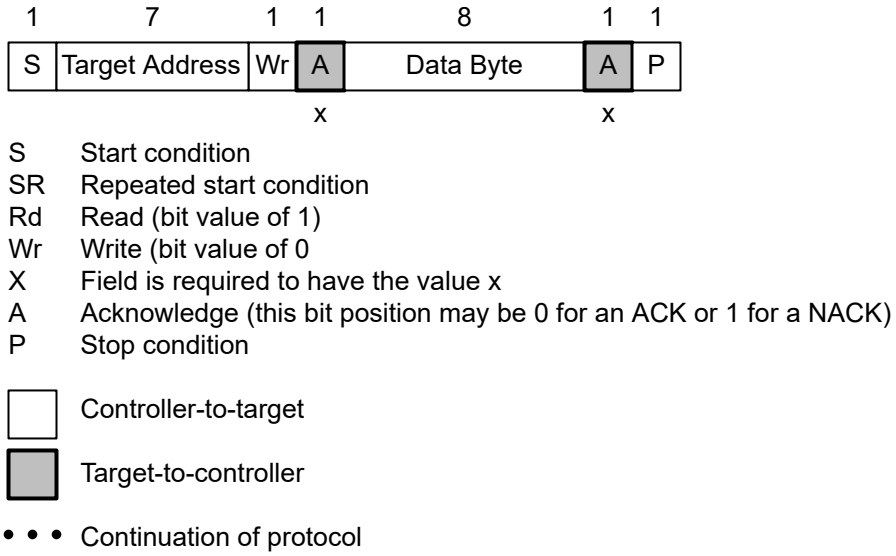

Figure 2-1. I2C Read/Write Protocol Key

2.3 Unique Address Interface

2.3.1 Unique Address Interface Protocol

The Unique Address Interface allows for complex interactions between an I2C controller and a single PD Controller. The I2C target unique address is used to receive or respond to Host Interface protocol commands. Figure 2-2 and Figure 2-3 show the write and read protocols, respectively. The Byte Count used during a register write can be longer than the number of bytes actually written, in other words the controller can issue the stop bit without writing N bytes. Similarly, during a register read, the controller can issue the stop bit before reading all N bytes.

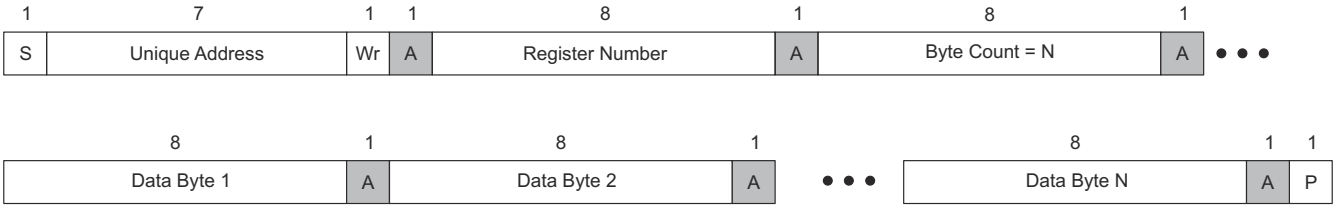

Figure 2-2. I2C Unique Address write register protocol

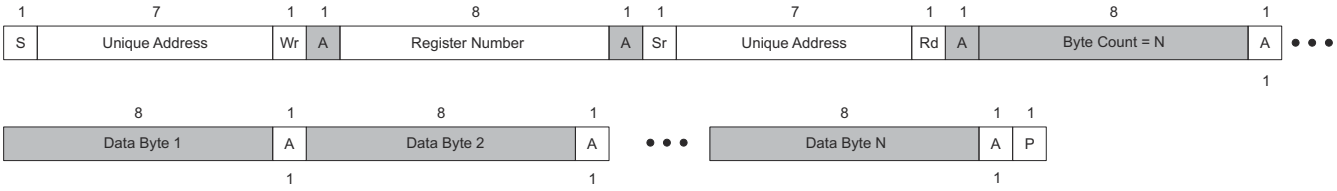

Figure 2-3. I2C Unique Address read register protocol

3 PD Controller Policy Modes

3.1 Overview

The PD Controller implements a mode for "SNK Policy" (issuing Requests for Sink contracts). Certain fields are referred to differently, the table below summarizes the naming conventions assumed.

Table 3-1. Naming Conventions

Reference Name	TPS25730AS Reference	TPS25730AD Reference
PP_EXT, PP3 Switch	PP_EXT	PP_HV

3.2 Sink Policy Mode

The PD Controller will always prepare its own Request message based on the settings in the *AUTO_NEGOTIATE_SINK* register (0x37) and the *TX_SINK_CAPS* register (0x33). The PD Controller will send its prepared Request message as soon as it is ready. The host can change the *AUTO_NEGOTIATE_SINK* register and the *TX_SINK_CAPS* register, then issue the 'ANeg' 4CC Task and the PD controller will re-negotiate the PD contract based on the updated values.

4 Register Overview

Note

Some registers contain 'Reserved' fields that should not be overwritten by an I2C Host Controller (EC). To prevent overwriting these 'Reserved' fields the host controller should follow the read-modify-write instruction before configuring any PD registers.

4.1 TPS25730A Registers

Table 4-1 lists the memory-mapped registers for the TPS25730A registers. All register offset addresses not listed in Table 4-1 should be considered as reserved locations and the register contents should not be modified.

Table 4-1. TPS25730A Registers

Offset	Acronym	Register Name	Section
8h	Command Register (CMD1) for I2Ct	Command Register (CMD1) for I2Ct	Section 4.1.1
9h	Data Register (DATA1) for CMD1	Data Register (DATA1) for CMD1	Section 4.1.2
1Ah	Status	Status	Section 4.1.3
26h	Power Path Status	Power Path Status	Section 4.1.4
30h	Received Source Capabilities	Received Source Capabilities	Section 4.1.5
33h	Transmit Sink Capabilities	Transmit Sink Capabilities	Section 4.1.6
34h	Active PDO Contract	Active PDO Contract	Section 4.1.7
35h	Active RDO Contract	Active RDO Contract	Section 4.1.8
37h	Autonegotiate Sink	Autonegotiate Sink	Section 4.1.9
40h	PD Status	PD Status	Section 4.1.10

Complex bit access types are encoded to fit into small table cells. Table 4-2 shows the codes that are used for access types in this section.

Table 4-2. TPS25730A Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

4.1.1 Command Register (CMD1) for I2Ct (Offset = 8h) [Reset = 00000000h]

Command Register (CMD1) for I2Ct is shown in Table 4-3.

Return to the Summary Table.

Command register. The PD Controller clears it on initialization and after completing a command.

Table 4-3. Command Register (CMD1) for I2Ct Field Descriptions

Bit	Field	Type	Reset	Description
31-0	Command	R/W	0h	Command register for the command interface. The PD Controller clears this register to 0x0 on initialization and after completing any recognized commands. Unrecognized commands are overwritten with !CMD.

4.1.2 Data Register (DATA1) for CMD1 (Offset = 9h) [Reset = 0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000h]

Data Register (DATA1) for CMD1 is shown in Table 4-4.

Return to the Summary Table.

Data register (DATA1) for the command interface (CMD1).

Table 4-4. Data Register (DATA1) for CMD1 Field Descriptions

Bit	Field	Type	Reset	Description
511-0	Data	R/W	0h	Data register (DATA1) for the command interface (CMD1). The first byte of this register will contain the return code if applicable, and the remaining bytes will contain the command's output

4.1.3 Status Register (Offset = 1Ah) [Reset = 0000000000h]

Status is shown in Table 4-5.

Return to the Summary Table.

Status bit field for events.

Table 4-5. Status Register Field Descriptions

Bit	Field	Type	Reset	Description
39-26	RESERVED	R	0h	
25-24	Acting as Legacy	R	0h	Indicates when PD Controller has gone into a mode where it is acting like a legacy (non PD) device. It can take approximately 10 seconds for the PD controller to determine that it is attached to a legacy source or sink. 0h = PD Controller is not in a legacy (non PD) mode 1h = PD Controller is acting like a legacy sink 3h = Acting as legacy sink due to dead-battery.
23-22	RESERVED	R	0h	
21-20	VBUS Status	R	0h	Indicates the present state of VBUS. 0h = At vSafe0V (less than 0.8V) 1h = At vSafe5V (4.75V to 5.5V) 2h = Within expected limits 3h = Not within any of the other specified ranges
19-7	RESERVED	R	0h	
6	Data Role	R	0h	PD controller data role. This is only valid once there is a connection. 0h = Upward-facing port (UFP)
5	Port Role	R	0h	Current state of PD Controller CCx terminations. This also indicates the PD Controller Power Role, once connected. This bit does not toggle during Unattached state transitions. 0h = PD Controller is in the Sink role
4	Plug Orientation	R	0h	Plug orientation indicator. Indicates port orientation when known (requires connection). 0h = Upside-up orientation (plug CC on CC1) 1h = Upside-down orientation (plug CC on CC2)
3-1	Connection State	R	0h	Details of a connected plug. 0h = No connection 1h = Port is disabled 2h = Corrosion Mitigaion (Ra/Ra) 3h = Debug connection (Rd/Rd) 4h = No connection Ra detected (Ra but no Rd) 5h = Debug connection (Rp/Rp) 6h = Connection present no Ra detected 7h = Connection present Ra detected
0	Plug Present	R	0h	Status of the plug 0h = No plug is connected 1h = A plug is connected

4.1.4 Power Path Status Register (Offset = 26h) [Reset = 0000000000h]

Power Path Status is shown in Table 4-6.

Return to the Summary Table.

Power Path Status. This is a hardware dependent register.

Table 4-6. Power Path Status Register Field Descriptions

Bit	Field	Type	Reset	Description
39-38	Power Source	R	0h	Indicates current PD Controller power source. NOTE: Since the Dead Battery flag forces PD Controller to be powered from VBUS, only 10b is valid when this flag is set. Any other setting indicates that the Dead Battery flag is not set. 0h = Reserved 1h = PD Controller is powered from VIN_3V3 2h = PD Controller is powered from VBUS 3h = Reserved
37-21	RESERVED	R	0h	
20-18	RESERVED	R	0h	
17-15	RESERVED	R	0h	
14-12	PP3 Switch	R	0h	Indicates current state of PP3 (PP_EXT). 0h = PP3 switch disabled 1h = PP3 switch currently disabled due to fault 3h = PP3 switch enabled (system input)
11-0	RESERVED	R	0h	

4.1.5 Received Source Capabilities Register (Offset = 30h) [Reset = 00000000000000000000000000000000000000000000000000000000h]

Received Source Capabilities is shown in Table 4-7.

Return to the Summary Table.

Received Source Capabilties. This register stores latest Source Capabilities message received over BMC.

Table 4-7. Received Source Capabilities Register Field Descriptions

Bit	Field	Type	Reset	Description
231-200	Source PDO 7	R	0h	Seventh Source Capabilities PDO received
199-168	Source PDO 6	R	0h	Sixth Source Capabilities PDO received
167-136	Source PDO 5	R	0h	Fifth Source Capabilities PDO received
135-104	Source PDO 4	R	0h	Fourth Source Capabilities PDO received
103-72	Source PDO 3	R	0h	Third Source Capabilities PDO received
71-40	Source PDO 2	R	0h	Second Source Capabilities PDO received
39-8	Source PDO 1	R	0h	First Source Capabilities PDO received
7-3	RESERVED	R	0h	
2-0	Number Valid PDOs	R	0h	Number of valid SPR PDOs in this register. Each PDO is 4 bytes. (max of 7)

4.1.6 Transmit Sink Capabilities Register (Offset = 33h) [Reset = 00000000000000000000000000000000000000000000000000000000000002h]

Transmit Sink Capabilities is shown in Table 4-8.

Return to the Summary Table.

Sink Capabilities for sending. This register stores PDOs for outgoing Sink Capabilities USB PD messages. The PD controller transmits the contents of this register as a Sink_Capabilities message after receiving a Get_Sink_Cap message unless its configuration or USB PD rules require a different response in the context. Each PDO in this TX_SINK_CAPS register follows the definition from the USB PD specification. For more details on the meaning of each field refer to the USB PD specification.

Table 4-8. Transmit Sink Capabilities Register Field Descriptions

Bit	Field	Type	Reset	Description
231-200	TX Sink PDO 7	R/W	0h	SPR Seventh Sink Capabilities PDO contents.
199-168	TX Sink PDO 6	R/W	0h	SPR Sixth Sink Capabilities PDO contents.
167-136	TX Sink PDO 5	R/W	0h	SPR Fifth Sink Capabilities PDO contents.
135-104	TX Sink PDO 4	R/W	0h	SPR Fourth Sink Capabilities PDO contents.
103-72	TX Sink PDO 3	R/W	0h	SPR Third Sink Capabilities PDO contents.
71-40	TX Sink PDO 2	R/W	0h	SPR Second Sink Capabilities PDO contents.
39-8	TX Sink PDO 1	R/W	0h	SPR First Sink Capabilities PDO contents.
7-3	RESERVED	R	0h	
2-0	Number Valid PDOs	R/W	2h	Number of valid PDOs in this register. Each PDO is 4 bytes. (max of 7)

4.1.7 Active PDO Contract Register (Offset = 34h) [Reset = 000000000000h]

Active PDO Contract is shown in Table 4-9.

Return to the Summary Table.

Power data object for active contract. This register stores PDO data for the current explicit USB PD contract, or all zeroes if no contract.

Table 4-9. Active PDO Contract Register Field Descriptions

Bit	Field	Type	Reset	Description
47-42	RESERVED	R	0h	
41-32	First PDO Control Bits	R	0h	Contains PDO specific information of the first PDO. It does not matter which PDO was selected, this field is always drawn from the first PDO.
31-0	Active PDO	R	0h	Power data object. This field contains the contents of the PDO requested by PD controller as sink and accepted by source, once it is accepted by source.

4.1.8 Active RDO Contract Register (Offset = 35h) [Reset = 00000000h]

Active RDO Contract is shown in Table 4-10.

Return to the Summary Table.

Power data object for the active contract. This register stores the RDO of the current explicit USB PD contract, or all zeroes if no contract.

Table 4-10. Active RDO Contract Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	Object Position	R	0h	Corresponding PDO location in the Source_Capabilities Message.
27	Give Back Flag	R	0h	When set, the device will respond to a GotoMin message by reducing its load to minimum operating current
26	Capabality Missmatch	R	0h	Set when Source cannot satisfy the Sink's power or voltage requirements per Souce Capabilities
25	USB Communication Capable	R	0h	When set, the device has USB data lines and is capable of communicating using USB2, USB3 or USB4 protocols
24	No USB Suspend	R	0h	This flag is used to indicate what actions are taken in USB Suspend.
23	Unchunked Supported	R	0h	When set, the port supports chunked and unchucked messagse.
22-20	RESERVED	R	0h	
19-10	Operating Current	R	0h	Operating current (10mA per LSB)
9-0	Max Min Operation Current	R	0h	Shall be assigned same as Operating Current field

4.1.9 Autonegotiate Sink Register (Offset = 37h) [Reset = 0000000000000036h]

Autonegotiate Sink is shown in Table 4-11.

Return to the Summary Table.

Configuration for sink power negotiations. This register defines the voltage range between which the system can function properly, allowing the PD Controller to negotiate its own contracts. The register is set by any power up settings and overwritten by customer configurations or MCU.

Table 4-11. Autonegotiate Sink Register Field Descriptions

Bit	Field	Type	Reset	Description
63-62	RESERVED	R	0h	
61-52	Auto Neg Capabilities Mismatch Power	R/W	0h	Capabilities Mismatch Power Threshold. If the selected PDO offers less power than what is specified in this register, then the PD controller will assert the Capability Mismatch bit in its Request message unless NoCapabilityMismatch is set to 1. (250mW per LSB)
51-42	Auto Neg Min Voltage	R/W	0h	Minimum voltage to request. During PD power contract negotiation, the PD controller will only select voltages that are greater than or equal to the value specified in this field. Not used unless AutoComputeSinkMinVoltage=0. (50mV per LSB)
41-32	Auto Neg Max Voltage	R/W	0h	Maximum voltage to request. During PD power contract negotiation, the PD controller will only select voltages that are less than or equal to the value specified in this field. Not used unless AutoComputeSinkMinVoltage=0. (50mV per LSB) See description in AutoComputeSinkMinPower.
31-22	Auto Neg Sink Min Required Power	R/W	0h	Minimum operating power required by the Sink. The PD Controller will always attempt to receive this power level from the Source. (250mW per LSB)
21-12	Auto Neg Max Current	R/W	0h	Maximum current to request. The PD controller will not request more current than indicated by this field. The host should ensure that the max current for all PDO's in the TX_SINK_CAPS register do not exceed this value. (10mA per LSB).
11-8	RESERVED	R	0h	
7	RESERVED	R	0h	
6	Auto Disable Sink Upon Capability Mismatch	R/W	0h	Sink path and capability mismatch settings. If this bit is asserted, then any time the implicit or explicit power contract would cause the Capability Mismatch bit to be set the PD controller will disable the sinking path. This bit should only be asserted if the NoCapabilityMismatch bit is set to 0.
5	Auto Compute Sink Max Voltage	R/W	1h	Configuration for maximum voltage. The PD controller can automatically compute ANMaxVoltage, or allow the host to specify it. 0h = Provided by host 1h = Computed by PD controller
4	Auto Compute Sink Min Voltage	R/W	1h	Configuration for minimum voltage. The PD controller can automatically compute ANMinVoltage, or allow the host to specify it. 0h = Provided by host 1h = Computed by PD controller
3	No Capability Mismatch	R/W	0h	Configuration for capability mismatch in RDO. There is one conditions that will trigger a capability mismatch: • If the attached source does not offer a PDO whose power is greater or equal to the ANSinkCapMismatchPower field in this register. If the condition is true, then the PD controller will assert the capability mismatch bit in its request unless this bit is asserted. 0h = Capabiltiy mismatch enabled 1h = Capability mismatch disabled.

Table 4-11. Autonegotiate Sink Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2	Auto Compute Sink Min Power	R/W	1h	Minimum power sink requires. The minimum sink power is the largest power reported in any valid PDO in the TX_SINK_CAPS (0x33). The power for a particular PDO from the TX_SINK_CAPS follows for each supply type: • Battery Supply: OperatingPower • Variable Supply: MaxVoltage*OperatingCurrent • Fixed Supply: Voltage*OperatingCurrent. However, if the TX_SINK_CAPS register includes Battery supply type PDO(s), then ANSinkMinRequiredPower = maximum OperatingPower in a Battery supply type PDO. 0h = Provided by host 1h = Computed by PD controller
1	No USB Suspend	R/W	1h	Value used for the NoUSBSusp Flag in the RDO. This is as defined by USB PD.
0	Auto Neg RDO Priority	R/W	0h	Configuration for tie-breaker in PDO selection. The PD controller will find the set of PDOs that fulfill the voltage requirements. From that set of PDOs it will pick the one with higher power. If two acceptable PDOs have the same power, Fixed Supply Type is preferred, and then Variable Supply has second preference. If two PDOs have the same power and the same type, then this bit determines which PDO is selected. 0h = Higher voltage 1h = Lower voltage

4.1.10 PD Status Register (Offset = 40h) [Reset = 00000000h]

PD Status is shown in Table 4-12.

Return to the Summary Table.

Status of PD and Type-C state-machine. This register contains details regarding the status of PD messages and the Type-C state machine.

Table 4-12. PD Status Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	RESERVED	R	0h	
27-22	Error Recovery Details	R	0h	Reason for Error Recovery 0h = reset value: no error recovery 3h = System: fault input GPIO was asserted 4h = System: Over-voltage detected on the Px_VBUS pin 8h = System: OVP on CC detected
21-16	Hard Reset Details	R	0h	Reason for Hard Reset 0h = Reset value no hard reset 1h = Received from Port Partner 2h = Requested by host 3h = DR_Swap Message is a received when there is an active mode between the port-partners 4h = The source output voltage failed to settle at vSrcNew within tSrcSettle during high-low voltage transition. Applicable only when the port role is SRC 6h = The port-partner did not respond to SoftReset message with Accept within tSenderResponse time 7h = The port-partner did not respond to Request message with Accept within tSenderResponse time 8h = The port-partner did not respond to Accept message with PS_RDY within tPSTransition time 9h = The port-partner did not send Source_Capabilities message for a duration of tTypeCSinkWaitCap after presenting VBUS Ah = The port-partner did not respond to SoftReset messsage after nRetryCount attempts 11h = An unexpected message is received during power transitions
15-13	RESERVED	R	0h	
12-8	Soft Reset Details	R	0h	Reason for Soft Reset 0h = Reset value no soft reset 1h = Soft reset received from Port Partner 2h = Reserved 3h = Reserved 4h = Received source capabilities message was invalid 6h = Received an accept message unexpectedly 7h = Received a control message unexpectedly 8h = Received a GetSinkCap message unexpectedly 9h = Received a GetSourceCap message unexpectedly Ah = Received a GotoMin message unexpectedly Bh = Received a PS_RDY message unexpectedly Ch = Received a Ping message unexpectedly Dh = Received a Reject message unexpectedly Eh = Received a Request message unexpectedly Fh = Received a Sink Capabilities message unexpectedly 10h = Received Source Capabilities message unexpectedly 11h = Received a Swap message unexpectedly 12h = Received a Wait Capabilities message unexpectedly 13h = Received an unknown control message 14h = Received an unknown data message 17h = Received an Extended message unexpectedly 18h = Received an unknown Extended message 19h = Received a data message unexpectedly 1Ah = Received a Not Supported message unexpectedly 1Bh = Received a Get_Status message unexpectedly
7	RESERVED	R	0h	

Table 4-12. PD Status Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
6	Present PD Role	R	0h	Present PD power role. The PD Controller is acting under this PD power role. 0h = Sink
5-4	Port Type	R	0h	Present Type-C power role. The PD Controller is acting under this Type-C power role. 1h = Sink
3-2	CC Pullup	R	0h	CC Pull-up value. The pull-up value detected by PD Controller when in CC Pull-down mode. 0h = Not in CC pull-down mode / no CC pull-up detected 1h = USB Default current 2h = 1.5 A (SinkTxNG) 3h = 3.0 A (SinkTxOK)
1-0	RESERVED	R	0h	

5 4CC Task Detailed Descriptions

5.1 Overview

This section describes the 4CC Tasks defined by the PD Controller Host Interface. The Tasks are categorized into various sub-groups in this section. All Tasks that return data using the DATAx registers will always ensure the proper output data is loaded into those registers before setting the CMDx register to 0 to indicate Task completion. DATAx is never modified by PD Controller after CMDx has been changed to 0, to ensure the Host can retrieve data from the previously-executed Task, and to ensure the Host can load these registers for a future Task without risk of overwriting. Note that other registers can continue to be updated after a Task completes, as Tasks can have additional side effects.

Many of the Tasks return a status code in the first byte of the DATAx register. The standard Task response byte is defined in Table 5-1. The remaining DATAx bytes can be used at each Task's discretion.

Table 5-1. Standard Task Response

Description	**Tasks are a special form of Tasks that return a status code in the first byte of the DATAX register.**			
Output DATAX	**Bit**	**Name**	**Description**	
	Byte 1: Task Return Code			
	7:4	Reserved	Reserved for standard Tasks. May be used by certain Tasks for Task-specific return codes. Successful return codes can use this byte provided TaskResult is 0x0.	
	3:0	TaskResult	Standard Task return codes.	
			0x0	Task completed successfully.
			0x1	Task timed-out.
			0x2	Reserved.
			0x3	Task rejected.
			0x4	Task rejected because the Rx Buffer was locked. This is for Tasks that can require the PD controller to use the Rx Buffer.
			0x5-0xF	Reserved for standard Tasks. May be used by certain Tasks for Task-specific error codes. Treated as an error when encountered.

5.2 PD Message Tasks

5.2.1 'GSrC' - PD Get Source Capabilities

Table 5-2. 'GSrC' - PD Get Source Capabilities

Description	**The 'GSrC' Task instructs PD Controller to issue a Get_Source_Cap message to the Port Partner device at the first opportunity while maintaining policy engine compliance.**
INPUT DATAX	None
OUTPUT DATAX	Byte 1: Standard Task Return Code. See also Table 5-1.
Task Completion	The 'GSrC' Task completes either when the Source Capabilities message is received or the Task otherwise fails. The 'GSrC' Task shall be considered rejected if: • The Port Partner is a Sink and indicated (through previous Source or Sink Capabilities) it was not Dual-Role Power. • The Port Partner responds to the Get_Source_Cap message with a Reject or Not_Supported message. The 'GSrC' Task shall be considered timed-out if: • The Port Partner fails to respond within the time required by the PD spec. The 'GSrC' Task shall be considered successful if: • The Get_Source_Cap message is sent, GoodCRC'ed and a Source Capabilities message is received and processed.
Side Effects	When the 'GSrC' Task completes successfully the *RX_SOURCE_CAPS* register (0x30) will have been updated.
Additional Information	None

5.2.2 'GPPI' - PD Get Port Partner Information

The 'GPPI' Task can be used to cause the PD controller to issue these types of USB PD Get messages:

- Get_Source_Cap_Extended (control message)
- Get_Status (Control message)
- Get_Battery_Status (Extended message)
- Get_Battery_Cap (Extended message)
- Get_Manufacturer_Info (Extended message)

The PD controller does not have dedicated registers to store the response to these messages. The host must get that response from the DATAX register associated with this Task.

The host must NOT use 'GPPI' to send Get_Source_Capabilities messages, because the USB PD spec requires specific actions be taken by the PD controller any time those messages are received. While executing the 'GPPI' Task, the PD controller does not parse the returned message to carry out those checks. Instead, the host must use 'GSrC' to send Get_Source_Capabilities messages.

Table 5-3. 'GPPI' - Send a USB PD Get* Message

Description	**The 'GPPI' Task instructs PD Controller to issue a specific USB PD message to the Port Partner at the first opportunity while maintaining policy engine compliance.**			
INPUT DATAX	**Bit**	**Name**	**Description**	
	15	Reserved		
	14:13	FrameType	00b	SOP
			01b	SOP'
			10b	SOP''
			11b	Reserved
	12:8	NumBytes		
	7	Reserved		
	6:5	MessageCategory	00b	Control message (no payload)
			01b	Data message (requires payload)
			10b	Extended message (requires payload)
			11b	Reserved
	4:0	MessageType	This field must be the MessageType as defined in the USB PD specification. It specifies the Type of message the PD controller will send.	
OUTPUT DATAX	Byte 1: Standard Task Return Code. See also Table 5-1.			
Task Completion	The 'GPPI' Task completes either when the appropriate message is received or the Task otherwise fails.The 'GPPI' Task shall be considered rejected if: • Sending the requested message can violate the USB PD spec. • The PortPartner replies with a Reject or Not_Supported message. The 'GPPI' Task shall be considered timed-out if: • The requested message is sent, GoodCRC'ed and the recipient (Port Partner or Cable Plug) fails to respond within the time required by the PD spec. • A PD Hard Reset or a disconnection happens before the Task completes. The 'GPPI' Task shall be considered successful if: • The requested message is sent, GoodCRC'ed and an appropriate response is received and processed. The 'GPPI' Task shall be aborted when the Rx Buffer is locked. The Rx Buffer is locked after data from a receive message is placed in the DATAX register. The Rx Buffer is unlocked after disconnect and by the 'MBRd' Task.			
Side Effects	If the PD controller is in the sink power role and it reads Rp = SinkTxNG, it will wait until Rp = SinkTxOK before initiating the atomic message sequence requested by this 'GPPI' Task. This can cause an non-deterministic delay in completing the Task.			

Table 5-3. 'GPPI' - Send a USB PD Get* Message (continued)

Description	**The 'GPPI' Task instructs PD Controller to issue a specific USB PD message to the Port Partner at the first opportunity while maintaining policy engine compliance.**
Additional Information	The PD controller will continue trying to execute this Task until it times out or aborts as described above. Some scenarios where this can happen are: • The PD controller with a sink power role (that is PresentRole = Sink) is required to wait for Rp = SinkTxOK before initiating an Atomic Message Sequence. The PD controller will continue waiting for Rp = SinkTxOK until it is able to send the appropriate message required for this 'GPPI' Task. While executing the 'GPPI' Task, the PD controller uses the same shared buffer that is used to store other extended messages. Therefore, the host must not use the 'GPPI' Task when any other atomic message sequence is ongoing. To read the PD response received as a result of issuing the 'GPPI' Task after it is completed, the host must use the 'MBRd' 4CC command. The 'MBRd' Task must also be used to unlock the Rx Buffer for other incoming message.

5.2.3 'MBRd' - Message Buffer Read

Table 5-4. 'MBRd' - Read from PD message buffer.

Description	**The MBRd Task instructs the PD Controller to read data from the extended message buffer previously received from the Port Partner.**			
INPUT DATAX	**Bit**	**Name**	**Description**	
	23	Reserved	Reserved (Write as 0).	
	22	UnlockRxBuffer	This input controls whether or not the PD controller unlocks its internal buffer after this Task is completed. It is recommended to unlock the internal buffer as soon as possible to make room for other incoming messages. It is important that the host only set this bit to 1 after it has received an alert that the Rx Buffer is locked.	
			0b	Do not clear the internal buffer, another 'MBRd' Task can be used later.
			1b	Clear the internal buffer after this Task completes and the requested data is in the DATAx register.
	21:16	DataSize	Number of data bytes to be read in from the message buffer. Up to 62 bytes can be read at after.	
	15:0	BuffOffset	Buffer Offset. Values 0 to 259 are possible.	
OUTPUT DATAX	**Bit**	**Name**	**Description**	
	511:16	DataByte1	First Byte of data read at BuffOffset.	
	15:0	MessageSize	Size of message in bytes.	
Task Completion	The MBRd Task completes after buffer data of DataSize at BuffOffset has been read from the message buffer.			
Side Effects	None			
Additional Information	This Task is required for the host to obtain the information from the response due to the usage of the 'GPPI' Task. The PD controller has a single buffer per port that is shared for these messages.			

5.3 System Tasks

5.3.1 'ANeg' - Auto Negotiate Sink Update

Table 5-5. 'ANeg' - Re-evaluate the auto-negotiate sink register

Description	**The 'ANeg' Task instructs PD Controller to re-evaluate the *Auto Negotiate Sink* register (0x37). If the re-evaluation produces a different RDO than the Active Contract RDO then a new Request message is sent.**
INPUT DATAX	None
OUTPUT DATAX	Byte 1: Standard Task Return Code. See also Table 5-1.
Task Completion	The 'ANeg' Task completes after the new RDO is calculated and PD Controller either decides to send a new Request message (and that message is sent and the GoodCRC received) or determines that no Request is necessary.
Side Effects	The side effects include updating the sink capabilities, a new PD contract negotiation, and updates to the associated registers.
Additional Information	None

5.3.2 'DBfg' - Clear Dead Battery Flag

Table 5-6. 'DBfg' - Clear Dead Battery Flag

Description	The 'DBfg' Task is used to clear the dead battery flag. This Task does not disable the PP_EXT input switch.
INPUT DATAX	None
OUTPUT DATAX	None
Task Completion	The 'DBfg' Task completes after the effects of clearing the Dead Battery Flag are complete.
Side Effects	The Dead Battery Flag controls the power that drives the device logic and will have side effect in the system if the device is bus powered only. PD Controller 's power input is forced to VBUS until the Dead Battery Flag is cleared, so executing this Task will change PD Controller 's power input.
Additional Information	None

There are several limitations placed on the PD controller while the Dead-Battery Flag is asserted.

- VBUS is selected as the main supply for the PD controller, even if the 3.3 V input is present.

6 User Reference

6.1 AUTO_NEGOTIATE_SINK Register

In general, writing to AUTO_NEGOTIATE_SINK register while a sink contract is in place does not cause an automatic renegotiation, and changes take effect the next time a contract is negotiated. The ANeg command forces a re-evaluation of this register and a new Request message is issued if appropriate.

If the first four bytes of this register are written as zero, then the PD controller always requests a 5V Fixed Suppy contract at 100mA.

The following is a high-level summary of how this register drives the PDO selection when PPS is disabled or no matching APDO is found.

- Parse the received PDOs in the register RX_SOURCE_CAPS. Discard any PDO whose voltage range is below ANMinVoltage or above ANMaxVoltage.
- Calculate the PDO power for each received PDO (RX_SOURCE_CAPS.SourcePdoX). Rank all PDOs according to the PDO power.
  - PDO Power = Voltage × MaximumCurrent (Fixed Supply)
  - PDO Power = MinimumVoltage × MaximumCurrent (Variable Supply)
  - PDO Power = MaximumPower (Battery Supply)
- The PDO with maximum PDO Power that also passes the voltage check is selected. In case there are multiple PDOs that pass the voltage check and have the same maximum PDO Power, tie breakers are applied as described below:
  - A Fixed supply type is preferred, and Variable supply type is preferred over Battery supply type.
  - If the PDOs being compared have the same supply type, then ANRDOPriority specifies how to break the tie.

6.1.1 AUTO_NEGOTIATE_SINK Usage Example 1

When attached to a 36W source the PD controller has RX_SOURCE_CAPS:

- PDO1: 5V at 3A
- PDO2: 9V at 3A
- PDO3: 15V at 2.4A
- PDO4: 20V at 1.8A

The PD controller has TX_SINK_CAPS set as:

- PDO1: 5V at 3A (fixed)
- PDO2: 20V at 3A (fixed)

The PD controller has AUTO_NEGOTIATE_SINK set as:

- AUTO_NEGOTIATE_SINK = 0
- AUTO_NEGOTIATE_SINK.ANSinkCapMismatchPower = 240d (60W)
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMinPower = 1
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMinVoltage = 1
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMaxVoltage = 1
- AUTO_NEGOTIATE_SINK.NoCapabilityMismatch = x (see table below)
- AUTO_NEGOTIATE_SINK.ANRDOPriority = y (see table below)

The settings give the following results:

- ANSinkMinRequiredPower computed as 60 W
- ANMaxVoltage computed as 20V
- ANMinVoltage computed as 4.75V

Table 6-1. AUTO_NEGOTIATE_SINK Usage Example 1

AUTO_NEGOTIATE_SINK		ACTIVE_CONTRACT_RDO			
NoCapabilityMismatch	ANRDOPriority	OperatingX	MinMaxOperatingX	ObjectPosition	Capability Mismatch
0	0	1.8A	3.0A	4	1
1	0	1.8A	1.8A	4	0

Table 6-1. AUTO_NEGOTIATE_SINK Usage Example 1 (continued)

AUTO_NEGOTIATE_SINK		ACTIVE_CONTRACT_RDO			
NoCapabilityMismatch	ANRDOPriority	OperatingX	MinMaxOperatingX	ObjectPosition	Capability Mismatch
1	1	2.4A	2.4A	3	0

6.1.2 AUTO_NEGOTIATE_SINK Usage Example 2

When attached to a 36W source the PD controller has RX_SOURCE_CAPS:

- PDO1: 5V at 3A
- PDO2: 9V at 3A
- PDO3: 15V at 2.4A

The PD controller has TX_SINK_CAPS set as:

- PDO1: 5V at 0.1A (fixed)
- PDO2: 20V at 3A (fixed)

The PD controller has AUTO_NEGOTIATE_SINK set as:

- AUTO_NEGOTIATE_SINK = 0
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMinPower = 1
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMinVoltage = 0
- AUTO_NEGOTIATE_SINK.ANMinVoltage = 20V
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMaxVoltage = 1
- AUTO_NEGOTIATE_SINK.NoCapabilityMismatch = x (see table below)
- AUTO_NEGOTIATE_SINK.ANRDOPriority = 0

The settings give the results in the table below. Note that ANMaxVoltage computed as 20V, but it does not affect the result. Because the ANMinVoltage was set to 20V, and the source is not offering 20V none of the source PDOs fulfill the sink requirements. Even though ANSinkCapMismatchPower=0 in this example, because the voltages offered are insufficient, the capability mismatch bit can still be set.

Table 6-2. AUTO_NEGOTIATE_SINK Usage Example 2

AUTO_NEGOTIATE_SINK	ACTIVE_CONTRACT_RDO			
NoCapabilityMismatch	OperatingX	MinMaxOperatingX	ObjectPosition	Capability Mismatch
0	3.0A	3.0A	1	1
1	3.0A	3.0A	1	0

6.1.3 AUTO_NEGOTIATE_SINK Usage Example 3

When attached to a 45W source the PD controller has RX_SOURCE_CAPS:

- PDO1: 5V at 3A
- PDO2: 9V at 3A
- PDO3: 15V at 3A
- PDO4: 20V at 2.25A

The PD controller has TX_SINK_CAPS set as:

- PDO1: 5V at 3A (fixed)
- PDO2: 20V at 2.25A (fixed)

The PD controller has AUTO_NEGOTIATE_SINK set as:

- AUTO_NEGOTIATE_SINK = 0
- AUTO_NEGOTIATE_SINK.ANSinkCapMismatchPower = 180d (45W)
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMinPower = 1
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMinVoltage = 1
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMaxVoltage = 1
- AUTO_NEGOTIATE_SINK.NoCapabilityMismatch = 0
- AUTO_NEGOTIATE_SINK.ANRDOPriority = y (see table below)

The settings give the following results:

- ANSinkMinRequiredPower computed as 45W
- ANMaxVoltage computed as 20V
- ANMinVoltage computed as 4.75V

Table 6-3. AUTO_NEGOTIATE_SINK Usage Example 3.

AUTO_NEGOTIATE_SINK	ACTIVE_CONTRACT_RDO			
ANRDOPriority	OperatingX	MinMaxOperatingX	ObjectPosition	Capability Mismatch
0	2.25A	2.25A	4	0
1	3.0A	3.0A	3	0

6.1.4 AUTO_NEGOTIATE_SINK Usage Example 4

When attached to a 100W source the PD controller has RX_SOURCE_CAPS:

- PDO1: 5V at 3A
- PDO2: 9V at 3A
- PDO3: 15V at 3A
- PDO4: 20V at 5A

The PD controller has TX_SINK_CAPS set as:

- PDO1: 5V at 3A (fixed)
- PDO2: 20V at 5A (fixed)

The PD controller has AUTO_NEGOTIATE_SINK set as:

- AUTO_NEGOTIATE_SINK = 0
- AUTO_NEGOTIATE_SINK.ANSinkCapMismatchPower = 240d (60W)
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMinPower = 1
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMinVoltage = 1
- AUTO_NEGOTIATE_SINK.AutoComputeSinkMaxVoltage = 1
- AUTO_NEGOTIATE_SINK.NoCapabilityMismatch = 0
- AUTO_NEGOTIATE_SINK.ANRDOPriority = y (see table below)

The settings give the following results:

- ANSinkMinRequiredPower computed as 100W
- ANMaxVoltage computed as 20V
- ANMinVoltage computed as 4.75V

Table 6-4. AUTO_NEGOTIATE_SINK Usage Example 3.

AUTO_NEGOTIATE_SINK	ACTIVE_CONTRACT_RDO			
ANRDOPriority	OperatingX	MinMaxOperatingX	ObjectPosition	Capability Mismatch
0	5A	5A	4	0
1	5A	5A	4	0

7 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

DATE	REVISION	NOTES
July 2026	*	Initial Release

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale, TI’s General Quality Guidelines, or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Last updated 10/2025