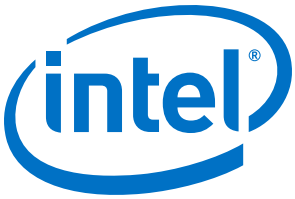


Mailbox Client Intel® Stratix® 10 FPGA IP Core User Guide

Subscribe
Send Feedback

UG-20087 | 2019.04.19
Latest document on the web: PDF | HTML

Contents

1.1. Feature Description....3
1.2. Commands and Responses....4
1.2.2. Error Code Responses....9
1.4. Mailbox Client Intel Stratix 10 FPGA IP Core Avalon-Memory Map....10
1.4.1. Interrupt Enable Register....11
1.4.2. Interrupt Status Register....11
1.5. Using the Mailbox Client Intel Stratix 10 FPGA IP Core....12
1.6. Mailbox Client Intel Stratix 10 FPGA IP Core Use Case Examples....16
1.7. Document Revision History for the Mailbox Client Intel Stratix 10 FPGA IP Core User Guide....16

1. Mailbox Client Intel® Stratix® 10 FPGA IP Core User Guide

The Mailbox Client Intel® Stratix® 10 FPGA IP core converts the Avalon®-ST interface to Avalon-MM interface for FPGA mailbox clients to communicate with the secure device manager (SDM) in Intel Stratix 10 devices. This IP core also enables you to create mailbox command packet, send mailbox commands, and receive status.

With the capability to communicate with SDM, this IP core allows you to perform various operations such as accessing the QSPI flash, accessing the FPGA temperature sensor or voltage sensor reading, reading device CHIP ID value and etc. For details on how to perform these operations, refer to the *Operation Commands* section of this user guide.

Related Information

- Avalon Interface Specifications
- Secure Device Manager in Intel Stratix 10 Devices
- Operation Commands on page 5

1.1. Feature Description

The Mailbox Client Intel Stratix 10 FPGA IP core features commands and responses for communication with the SDM.

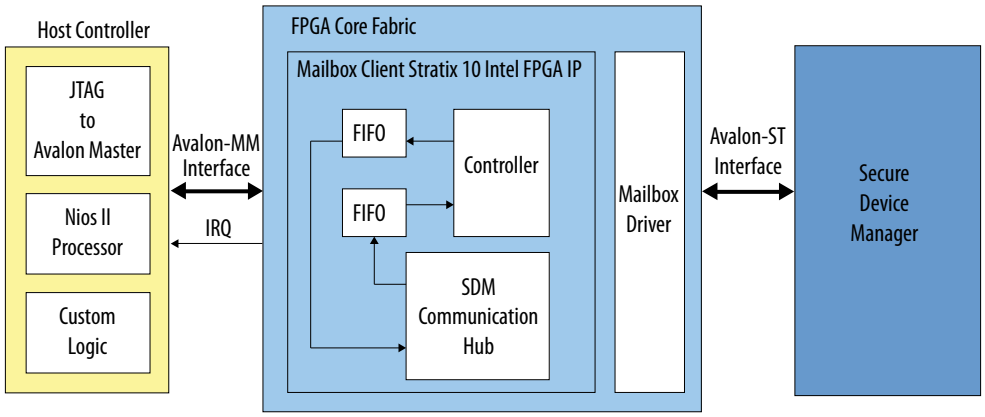
The Mailbox Client Intel Stratix 10 FPGA IP core supports:

- Command and response access.
- Depth-adjustable FIFO to buffer command and response packet.

  Note: The maximum FIFO depth that you can set in the IP for command and response FIFO is 1024.
- Configurable interrupt source:
  - Indication that command FIFO is not full and client can send more commands.
  - Indication that there is valid response in FIFO and client can begin to read out the response.

Figure 1. Mailbox Client Intel Stratix 10 FPGA IP Core and System Block Diagram

Note: Refer to the *Secure Device Manager* topic of the *Intel Stratix 10 Configuration User Guide* for the SDM block diagram showing the connection of the Mailbox Client Intel Stratix 10 FPGA IP core with the mailbox block in SDM.



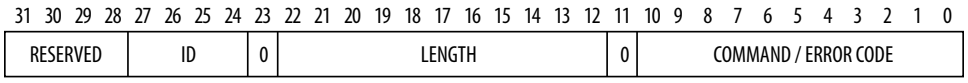
Related Information

Secure Device Manager of the Intel Stratix 10 Devices

1.2. Commands and Responses

The remote system update host communicates with the SDM using command and response packets via the Mailbox Client Intel Stratix 10 FPGA IP.

Figure 2. Command and Response Header Format



Note: The LENGTH field in the command header must match the command length of corresponding command.

The following table describes the fields of the header command.

Table 1. Mailbox Client Intel Stratix 10 FPGA IP Command and Response Header Description

Header	Bit	Description
Reserved	[31:28]	Reserved.
ID	[27:24]	The command ID. The response header returns the ID specified in the command header. Set different IDs in each command to match responses with commands.
0	[23]	Reserved.

continued...

Header	Bit	Description
`Length`	[22:12]	Number of words of arguments following the header. The IP responds with an error if a wrong number of words of arguments is entered for a given command.
Reserved	[11]	Reserved. Must be set to 0.
`Command Code/Error Code`	[10:0]	`Command Code` specifies the command. The `Error Code` indicates whether the command succeeded or failed. In the command header, these bits represent command code. In the response header, these bits represent error code.

3. Operation Commands

Table 2. Command List and Description

Command	Code (Hex)	Command Length (1)	Response Length (1)	Description
`NOOP`	0	0	0	Sends an OK status response.
`GET_IDCODE`	10	0	1	The response contains one argument which is the JTAG IDCODE for the device
`GET_CHIPID`	12	0	2	The response contains 64-bit CHIPID value with the least significant word first.
`GET_USERCODE`	13	0	1	The response contains one argument which is the 32-bit JTAG USERCODE provided to the device by the configuration bitstream.
`GET_VOLTAGE`	18	1	1	Command has a single argument which is a bitmask of which channels to read. Bit 0 is set to read channel 0, bit 1 to read channel 1 and so on. The response contains one word argument for each bit set in the bitmask. Each return value is a 32-bit value. The voltage returned is an unsigned fixed point number with 16 bits below the binary point. Example: A voltage of 0.75V returns 0x0000C000.
`GET_TEMPERATURE`	19	1	1	Command has a single argument which is a bitmask indicating which temperature sensors to read. The response contains one word for each channel (2) temperature requested. The temperature returned as a signed fixed value with 8 bits below the binary point. • Channel 0: Samples the temperature value from the core fabric. • Channels 1 to 6: Samples the temperature value from the specified transceiver tile. • Channels 7 to 8: Samples the temperature value from the high-bandwith DRAM memory (HBM2) stacks. Example: A Temperature of 10°C returns 0x00000A00 and a of temperature -1.5°C returns 0xFFFFFE80.
`RSU_IMAGE_UPDATE`	5C	2	0	Triggers reconfiguration from the data source selected by MSEL setting for initial configuration.

continued...

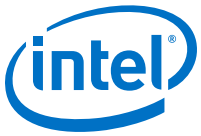
(1) This number does not include the command and response header.

(2) The availability of each transceiver tile varies among devices. For the temperature sensor channel numbers, refer to the related information.

<table>
<tr><th>Command</th><th>Code (Hex)</th><th>Command Length (1)</th><th>Response Length (1)</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td>This command takes an optional 64-bit argument to specify the reconfiguration data address in the flash. If the argument is not provided then its value is assumed to be 0.
• Bit [63:32]: Reserved (write as 0).
• Bit [31:0]: The start address of an application image.
Returns non-zero response if the device is already processing a configuration.
Note: You can use this command to check the configuration status during configuration and after it has completed.</td></tr>
<tr><td>CONFIG_STATUS</td><td>4</td><td>0</td><td>6</td><td>Reports the status of the last reconfiguration. You can use this command to check the configuration status during and after configuration. The response contains the following:
<table>
<tr><th>Word</th><th>Summary</th><th>Description</th></tr>
<tr><td>0</td><td>State</td><td>Describes the most recent configuration related error. 0 when no configuration errors..
The error field has 2 fields:
• Upper 16 bits: Major error code.
• Lower 16 bits: Minor errors that do not contain meaningful data.
Here are valid values for major error codes:
<table>
<tr><th>Major Error Code</th><th>Description</th></tr>
<tr><td>0xF001</td><td>BITSTREAM_ERROR</td></tr>
<tr><td>0xF002</td><td>HARDWARE_ACCESS_FAILURE</td></tr>
<tr><td>0xF003</td><td>BITSTREAM_CORRUPTION</td></tr>
<tr><td>0xF004</td><td>INTERNAL_ERROR</td></tr>
<tr><td>0xF005</td><td>DEVICE_ERROR</td></tr>
<tr><td>0xF006</td><td>HPS_WATCHDOG_TIMEOUT</td></tr>
<tr><td>0xF007</td><td>INTERNAL_UNKNOWN_ERROR</td></tr>
</table></td></tr>
<tr><td>1</td><td>Version</td><td>0 for this version.</td></tr>
<tr><td>2</td><td>Pin status</td><td>• Bit [31]: Current nSTATUS output value (active low).
• Bit [30]: Detected nCONFIG input value (active low).
• Bit [29:3]: Reserved.
• Bit [2:0]: The MSEL value at power up.</td></tr>
<tr><td>3</td><td>Soft function status</td><td>Contains the value of each of the soft functions, regardless if the function has been assigned to an SDM pin.</td></tr>
</table></td></tr>
</table>

continued...

(1) This number does not include the command and response header.



<table>
<tr><th>Command</th><th>Code (Hex)</th><th>Command Length (1)</th><th>Response Length (1)</th><th colspan="3">Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>• Bit [31:4]: Reserved
• Bit [3]: SEU_ERROR
• Bit [2]: CVP_DONE
• Bit [1]: INIT_DONE
• Bit [0]: CONF_DONE</td></tr>
<tr><td></td><td></td><td></td><td></td><td>4</td><td>Error location</td><td>Contains the error location. Returns 0 for no error.</td></tr>
<tr><td></td><td></td><td></td><td></td><td>5</td><td>Error details</td><td>Contains the error details. Returns 0 for no error.</td></tr>
<tr><td>RSU_STATUS</td><td>5B</td><td>0</td><td>8</td><td colspan="3">Reports the current remote system upgrade status. This command returns the following responses:</td></tr>
<tr><td></td><td></td><td></td><td></td><td>Word</td><td>Summary</td><td>Description</td></tr>
<tr><td></td><td></td><td></td><td></td><td>0-1</td><td>Current image</td><td>Flash offset of the currently running application image.</td></tr>
<tr><td></td><td></td><td></td><td></td><td>2-3</td><td>Last failing image</td><td>Flash offset of the last failing application image. The value of all 1s indicates no failing images. If no failing images, the following words do not contain meaningful data.</td></tr>
<tr><td></td><td></td><td></td><td></td><td>4</td><td>State</td><td>Failure code of the last failing image.
The error field has two parts:
• Upper 16 bits: Major error code.
• Lower 16 bits: Minor errors that do not contain meaningful data.
The following major error codes are defined:
<table><tr><th>Major Error Code</th><th>Description</th></tr><tr><td>0xF001</td><td>BITSTREAM_ERROR</td></tr><tr><td>0xF002</td><td>HARDWARE_ACCESS_FAILURE</td></tr><tr><td>0xF003</td><td>BITSTREAM_CORRUPTION</td></tr><tr><td>0xF004</td><td>INTERNAL_ERROR</td></tr><tr><td>0xF005</td><td>DEVICE_ERROR</td></tr><tr><td>0xF006</td><td>HPS_WATCHDOG_TIMEOUT</td></tr><tr><td>0xF007</td><td>INTERNAL_UNKNOWN_ERROR</td></tr></table></td></tr>
<tr><td></td><td></td><td></td><td></td><td>5</td><td>Version</td><td>Contains the value of each of the soft functions, whether that function is on an SDM pin.</td></tr>
<tr><td></td><td></td><td></td><td></td><td>6</td><td>Error location</td><td>Contains the error location of the last failing image. Returns 0 for no error.</td></tr>
<tr><td></td><td></td><td></td><td></td><td>7</td><td>Error details</td><td>Contains the error details of the last failing image. Returns 0 for no error.</td></tr>
<tr><td colspan="7">***continued...***</td></tr>
</table>

(1) This number does not include the command and response header.

Command	Code (Hex)	Command Length (1)	Response Length (1)	Description
QSPI_OPEN	32	0	1	Requests exclusive access to the quad-serial peripheral interface (QSPI). If the SDM accepts the request (if the QSPI is not already in use or the SDM is not in the process of a device configuration), it returns the OK response, else it returns the error response. *Note:* The exclusive access is granted only to the client using this mailbox. Other clients are not be able to access the QSPI until it is closed by this client.
QSPI_CLOSE	33	0	1	Closes the exclusive access to the QSPI interface.
QSPI_SET_CS	34	1	1	Select which of the attached QSPI device via the chip select lines. Takes on one word argument as described below: • Bit [31:28]: Flash device to be selected. Bit setting 0000 is used to select flash that attached to nCSO[0]. • Bit [27:0]: Reserved (write as 0).
QSPI_READ	3A	2	N	Reads the attached QSPI device and takes two parameters: • The flash address offset from where you want to start reading from the QSPI device (one word). The read address must be word aligned. • Number of words to read (one word). A successful response returns an OK response code followed by the data read from the QSPI device. A failure response is either: • Returns an error code. • Returns OK but part of the data read from QSPI device is incorrect. For example, you may perform multiple QSPI_READ operations. Part of the data will be returned if the first read succeeds but the subsequent read fails. *Note:* The maximum transfer size is limited to 4K bytes and cannot be called while a configuration is in progress. Example: The FPGA device returns error code 0x1 if a non word-aligned write address is sent.
QSPI_WRITE	39	2+N	0	Writes data on the attached QSPI device and takes three parameters: • The flash address offset from where you want the command to start writing to the QSPI device (one word). The write address must be word aligned. • The number of words to write (one word). • The data to be written (one or more words). A successful write returns an OK response code. The client may need to issue QSPI_ERASE command before issuing this command to prepare the memory for writing. *Note:* The maximum transfer size is limited to 4K bytes and cannot be called while a configuration is in progress. Example: The FPGA device returns error code 0x3FF if a non word-aligned read address is sent.
QSPI_ERASE	38	2	0	Erases sector on the attached QSPI device and takes two parameters: • The flash address offset within the device to start erasing from (one word). The address must be the start address of a sector within the flash memory and therefore the address must be 64K bytes aligned. If a non-64K bytes aligned address is specified, an error will be responded. • The number of words to erase. The erase size can be specified in multiple of 4000 (hexadecimal) words.

continued...

(1) This number does not include the command and response header.

Command	Code (Hex)	Command Length (1)	Response Length (1)	Description
				A successful erase returns an OK response code.
QSPI_READ_DEVICE_REG	35	2	N	Reads registers from the attached QSPI device and takes two parameters. • The opcode for the read command. • The number of bytes to read (the maximum size is 8 bytes). A successful read returns an OK response code followed by the data read from the device. If the data is not an exact multiple of 4 bytes then it is padded with 0 bytes until the next word boundary.
QSPI_WRITE_DEVICE_REG	36	2+N	0	Writes to registers on the attached QSPI and takes three arguments: • The opcode for the write command. • The number of bytes to write (the maximum size is 8 bytes). • The data to write (maximum 2 words, padded with 0 to word boundary). A successful write returns an OK response code.
QSPI_SEND_DEVICE_OP	37	1	0	Sends a command opcode to the QSPI and takes one argument: • The opcode to send the attached QSPI device. A successful command returns an OK response code.

Related Information

Intel Stratix 10 Analog to Digital Converter User Guide
Provides more information about the temperature sensor channel numbers of the Intel Stratix 10 temperature sensing diodes (TSDs).

1.2.2. Error Code Responses

Table 3. Mailbox Client Intel Stratix 10 FPGA IP Error Code Responses and Description

Value (Hex)	Error Code Response	Description
0	OK	Indicates that the command completed successfully. Depending on the command delivered to the Mailbox Client, the response error code may not be sufficient to ensure that the operation completed successfully.
1	INVALID_COMMAND	Indicates that the command is in an incorrect format.
2	UNKNOWN_BR	Indicates that the command code is not understood.
3	UNKNOWN	Indicates that the command code is not understood by the currently loaded firmware.
100	NOT_CONFIGURED	Indicates that the device is not configured.
1FF	ALT_SDM_MBOX_RESP_DEVICE_BUSY	Indicates that the device is busy.
2FF	ALT_SDM_MBOX_RESP_NO_VALID_RESP_AVAILABLE	Indicates that there is no valid response available.
3FF	ALT_SDM_MBOX_RESP_ERROR	General Error

(1) This number does not include the command and response header.

1. Mailbox Client Intel Stratix 10 FPGA IP Core Signals

Table 4. Mailbox Client Intel Stratix 10 FPGA IP Core Signal Description

Signal Role	Width	Direction	Description
Avalon-MM Interface Signals			
avmm_address	4	Input	Avalon-MM address.
avmm_write	1	Input	Avalon-MM write request.
avmm_read	1	Input	Avalon-MM read request.
avmm_writedata	32	Input	Avalon-MM writedata bus.
avmm_readdata	32	Output	Avalon-MM readdata bus.
avmm_readdatavalid	1	Output	Avalon-MM readdata valid.
Clock and Reset			
clk	1	Input	Input clock to clock the IP core. The maximum frequency supported for this clock is 250 MHz.
reset	1	Input	Reset that resets the IP core. To reset the IP core, assert the reset signal high for at least 2 clk cycles. *Note:* For guidelines to initiate the IP core, Intel strongly recommends that you refer to *Intel Stratix 10 Reset Release IP* section in the *Intel Stratix 10 Configuration User Guide*.
irq	1	Output	Interrupt signal reflects update to interrupt status register with corresponding interrupt enable register.

Related Information

Intel Stratix 10 Configuration User Guide
More information on Intel Stratix 10 Reset Release IP.

1.4. Mailbox Client Intel Stratix 10 FPGA IP Core Avalon-Memory Map

Table 5. Mailbox Client Intel Stratix 10 FPGA IP Core Avalon-Memory Map

Offset (word)	R/W	31	1	0
Base + 0	W	Command		
Base + 1	W	Command last word (eop)		
Base + 2	R	Command FIFO empty space		
Base + 3	N/A	Reserved		
Base + 4	N/A	Reserved		
Base + 5	R	Response data		

continued...

Offset (word)	R/W	31	1	0
Base + 6	R	Response FIFO fill level	EOP	SOP
Base + 7	R/W	Interrupt enable register (IER)		
Base + 8	R	Interrupt status register (ISR)		

1.4.1. Interrupt Enable Register

Table 6. Interrupt Enable Register

Having the enable bit cleared disregards the corresponding interrupt status bit from causing interrupt output assertion (IRQ).

Note: These enable bits does not prevent the value of interrupt status bit from showing up in ISR, it only prevents the interrupt status bit from causing interrupt output assertion.

Bit	Fields	Access	Default Value	Description
31:2	Reserved			
1	EN_CMD_FIFO_NOT_FULL	R/W	0x0	The enable bit interrupt of command FIFO is not full. • 1: Enable the corresponding interrupt • 0: Disable the corresponding interrupt
0	EN_DATA_VALID	R/W	0x0	The enable bit of data valid in response FIFO. • 1: Enable the corresponding interrupt • 0: Disable the corresponding interrupt

1.4.2. Interrupt Status Register

Table 7. Interrupt Status Register

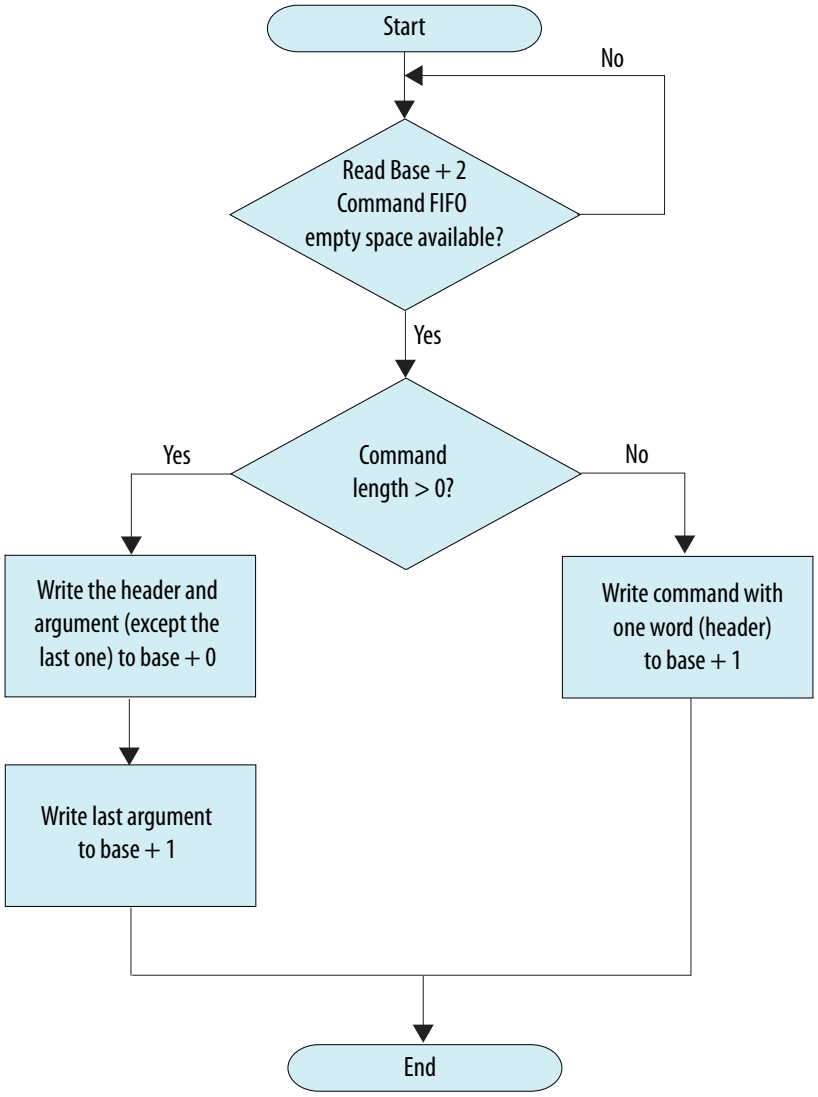
Bit	Fields	Access	Default Value	Description
31:2	Reserved			
1	CMD_FIFO_NOT_FULL	R	0x0	Command FIFO is not full interrupt. (3) • 1: Indicates command FIFO is not full, the client can send more data into it. • 0: Indicates FIFO is full.
0	DATA_VALID	R	0x0	Data valid interrupt. (3) • 1: Indicates data exist in FIFO, master can read it out. • 0: Indicates the FIFO is empty.

(3) This bit is cleared by operations on the FIFO. You do not need to clear this bit manually.

1.5. Using the Mailbox Client Intel Stratix 10 FPGA IP Core

Writing Command Packet

Figure 3. Flow Chart for Writing Command Packet



When you send a command to the SDM, write the command word into command register. The last word must be written to a `command_last_word` register to stay in sync with the hardware. If the command has 0 arguments, the header should be written to the `command_last_word` register.

Command with one word (header only): Write to base + 1.

Command with more than one word (header + argument):

1. Write the header and argument (except the last one) to base + 0.
2. Write last argument to base + 1.

You can read from base + 2, which shows the remaining available free space in the FIFO for command. The FIFO fills up when the SDM is busy. The IP requires 3 clock cycles to update the command FIFO empty space value. You can only start reading the value after 3 clock cycles after writing the command to the IP.

The behavior of the IP is undefined if you write to base + 0 and base + 1 while the FIFO is full. The write data is discarded.

Unexpected or undefined behavior of the IP may occur if you send more than the commands required. For example, the following commands are sent for reading the Chip ID value:

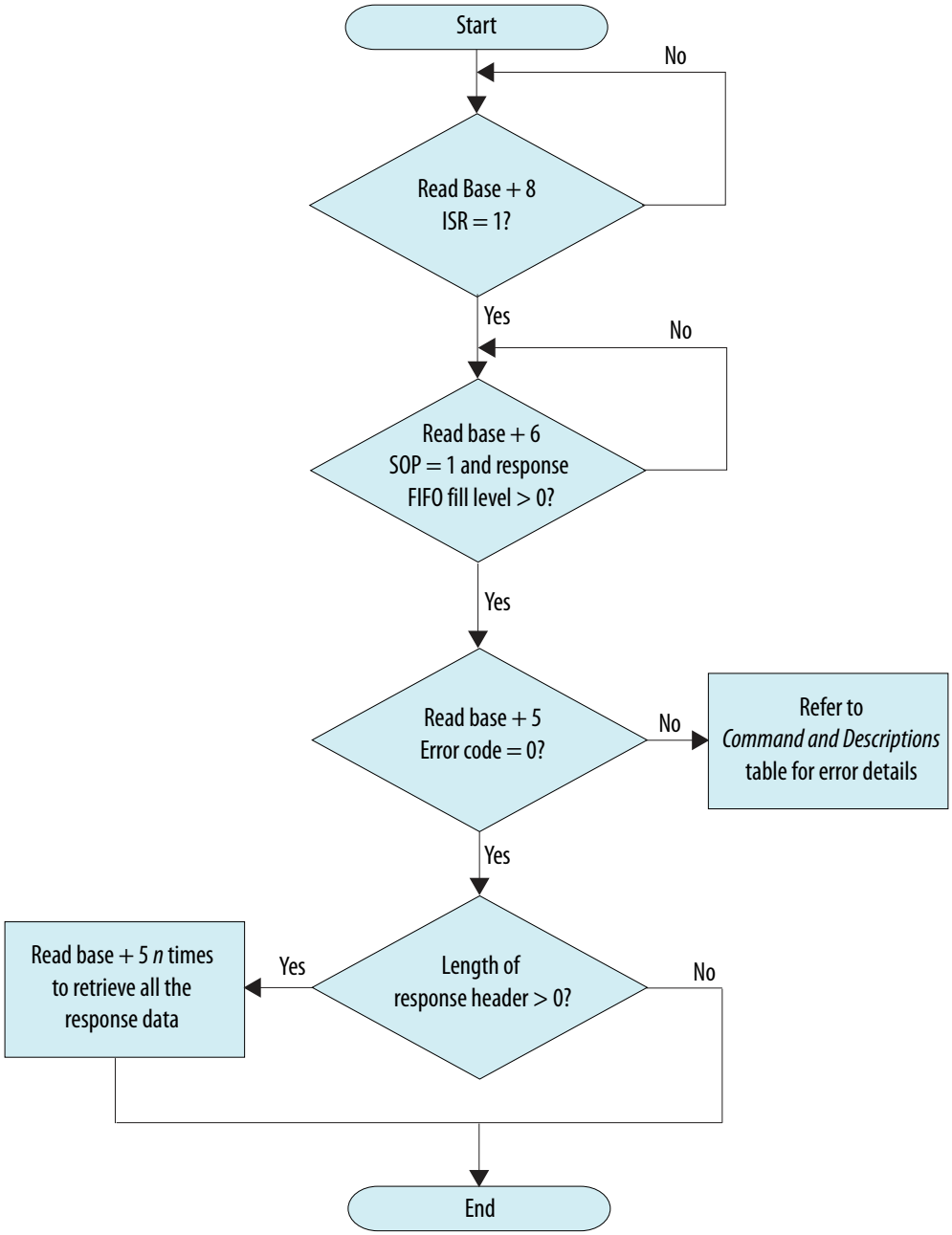
- Write the command header to base + 0.
- Write again the command header to base + 1.

In the above scenario, the IP core expects 3 responses but the SDM will only return one response, which is an error response code.

You must send commands in the correct order to the command or command last word register, as described in the Writing Command Packet on page 12, to prevent loss of services to all mailbox clients, including standalone IP cores, such as the Temperature Sensor Intel Stratix 10 FPGA IP, Voltage Sensor Intel Stratix 10 FPGA IP, Chip ID Intel Stratix 10 FPGA IP, and etc.

Reading Response Packet

Figure 4. Flow Chart for Reading Response Packet



1. Read base + 8 to check if bit 0 of ISR = 1.

   You can poll the ISR continuously until bit 0 of ISR = 1 after the writing a command packet is completed.
2. Read base + 6 to check the SOP, EOP, and fill level of response FIFO.

For reading multiple words:

- Check if SOP = 1 and EOP = 0. This indicates the response has multiple words.
- Check fill level of response FIFO to ensure that it is not zero.
- For example, if you perform a QSPI_READ operation to read 10 words from QSPI flash, a return value of `0x0000002d` indicates that 11 words are returned by SDM to fill up the response FIFO. The 11 words consist of a response header word and 10 words of data.

For reading single word:

- Check if SOP = 1 and EOP = 1.
- Check fill level of response FIFO to ensure that it is not zero.
- For example, a return value of `0x00000007` indicates that a word is returned by SDM to fill up the response FIFO where the word represents the start of packet and also is the end of packet.

3. Read base + 5 to read response header. The *length* value contains the number (eg: n) of argument a response has. Proceed to step 4 if the responded error code = 0. A non zero error code response indicates that the command writing to the mailbox is not completed successfully. Refer to Table 3 on page 9 for more information.
4. Read base + 5 n times to retrieve all the response data. While continuously read back the response data, you must also continuously poll base + 6 to check response FIFO fill level and ensure it is not zero before read the response data, EOP = 1 indicates that this is the last response data for the response packet.

   Note: Undefined value will be returned if the response FIFO is empty. You must wait for ISR to ensure that valid data is available and check the response FIFO fill level for reading the response data.

   Ensure that you read or flush out the content in the response FIFO before issuing a new command to the mailbox. Continuously sending commands without reading back the valid data from response FIFO will gradually fill up the response FIFO. This will result the SDM to freeze due to overflowing responses.

   Device reconfiguration is required to recover the SDM from freezing, which is supported only in Intel Quartus® Prime software version 19.1 onwards. Prior to the Intel Quartus Prime software version 19.1, power cycle the device to recover the SDM from freezing.

Restrictions

1. You can only issue one request and read back the responses before issuing a new request to the Mailbox Client IP. Wait for 10 ms interval if you want to send a back-to-back command to the SDM mailbox.
2. You should not to instantiate more than 6 mailbox clients in your design. If you need to use more than 6 mailbox clients, you can use the Mailbox Client IP to replace the functions of the following standalone IP cores, which are considered as mailbox clients:
   - Voltage Sensor Intel Stratix 10 FPGA IP
   - Chip ID Intel Stratix 10 FPGA IP
   - Stratix 10 Serial Flash Mailbox Client Intel FPGA IP

- Advanced SEU Detection Intel Stratix 10 FPGA IP
- Partial Reconfiguration Controller Stratix 10 Intel FPGA IP
- Partial Reconfiguration External Configuration Controller Stratix 10 Intel FPGA IP

1.6. Mailbox Client Intel Stratix 10 FPGA IP Core Use Case Examples

To use the Mailbox Client Intel Stratix 10 FPGA IP core, an Avalon master is required and the simplest Avalon master is JTAG-to-Avalon Master.

Refer to the `rsu1.tcl` script for the examples to perform the following operations using SDM commands:

- Reading device chip ID
- Reading, writing, and erasing operations to QSPI flash, etc.

The function written in the `rsu1.tcl` script can be executed via System Console of the Intel Quartus Prime software.

Below is an example of a sequence to perform the QSPI operation. You must follow this sequence to prevent errors from accessing the QSPI flash. For the details of each QSPI-related commands, refer to Table 2 on page 5.

1. `QSPI OPEN`
2. `QSPI_SET_CS`
3. QSPI operations (such as `QSPI_READ`, `QSPI_WRITE`, `QSPI_ERASE`, `QSPI_READ_DEVICE_REG`, `QSPI_WRITE_DEVICE_REG`, `QSPI_SEND_DEVICE_OP`)
4. `QSPI_CLOSE`

Related Information

Example of Tcl Script
More information on the Tcl script example for performing Mailbox Client operations.

1.7. Document Revision History for the Mailbox Client Intel Stratix 10 FPGA IP Core User Guide

Document Version	Changes
2019.04.19	• Updated the *Feature Description* topic. • Added a note to Figure: *Command and Response Header Format*. • Updated Table: *Mailbox Client Intel Stratix 10 FPGA IP Command and Response Header Description* to update the description for bit[11] of the command and response header. • Updated Table: *Command List and Description* to update the descriptions for `CONFIG_STATUS` and `RSU_STATUS`. • Renamed topic title *Mailbox Client Intel Stratix 10 FPGA IP Core Avalon-MM Interface* to *Mailbox Client Intel Stratix 10 FPGA IP Core Signals*. • Renamed table title *Mailbox Client Intel Stratix 10 FPGA IP Core Avalon-MM Interface* to *Mailbox Client Intel Stratix 10 FPGA IP Core Signal Description*.

continued...

Document Version	Changes
	• Updated Table: *Mailbox Client Intel Stratix 10 FPGA IP Core Signal Description* to include information on clock and reset signals. • Updated Table: *Mailbox Client Intel Stratix 10 FPGA IP Core Avalon-Memory Map* to remove urgent command and urgent FIFO empty space. • Updated the *Using the Mailbox Client Intel Stratix 10 FPGA IP Core* topic: — Added new Figures: *Flow Chart for Writing Command Packet* and *Flow Chart for Reading Response Packet*. — Added a new section—*Restrictions*. — Updated the description in the *Writing Command Packet* section. • Updated the *Mailbox Client Intel Stratix 10 FPGA IP Core Use Case Examples* topic. • Made editorial updates through out the document.
2019.03.14	• Updated the *Mailbox Client Intel Stratix 10 FPGA IP Core User Guide* topic. • Updated Figure: *Mailbox Client Intel Stratix 10 FPGA IP Core and System Block Diagram*. • Updated Table: *Command List and Description*: — Updated the column name *Number of Commands* to *Command Length*. — Updated the column name *Number of Responses* to *Respond Length*. — Corrected the description for `QSPI_READ`, `QSPI_WRITE`, and `QSPI_ERASE`.
2019.02.25	• Updated the description in the *Mailbox Client Intel Stratix 10 FPGA IP Core User Guide* topic. • Updated Figure: *Mailbox Client Intel Stratix 10 FPGA IP Core User Guide*. • Updated Table: *Interrupt Status Register* to update the description for `DATA_VALID`. • Renamed the following topic titles: — *Commands and Error Codes* to *Commands and Responses* — *Commands* to *Operation Commands*. • Updated Table: *Mailbox Client Intel Stratix 10 FPGA IP Command and Response Header Description* to update the descriptions for `Length` and `Command Code/Error Code`. • Updated Table: *Command List and Description*: — Updated the number of responses and description for `CONFIG_STATUS`. — Updated the number of responses for `RSU_STATUS`. — Updated the descriptions for `QSPI_READ`, `QSPI_WRITE`, and `QSPI_ERASE`. • Updated Table: *Mailbox Client Intel Stratix 10 FPGA IP Error Code Responses and Description* to update the description for `UNKNOWN_BR`.

continued...

Document Version	Changes
	• Updated the *Writing Command Packet* and *Reading Command Packet* sections in the *Using the Mailbox Client Intel Stratix 10 FPGA IP Core* topic. • Updated the *Mailbox Client Intel Stratix 10 FPGA IP Core Use Case Examples* topic. • Removed the following topics: — *Example 1: Reading Intel Stratix 10 IDCODE and Voltage* — *Example 2: Read and Write EPCQ-L or QSPI Devices*
2018.10.15	• Updated Table: *Command List and Description* to include the following commands: — Updated the descriptions for `GET_TEMPERATURE`. — Added new commands: • `RSU_IMAGE_UPDATE` • `CONFIG_STATUS` • `RSU_STATUS` — Removed the command `GET_DESIGNHASH`. • Updated Table: *Error Code Responses and Description* to update the value of the following error code responses: — `NOT_CONFIGURED` — `ALT_SDM_MBOX_RESP_DEVICE_BUSY` — `ALT_SDM_MBOX_RESP_NO_VALID_RESP_AVAILABLE` — `ALT_SDM_MBOX_RESP_ERROR` • Added a note to Figure: *Mailbox Client Intel Stratix 10 FPGA IP Core Block Diagram*. • Made minor editorial updates.
2018.02.14	Initial release.