



# **iCE40 UltraPlus I<sup>2</sup>C and SPI Hardened IP Usage Guide — Radiant Software**

## **Technical Note**

FPGA-TN-02053-1.0

February 2018

## Contents

Acronyms in This Document .....	4
1. Introduction .....	5
2. I <sup>2</sup> C IP Core Overview .....	6
2.1. Key Features for iCE40 UltraPlus .....	6
3. I <sup>2</sup> C Usage with Module Generation .....	7
3.1. Wakeup Enable .....	7
3.2. Master Clock (Desired Frequency) .....	8
3.3. Master Clock (Actual Frequency) .....	8
3.4. I <sup>2</sup> C Addressing .....	8
3.5. Interrupts .....	8
3.5.1. Arbitration Lost Interrupts .....	8
3.5.2. TX/RX Ready .....	8
3.5.3. Overrun or NACK .....	9
3.5.4. General Call Interrupts .....	9
3.6. Include IO Buffers .....	9
4. SB_I2C Hard IP Macro Ports and Wrapper Connections .....	10
5. I <sup>2</sup> C Usage Cases .....	10
6. SPI IP Core Overview .....	12
6.1. Key Features .....	12
7. SPI Usage with Module Generation .....	13
7.1. SPI Mode (Enable Slave Interface) .....	13
7.2. SPI Mode (Enable Master Interface) .....	13
7.3. Master Clock Rate (Desired Frequency) .....	14
7.4. Master Clock Rate (Actual Frequency) .....	14
7.5. LSB First .....	14
7.6. Inverted Clock .....	14
7.7. Phase Adjust .....	14
7.8. Slave Handshake Mode .....	14
7.9. Master Chip Selects .....	15
7.10. SPI Controller Interrupts .....	15
7.10.1. TX Ready .....	15
7.10.2. RX Ready .....	15
7.10.3. TX Overrun .....	15
7.10.4. RX Overrun .....	15
7.11. Wakeup Enable .....	15
7.12. Include IO Buffers .....	16
8. SB_SPI Hard IP Macro Ports and Wrapper Connections .....	17
9. iCE40 UltraPlus SPI Usage Cases .....	17
10. Module Generator .....	20
10.1. Key Features .....	20
10.1.1. Enable Left I <sup>2</sup> C .....	21
10.1.2. Enable Right I <sup>2</sup> C .....	21
10.1.3. Enable Left SPI .....	21
10.1.4. Enable Right SPI .....	21
10.1.5. System Clock .....	21
Technical Support Assistance .....	22
Revision History .....	22

## Figures

Figure 1.1. IP Block Diagram for iCE40 UltraPlus.....	5
Figure 3.1. I <sup>2</sup> C Module Interface iCE40 UltraPlusGeneral Call Enable .....	7
Figure 3.2. I <sup>2</sup> C IO Buffers.....	9
Figure 3.3. I <sup>2</sup> C IO Connections.....	9
Figure 5.1. I <sup>2</sup> C Circuit .....	11
Figure 7.1. SPI Module GUI for iCE40 UltraPlus .....	13
Figure 7.2. SPI IO Buffers .....	16
Figure 7.3. SPI IO Connections .....	16
Figure 9.1. External Master SPI Device Accessing the Slave User SPI .....	18
Figure 9.2. iCE40 UltraPlus User SPI Master Accessing One or Multiple External Slave SPI Devices .....	19
Figure 10.1. Soft IP Block Diagram .....	20

## Tables

Table 4.1. Pins for the Hard I <sup>2</sup> C IP.....	10
Table 4.2. Additional Ports in Soft IP .....	10
Table 8.1. Pins for the Hard SPI IP .....	17
Table 8.2. Additional Ports in Soft IP .....	17

## Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
I <sup>2</sup> C	Inter-Integrated Circuit
FIFO	First In First Out
FPGA	Field-Programmable Gate Array
IP	Intellectual Property
POR	Power On Reset
SPI	Serial Peripheral Interface

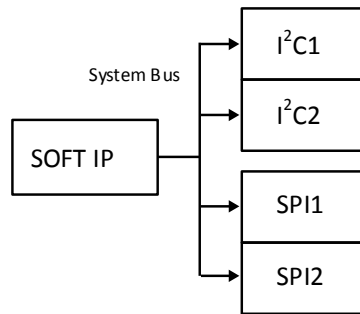
# 1. Introduction

The Lattice Semiconductor iCE40™ family of devices is an ultra-low power Field-Programmable Gate Array (FPGA) and sensor manager designed for ultra-low power mobile applications, such as smartphones, tablets and hand-held devices. The iCE40 UltraPlus™ includes integrated SPI and I<sup>2</sup>C blocks to interface with virtually all mobile sensors and application processors.

The key components available for iCE40 UltraPlus:

- Two I<sup>2</sup>C IP cores located at the upper left corner and upper right corner of the chip
- Two SPI IP cores located at lower left corner and lower right corner of the chip

Each device uses a System Bus to connect its Hard IP to the fabric. The device does not preload the Hard IP registers during configuration, thus a soft IP is required. Using Module Generator to generate the IP is recommended because Module Generator generates the corresponding soft IP as well. For details, refer to the [Module Generator](#) section.



**Figure 1.1. IP Block Diagram for ICE40 UltraPlus**

## 2. I<sup>2</sup>C IP Core Overview

The I<sup>2</sup>C hard IP provides industry standard two pin communication interface that conforms to V2.1 of the I<sup>2</sup>C bus specification. It could be configured as either master or slave port. In master mode, it supports configurable data transfer rate and performs arbitration detection to allow it to operate in multi-master systems. It supports clock stretching in both master and slave modes with enable/disable capability. It supports both 7 bits and 10 bits addressing in slave mode with configurable slave address. It supports general call address detection in both master and slave mode. It provides interrupt logic for easy communicating with host. It also provides configurable digital delay at SDA output for reliably generating start/stop condition.

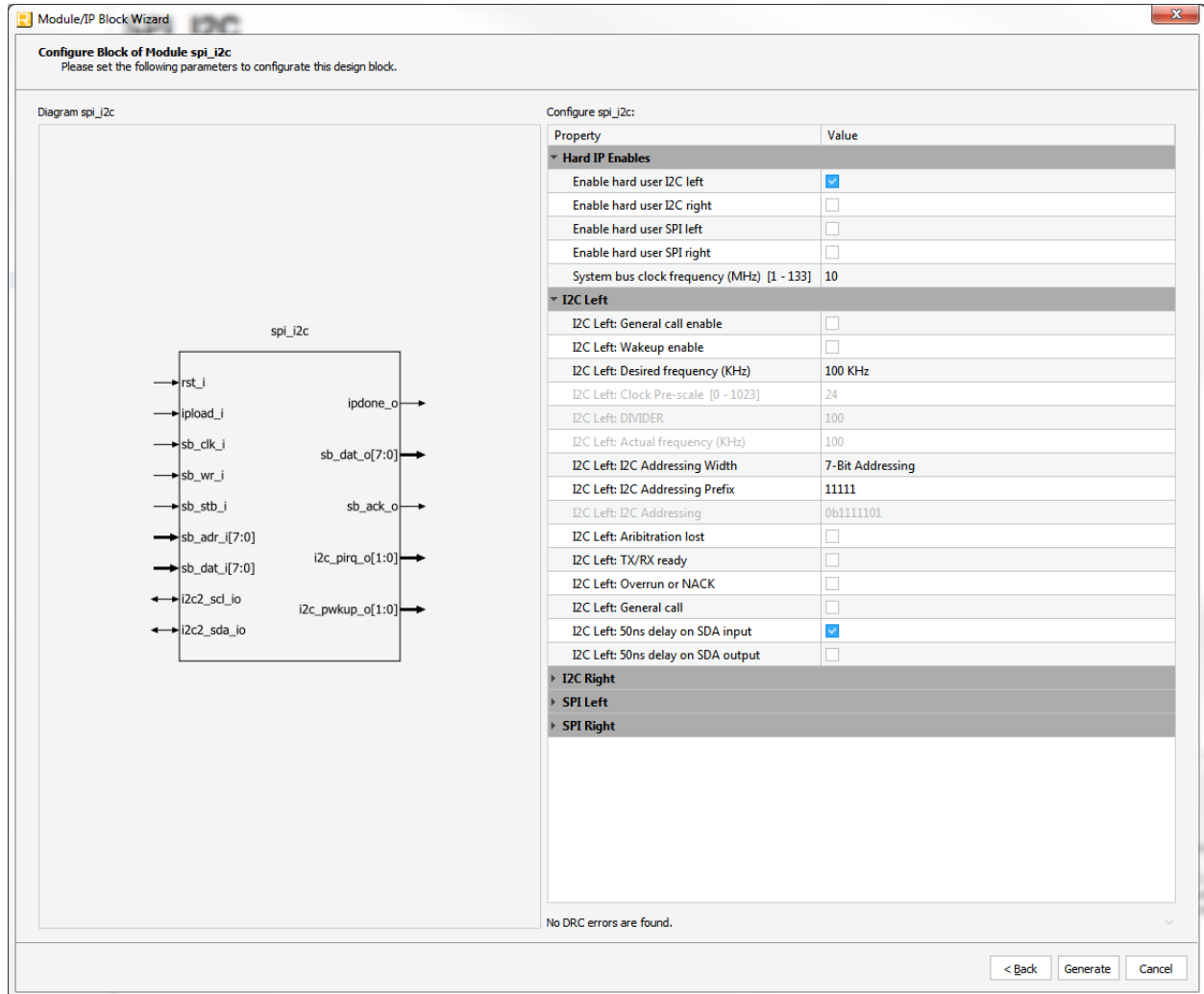
### 2.1. Key Features for iCE40 UltraPlus

- Configurable Master and Slave mode
- Support for 7-bit or 10-bit configurable Slave Address
- Multi-master Arbitration support
- Clock stretching to ensure data setup time
- Up to 400 kHz Data Transfer Speed; also support 100 kHz, 50 kHz modes
- General Call Support
- Master clock source from System Bus clock
- Communication with custom logic through 8-bit wide data bus
- Programmable 5 MSB bits for 7 bits Slave Address or 8 MSB bits for 10 bits Slave Address for user logic Slave I<sup>2</sup>C port.
- I<sup>2</sup>C port and all System Bus addressable registers are reset upon Power On Reset (POR)
- 30 ns analog delay required at SDA input for reliable START, STOP condition detection
- Interface to customer logic through the System Bus Interface

### 3. I<sup>2</sup>C Usage with Module Generation

The Module Generator is the recommended flow for using the I<sup>2</sup>C Hard IP block. See the [Module Generator](#) section for more information.

When the I<sup>2</sup>C portion of the Hard IP block is disabled, all settings on the I<sup>2</sup>C group are disabled.



**Figure 3.1. I<sup>2</sup>C Module Interface iCE40 UltraPlusGeneral Call Enable**

This setting enables the I<sup>2</sup>C General Call response (addresses all devices on the bus using the I<sup>2</sup>C address 0) in Slave mode. This setting can be modified dynamically by enabling the GCEN bit in the I<sup>2</sup>C Control Register I2CCR1.

#### 3.1. Wakeup Enable

Turns on the I<sup>2</sup>C wakeup on address match. Enables the Wakeup port. The WKUPEN bit in the I2CCR1 can be modified dynamically allowing the Wake Up function to be enabled or disabled.

## 3.2. Master Clock (Desired Frequency)

This edit box allows the user to specify a desired master clock frequency. A calculation is then made to determine a divider value to generate a clock close to this value from the input clock. The frequency of the input System Bus clock is specified on the main/general tab. The divider value is rounded to the nearest integer after dividing the input System Bus clock by the value entered in this field.

## 3.3. Master Clock (Actual Frequency)

Since it is not always possible to divide the input System Bus clock to the exact value requested by the user, the actual value is returned in this read-only field. When both the desired I<sup>2</sup>C clock and System Bus clock fields contain valid data and either is updated, the Master Clock field returns the value  $(\text{FREQ\_SB} / \text{I2C\_CLK\_DIVIDER})$ , rounded to an integer as shown in the example below. FREQ\_SB is the System Bus Clock Frequency that the customer will enter in the Module Generator. I2C\_CLK\_DIVIDER will be a factor to be calculated as in the example below.

Master Clock Calculation Example:

1.  $\text{Divider} = \text{I2C\_CLK\_DIVIDER} = (\text{I2C\_CLK\_PRESCALE} + 1) * 4$
2.  $\text{I2C\_CLK\_PRESCALE} = \text{int}(\text{FREQ\_SB} / (4 * \text{I2C\_CLK\_FREQ})) - 1$
3.  $\text{Master Clock (Actual)} = \text{FREQ\_SB} / \text{Divider}$

For example: FREQ\_SB = 42.5 MHz, I2C\_CLK\_FREQ = 400 KHZ

1.  $\text{I2C\_CLK\_PRESCALE} = \text{int}(\text{FREQ\_SB} / \text{I2C\_CLK\_FREQ}) - 1$   
 $\text{I2C\_CLK\_PRESCALE} = \text{int}(42500 / (4 * 400)) - 1 = 26$   
 $\text{Divider} = \text{I2C\_CLK\_DIVIDER} = (\text{I2C\_CLK\_PRESCALE} + 1) * 4$   
 $\text{Divider} = (26 + 1) * 4 = 108$
2.  $\text{Master Clock (Actual)} = \text{FREQ\_SB} / \text{Divider}$   
 $\text{Actual frequency} = 42500 / 108 = 393.5 \text{ kHz}$

In this example, user sees that an I2C\_CLK\_FREQ of 400 kHz cannot be generated. User may choose to use the actual value or change the System Bus Clock. For this reason, user needs to enter a frequency and the actual frequency will be displayed. This value may then be adjusted if desired.

## 3.4. I<sup>2</sup>C Addressing

This option allows the user to set 7-bit or 10-bit addressing and define the Hard I<sup>2</sup>C address.

## 3.5. Interrupts

When any of the interrupts are enabled, the I<sup>2</sup>C port is also enabled.

### 3.5.1. Arbitration Lost Interrupts

An interrupt which indicates I<sup>2</sup>C lost arbitration. This interrupt is bit IRQARBL of the register I2CIRQ. When enabled, it indicates that ARBL is asserted. Writing a '1' to this bit clears the interrupt. This option can be changed dynamically by modifying the bit IRQARBLLEN in the register I2CIRQEN.

### 3.5.2. TX/RX Ready

An interrupt which indicates that the I<sup>2</sup>C transmit data register (I2CTXDR) is empty or that the receive data register (I2CRXDR) is full. The interrupt bit is IRQTRRDY of the register I2CIRQ. When enabled, it indicates that TRRDY is asserted. Writing a '1' to this bit clears the interrupt. This option can be changed dynamically by modifying the bit IRQTRRDYEN in the register I2CIRQEN.



### 3.5.3. Overrun or NACK

An interrupt which indicates that the I2CRXDR received new data before the previous data. The interrupt is bit IRQROE of the register I2CIRQ. When enabled, it indicates that ROE is asserted. Writing a '1' to this bit clears the interrupt. This option can be changed dynamically by modifying the bit IRQROEEN in the register I2CIRQEN.

### 3.5.4. General Call Interrupts

An interrupt which indicates that a general call has occurred. The interrupt is bit IRQHGC of the register I2CIRQ. When enabled, it indicates that ROE is asserted. Writing a '1' to this bit clears the interrupt. This option can be changed dynamically by modifying the bit IRQHGCEN in the register I2CIRQEN.

## 3.6. Include IO Buffers

Includes buffers for the I<sup>2</sup>C pins.

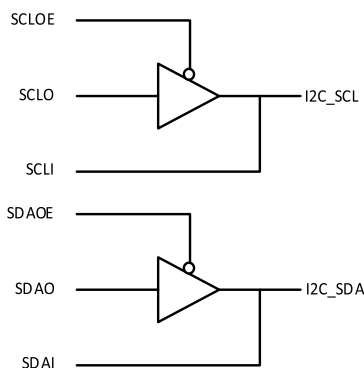


Figure 3.2. I<sup>2</sup>C IO Buffers

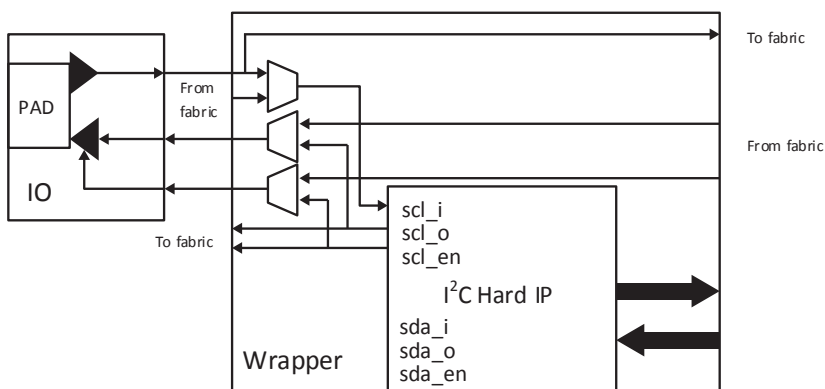


Figure 3.3. I<sup>2</sup>C IO Connections

## 4. SB\_I2C Hard IP Macro Ports and Wrapper Connections

When the I<sup>2</sup>C Hard IP is enabled, the necessary signals are included in the generated module.

**Table 4.1. Pins for the Hard I<sup>2</sup>C IP**

Pin Name	Pin Direction	Description
sb_clk_i	I	System clock input
sb_wr_i	O	System Read/Write input. R=0, W=1
sb_stb_i	O	System Strobe Signal
sb_adr_i[7:0]	I	System Bus Control Registers Address
sb_dat_i[7:0]	I	System Data Input
sb_dat_o[7:0]	O	System Data Output
sb_ack_o	O	System Acknowledgement
i2c_pirq_o	O	Interrupt request output signal of the I <sup>2</sup> C core – The intended use of this signal is for it to be connected to a Master controller (such as a microcontroller or state machine) and request an interrupt when a specific condition is met.
i2c_pwkup_o	O	Wake-up signal – The signal is enabled only if the Wakeup Enable feature is set.
i2cx_scl_io*	BI-directional	Open drain clock line of the I <sup>2</sup> C core – The signal is an output if the I <sup>2</sup> C core is performing a Master operation. The signal is an input for Slave operations. This signal can use a dedicated I/O pin.
i2cx_sda_io	BI-directional	Open drain data line of the I <sup>2</sup> C core – The signal is an output when data is transmitted from the I <sup>2</sup> C core. The signal is an input when data is received into the I <sup>2</sup> C core. This signal can use dedicated I/O pin.

**\*Note:** x indicates the I<sup>2</sup>C: x=1 for the left I<sup>2</sup>C; x=2 for the right I<sup>2</sup>C.

Below is the list of additional ports included in Soft IP.

**Table 4.2. Additional Ports in Soft IP**

Pin Name	Pin Direction	Description
rst_i	I	System active HIGH reset
ipload_i	I	When asserted, the soft IP logic programs or initialize the I <sup>2</sup> C registers based on parameter settings
ipdone_o	O	Indicates the initialization of I <sup>2</sup> C registers is done

## 5. I<sup>2</sup>C Usage Cases

The I<sup>2</sup>C usage cases described below refer to [Figure 5.1](#).

1. Master iCE40 I<sup>2</sup>C Accessing Slave External I<sup>2</sup>C Devices
  - a. A System Bus Master is implemented in the iCE40 logic.
  - b. I<sup>2</sup>C devices 1, 2, and 3 are all Slave devices.
  - c. The Master performs bus transactions to the left I<sup>2</sup>C controller to access external Slave I<sup>2</sup>C Device 1 on Bus A.
  - d. The Master performs bus transactions to the right I<sup>2</sup>C controller to access the external Slave I<sup>2</sup>C Devices number 2 or 3 on Bus B.
2. External Master I<sup>2</sup>C Device Accessing Slave iCE40 I<sup>2</sup>C
  - a. The I<sup>2</sup>C devices 1, 2, and 3 are I<sup>2</sup>C Master devices.
  - b. The external master I<sup>2</sup>C Device 1 on Bus A performs I<sup>2</sup>C memory cycles to access the left I<sup>2</sup>C controller using address yyyxxxxx01.
  - c. The external master I<sup>2</sup>C Device 2 or 3 on Bus B performs I<sup>2</sup>C memory cycles to access the right I<sup>2</sup>C User with the address yyyxxxxx10.
  - d. A Master in the iCE40 fabric must manage data reception and transmission. The Master can use interrupts or polling techniques to manage data transfer, and to prevent data overrun conditions.

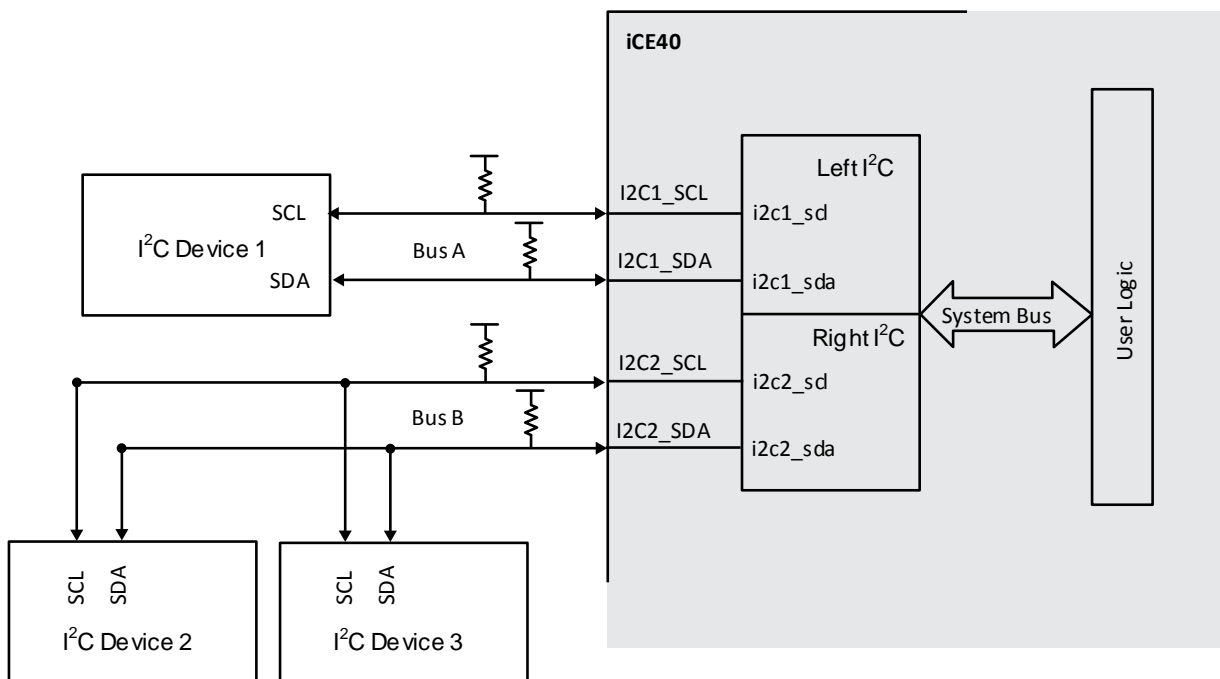


Figure 5.1. I<sup>2</sup>C Circuit

## 6. SPI IP Core Overview

The SPI hard IP provide industry standard four-pin communication interface with 8-bit wide System Bus to communicate with System Host. It could be configured as Master or Slave SPI port with separate Chip Select Pin. In master mode, it provides programmable baud rate, and supports CS HOLD capability for multiple transfers. It provides variety status flags, such as Mode Fault Error flag, Transmit/Receive status flag etc. for easy communicate with system host.

### 6.1. Key Features

Configurable Master and Slave mode

- Full-Duplex data transfer
- Mode fault error flag with CPU interrupt capability
- Double-buffered data register
- Serial clock with programmable polarity and phase
- LSB First or MSB First Data Transfer
- Communicate with custom logic through 8 bit wide System Bus
- Maximum 4 Slave select output could be routed to any IO through fabric for Master mode
- Slave chip select pin could be routed to any IO through fabric for custom logic
- Clock source for Master clock (MCLK) generation come from System Bus clock
- Optionally send out status byte to inform remote SPI master the slave receiving register status during slave write.
- Optionally enable the Slow Response Mode for slave SPI read, which will automatically deploy the Lattice specific protocol to resolve the issue that caused by the slow initial respond of the System host at high SPI clock rate.
- The SPI port and all System Bus addressable registers will be reset upon Power On Reset (POR).
- Interface to customer logic through the System Bus Interface

## 7. SPI Usage with Module Generation

The Module generator is the recommended flow for using the SPI Hard IP block. See the [Module Generator](#) section for more information.

When the SPI portion of the Hard IP block is disabled, all settings on the SPI tab shall be disabled.

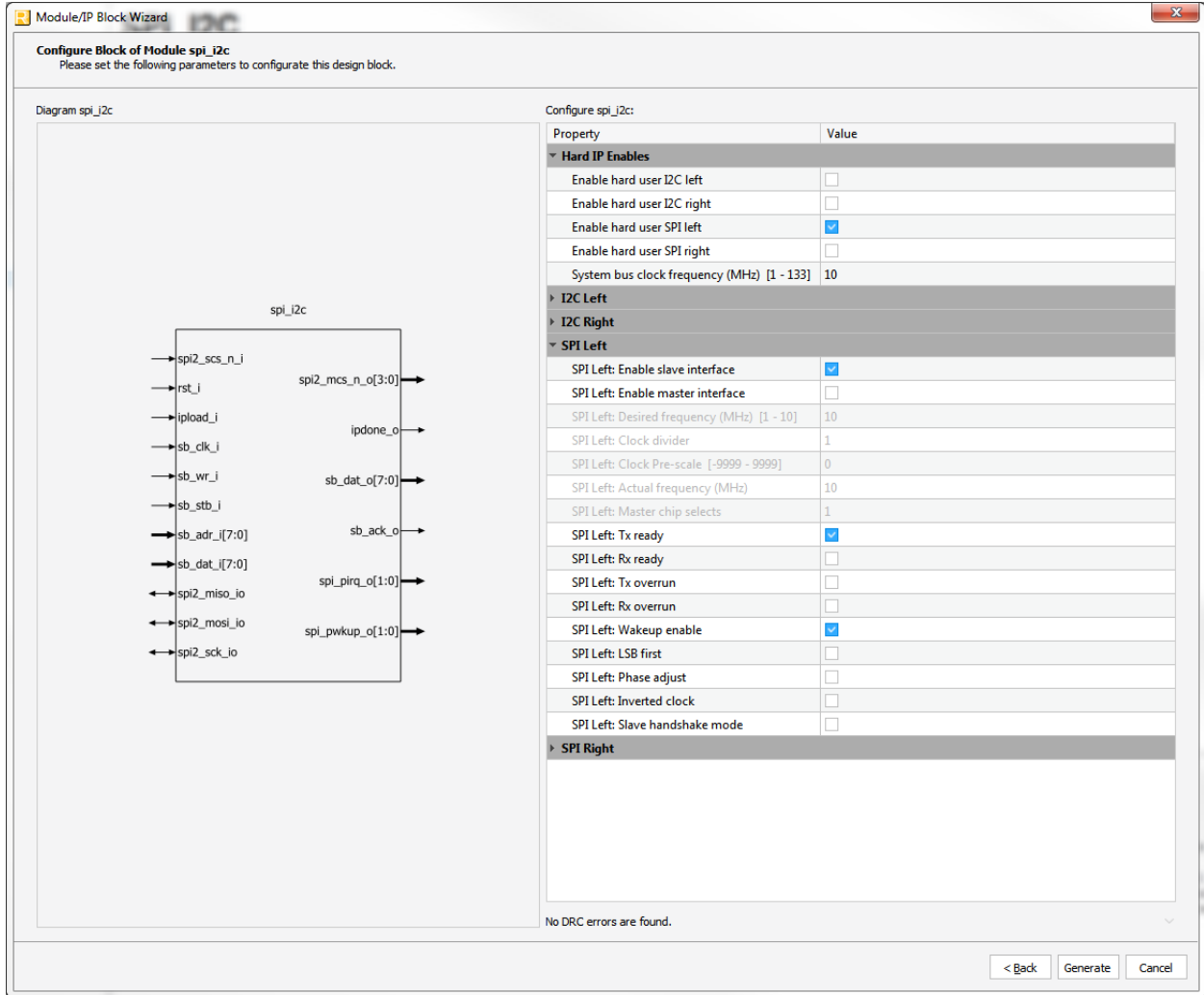


Figure 7.1. SPI Module GUI for iCE40 UltraPlus

### 7.1. SPI Mode (Enable Slave Interface)

This option allows the user to enable Slave Mode interface for the initial state of the SPI block. By default, Slave Mode interface is enabled. Options and ports that are applicable to only one mode will be disabled when the other is chosen.

### 7.2. SPI Mode (Enable Master Interface)

This option allows the user to enable Master Mode interface for the initial state of the SPI block. This option can be updated dynamically by modifying the MSTR bit of the register SPICR2. Options and ports that are applicable to only one mode will be disabled when the other is chosen.

### 7.3. Master Clock Rate (Desired Frequency)

(“Enable Master Interface” only) This edit box allows the user to specify a desired master clock frequency. A calculation is then made to determine a divider value to generate a clock close to this value from the input clock. The frequency of the input System Bus clock is specified on the main/general tab. The divider value is rounded to the nearest integer after dividing the input System Bus clock by the value entered in this field.

### 7.4. Master Clock Rate (Actual Frequency)

(“Enable Master Interface” only) Since it is not always possible to divide the input System Bus clock exactly to that requested by the user, the actual value will be returned in this read-only field. When both the desired SPI clock and System Bus clock fields have valid data and either is updated, this field returns the value  $(FREQ\_SB/SPI\_CLK\_DIVIDER)$ , rounded to two decimal places as shown in the example below.  $FREQ\_SB$  is the System Bus Clock Frequency that the customer enters in the Module Generator.  $SPI\_CLK\_DIVIDER$  is a factor to be calculated, which is also shown in the example below.

Master Clock Calculation Example:

1.  $Divider = SPI\_CLK\_DIVIDER = FREQ\_SB / SPI\_CLK\_FREQ$
2.  $Master\ Clock\ (Actual) = FREQ\_SB / DividerInteger$   
For example:  $FREQ\_SB = 66\ MHz$ ,  $SPI\_CLK\_FREQ = 25\ MHz$ 
  1.  $Divider = SPI\_CLK\_DIVIDER = FREQ\_SB / SPI\_CLK\_FREQ$   
 $Divider = 66 / 25 = 2.64$   
Therefore  $ROUND\ DividerInteger = 3$
  2.  $Master\ Clock\ (Actual) = FREQ\_SB / DividerInteger$   
 $Actual\ frequency = 66 / 3 = 22\ MHz$

In this example, if the user sees that an  $SPI\_CLK\_FREQ$  of 25 MHz cannot be generated, the user may choose to use the actual value or change the System Bus Clock. For this reason, user needs to enter a frequency and the actual frequency will be displayed. This value may then be adjusted if desired.

### 7.5. LSB First

This setting specifies the order of the serial shift of a byte of data. The data order (MSB or LSB first) is programmable within the SPI core. This option can be updated dynamically by modifying the LSBF bit in the register SPICR2.

### 7.6. Inverted Clock

This option inverts the clock polarity used to sample and output data is programmable for the SPI core. When selected, the edge changes from the rising to the falling clock edge. This option can be updated dynamically by accessing the CPOL bit of register SPICR2.

### 7.7. Phase Adjust

An alternate clock-data relationship is available for SPI devices with particular requirements. This option allows user to specify a phase change to match the application. This option can be updated dynamically by accessing the CPHA bit in the register SPICR2.

### 7.8. Slave Handshake Mode

Enables Lattice proprietary extension to the SPI protocol. For use when the internal sup-port circuit (e.g. WISHBONE host) cannot respond with initial data within the time required, and to make the Slave read out data predictably available at high SPI clock rates. This option can be updated dynamically by accessing the SDBRE bit in the register SPICR2.

## 7.9. Master Chip Selects

(“Enable Master Interface” only) The core has the ability to provide up to 4 individual chip select outputs for master operation. This field allows the user to prevent extra chip selects from being brought out of the core. This option can be updated dynamically by modifying the register SPICSR.

## 7.10. SPI Controller Interrupts

### 7.10.1. TX Ready

An interrupt which indicates the SPI transmit data register (SPITXDR) is empty. The interrupt bit is IRQTRDY of the register SPIIRQ. When enabled, indicates TRDY was asserted. Write a ‘1’ to this bit to clear the interrupt. This option can be changed dynamically by modifying the bit IRQTRDYEN in the register SPIIRQEN.

### 7.10.2. RX Ready

An interrupt which indicates the receive data register (SPIRXDR) contains valid receive data. The interrupt is bit IRQRRDY of the register SPIIRQ. When enabled, indicates RRDY was asserted. Write a ‘1’ to this bit to clear the interrupt. This option can be change dynamically by modifying the bit IRQRRDYEN in the register SPICSR.

### 7.10.3. TX Overrun

An interrupt which indicates the Slave SPI chip select (SPI\_SCSN) was driven low while an SPI Master. The interrupt is bit IRQMDF of the register SPIIRQ. When enabled, indicates MDF (Mode Fault) was asserted. Write a ‘1’ to this bit to clear the interrupt. This option can be changed dynamically by modifying the bit IRQMDFEN in the register SPIIRQEN.

### 7.10.4. RX Overrun

An interrupt which indicates SPIRXDR received new data before the previous data. The interrupt is bit IRQROE of the register SPIIRQ. When enabled, indicates ROE was asserted. Write a ‘1’ to this bit to clear the interrupt. This option can be changed dynamically by modifying the bit IRQROEEN in the register SPIIRQEN.

## 7.11. Wakeup Enable

The core can optionally provide a wakeup signal to the device to resume from low power mode. This option can be updated dynamically by modifying the bit WKUPEN\_USER in the register SPICR1.

## 7.12. Include IO Buffers

Includes buffers for the SPI pins.

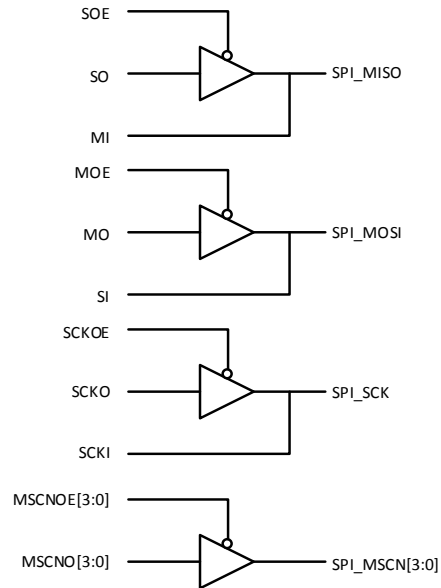


Figure 7.2. SPI IO Buffers

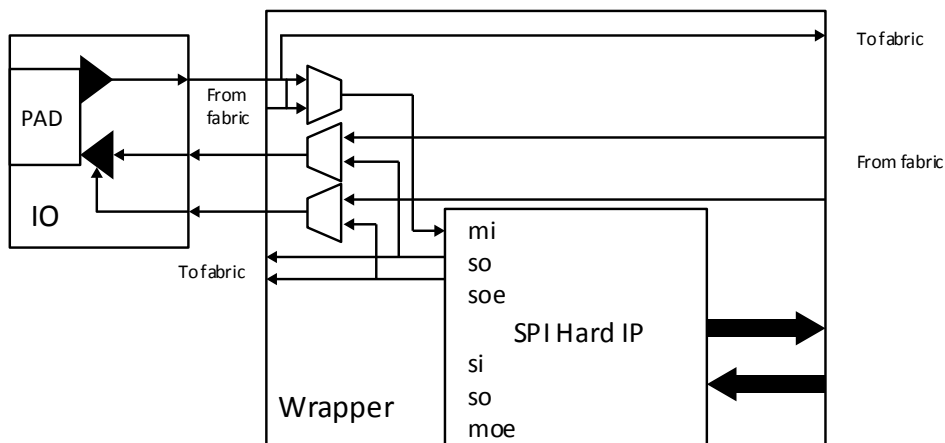


Figure 7.3. SPI IO Connections



## 8. SB\_SPI Hard IP Macro Ports and Wrapper Connections

When the SPI Hard IP is enabled, the necessary signals will be included in the generated module.

**Table 8.1. Pins for the Hard SPI IP**

Pin Name	Pin Direction	Description
sb_clk_i	I	System clock input
sb_wr_i	O	System Read/Write input. R=0, W=1
sb_stb_i	O	System Strobe Signal
sb_adr_i[7:0]	I	System Bus Control Registers Address
sb_dat_i[7:0]	I	System Data Input
sb_dat_o[7:0]	O	System Data Output
sb_ack_o	O	System Acknowledgement
spi_pirq_o	O	Interrupt request output signal of the I <sup>2</sup> C core – The intended use of this signal is for it to be connected to a Master controller (i.e. a microcontroller or state machine) and request an interrupt when a specific condition is met.
spi_pwkup_o	O	Wake-up signal – The signal is enabled only if the Wakeup Enable feature has been set.
spix_sck_io*	BI-directional	The signal is an output if the SPI core is in Master mode (MCLK). The signal is an input if the SPI core is in Slave mode (CCLK). This signal could use dedicated I/O pin.
spix_mosi_io*	BI-directional	The signal is an output if the SPI core is in Master mode (SISPI). The signal is an input if the SPI core is in Slave mode (SI). This signal could use dedicated I/O pin.
spix_miso_io*	BI-directional	The signal is an input if the SPI core is in Master mode (SPISO). The signal is an output if the SPI core is in Slave mode (SO). This signal could use dedicated I/O pin.
spix_scs_n_i*	I	User Slave Chip Select (Active Low). An external SPI Master controller asserts this signal to transfer data to/from the SPI Controllers Transmit Data/Receive Data registers. This signal could use dedicated I/O pin. The dedicated pin is shared with SPI_MCSN[0].
spix_mcs_n_o[3:0]*	O	Master Chip Select (Active Low). Up to 4 independent slave SPI devices can be accessed using the SPI Controller when it is in Master SPI mode. Only SPI_MCSN[0] could use dedicated I/O pin.

\*Note: x indicates the SPI: x=1 for the left SPI; x=2 for the right SPI.

Below is the list of additional ports included in Soft IP.

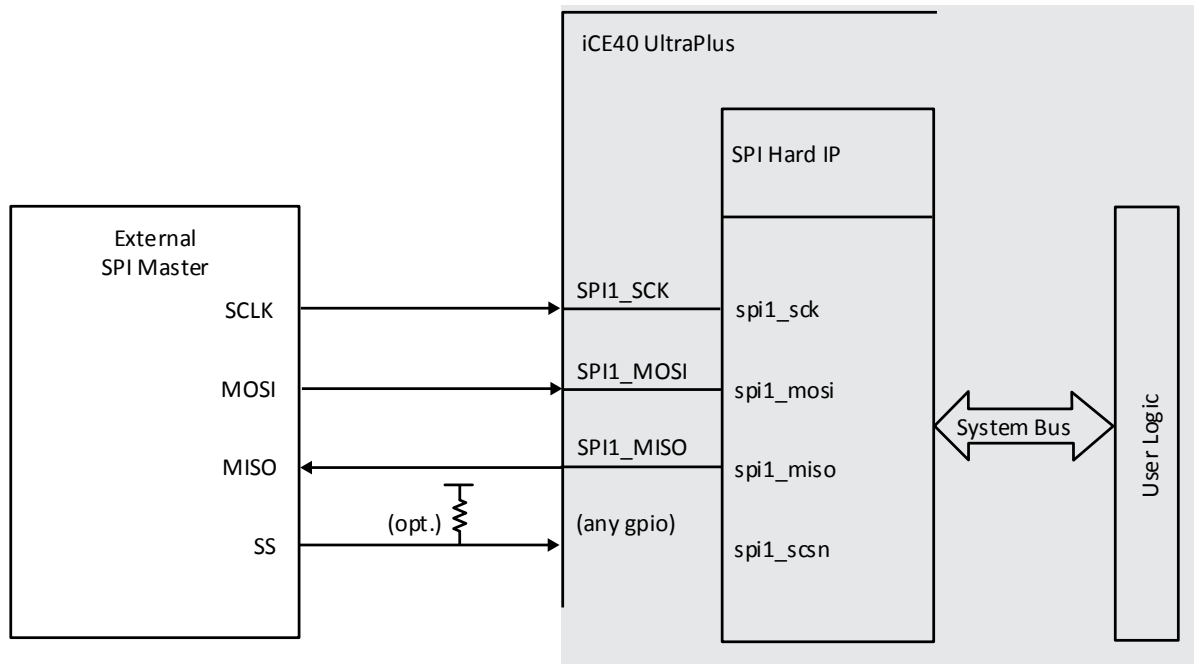
**Table 8.2. Additional Ports in Soft IP**

Pin Name	Pin Direction	Description
rst_i	I	System active HIGH reset
ipload_i	I	When asserted, the soft IP logic programs or initialize the I <sup>2</sup> C registers based on parameter settings
ipdone_o	O	Indicates the initialization of I <sup>2</sup> C registers is done

## 9. iCE40 UltraPlus SPI Usage Cases

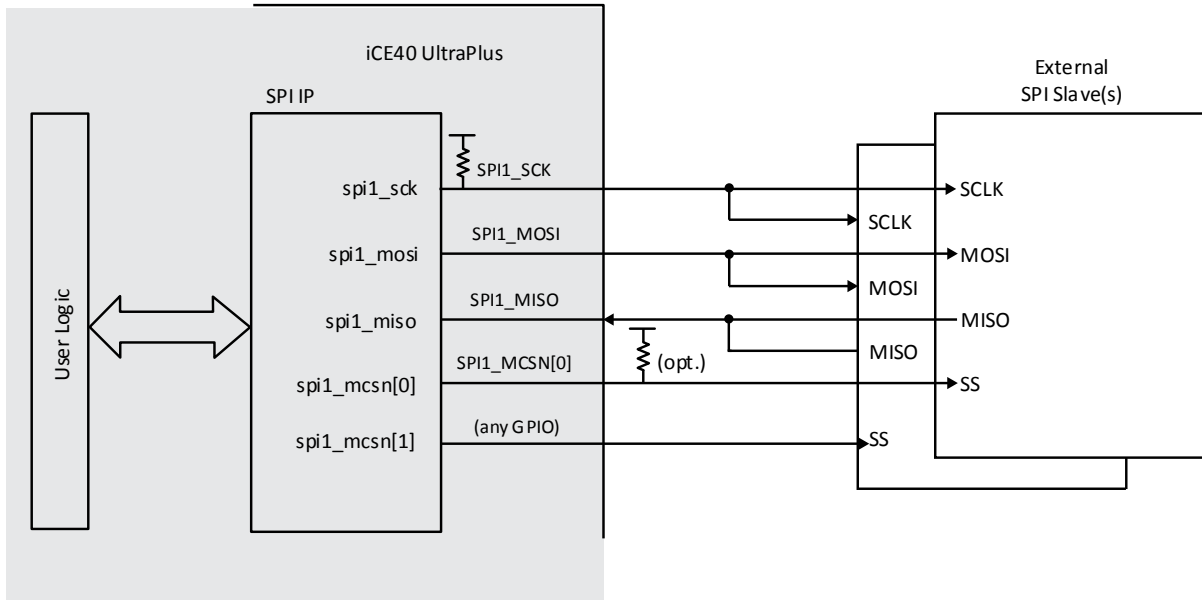
Refer to [Figure 9.1](#) and [Figure 9.2](#) for the SPI usage cases described below.

1. External Master SPI Device Accessing the Slave iCE40 UltraPlus User SPI
  - a. The External Master SPI is connected to the iCE40 UltraPlus using the dedicated SI, SO, CCLK pins. The spi\_scsn is placed on any Generic I/O. The SPI Mode is set to Slave only.
  - b. A Master controller is implemented in the general purpose logic array. The master controller monitors the availability to transmit or receive data by polling the SPI status registers, or by responding to interrupts generated by the SPI Controller.



**Figure 9.1. External Master SPI Device Accessing the Slave User SPI**

2. iCE40 UltraPlus User SPI Master accessing one or multiple External Slave SPI devices
  - a. The iCE40 UltraPlus Master is connected to External SPI Slave devices. The Chip Selects are configured as follows:
    - i. The iCE40 UltraPlus SPI Master Chip Select spi\_mcsn[0] is placed on the dedicated CSSPIN and connected to the External Slave Chip Select.
    - ii. The UltraPlus SPI Master Chip Select spi\_mcsn[1] is placed on any I/O and connected to another External Slave Chip Select.
    - iii. Up to 4 External Slave SPIs can be connected using spi\_mcsn[3:0].
  - b. A Master controller is implemented in the iCE40 UltraPlus general logic. It controls transfers to the slave SPI devices. It can use a polling method, or it can use SPI Controller interrupts to manage transfer and reception of data.



**Figure 9.2. iCE40 UltraPlus User SPI Master Accessing One or Multiple External Slave SPI Devices**

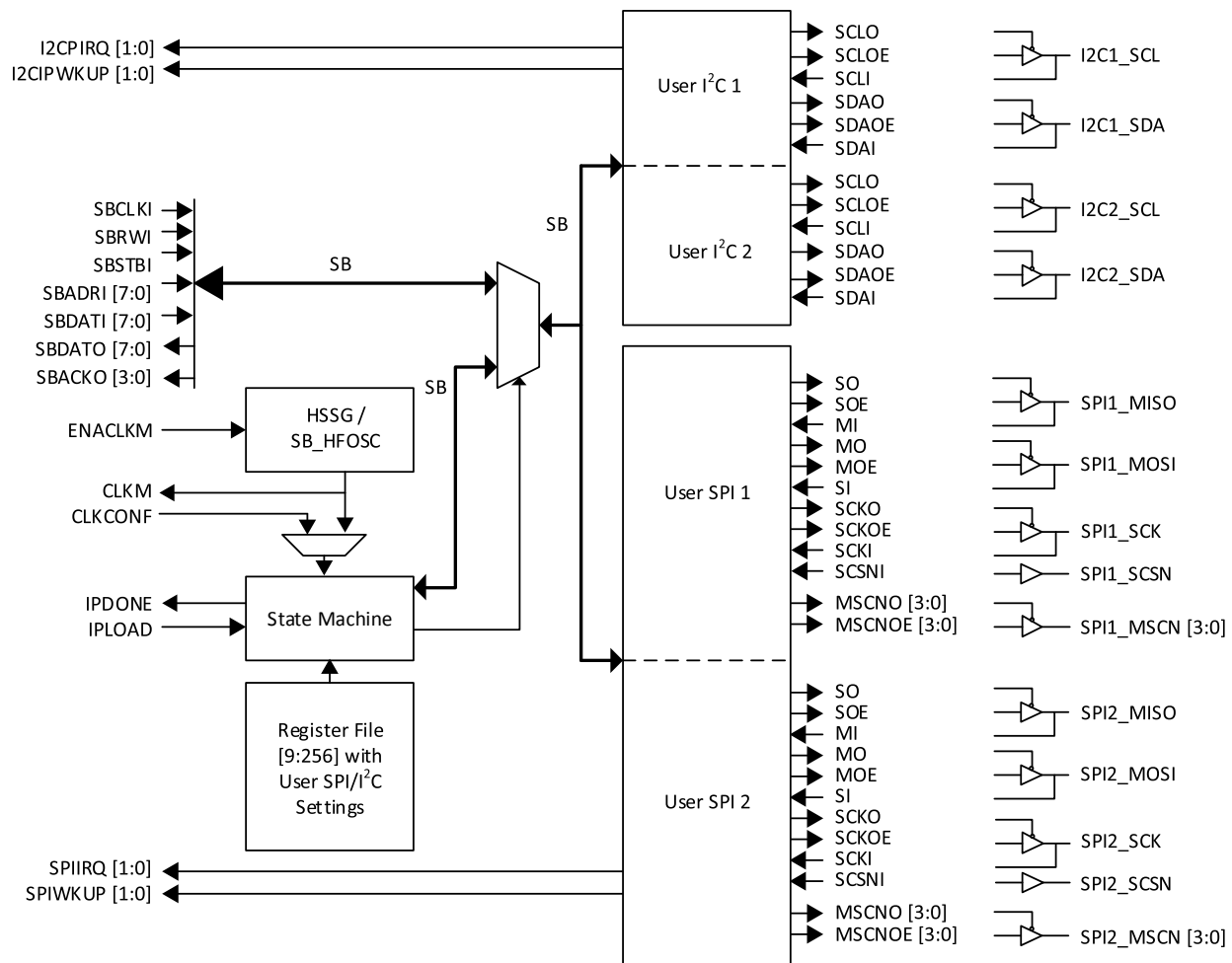
# 10. Module Generator

iCE40 UltraPlus uses a System Bus to connect its Hard IP to the fabric. The System Bus is connected to two SPI and two I<sup>2</sup>C Hard IPs. The iCE40 UltraPlus devices do not preload the Hard IP registers during configuration, so a soft IP will be required.

## 10.1. Key Features

- Module Generator generates the Soft IP which is a wrapper around the Hard IP.
- Soft IP releases the System Bus after the Hard IPs are configured.
- The Register file is [9:256].
  - The address size is 256 bits which encompasses all addresses of the Hard IPs.
  - The data size is 9 bits with the format [W, Register Data]. When W=1, the state machine writes the corresponding Hard IP address with data.

See FPGA-TN-02011, [Advanced iCE40 I2C and SPI Hardened IP Usage Guide](#) for more information.



Notes: Only SPI\*\_MSCN[1:0] have dedicated pins

IPDONE = 1 when Hard IP Configuration is complete, IPDONE = 0 during configuration

IPILOAD is an input signal which is used to begin configuration on a positive edge

**Figure 10.1. Soft IP Block Diagram**

### **10.1.1. Enable Left I<sup>2</sup>C**

This option allows the user to enable left I<sup>2</sup>C on the I<sup>2</sup>C Tab.

### **10.1.2. Enable Right I<sup>2</sup>C**

This option allows the user to enable right I<sup>2</sup>C on the I<sup>2</sup>C Tab.

### **10.1.3. Enable Left SPI**

This option allows the user to enable left SPI on the SPI Tab.

### **10.1.4. Enable Right SPI**

This option allows the user to enable right SPI on the SPI Tab.

### **10.1.5. System Clock**

User setting which is used to set the divider settings of the I<sup>2</sup>C and SPI.

## Technical Support Assistance

Submit a technical support case through [www.latticesemi.com/techsupport](http://www.latticesemi.com/techsupport).

## Revision History

Date	Version	Change Summary
February 2018	1.0	Initial release.



7<sup>th</sup> Floor, 111 SW 5<sup>th</sup> Avenue  
Portland, OR 97204, USA  
T 503.268.8000  
[www.latticesemi.com](http://www.latticesemi.com)