

## iCE65 as SPI Master Interface

### Overview

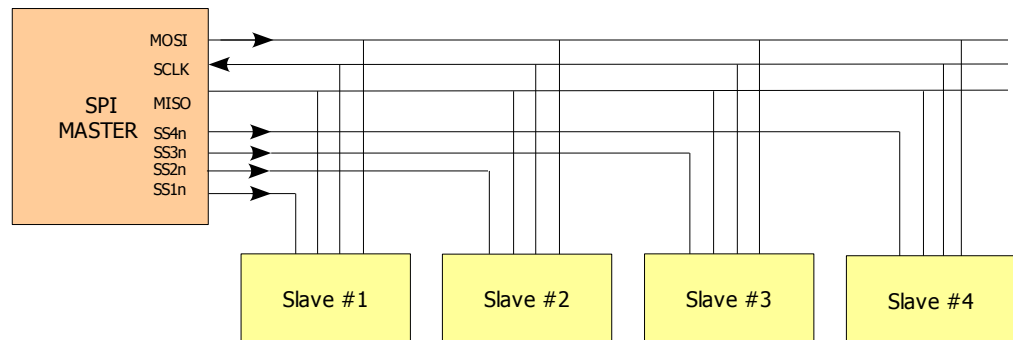
This design example illustrates the implementation of an SPI Master using ICE65 FPGAs. It generates all the necessary interfacing signals to connect an SPI Master to an SPI Slave as well as Microprocessor using standard address/data bus system and hence enables host processor to connect to the SPI bus and transact with SPI Slaves. The Master is capable of handling up to four SPI slaves.

### SPI Description

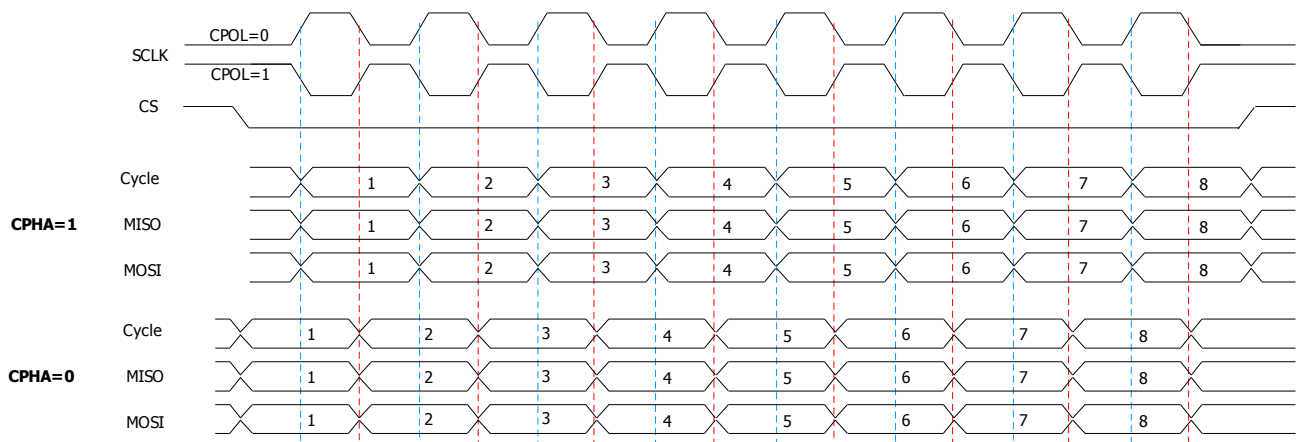
SPI™, or Serial Peripheral Interface is a serial, synchronous 4-wire communication protocol that is standard across microprocessors, microcontrollers, and peripherals that enables communication between microprocessors, peripherals and inter-processors. The communication makes use of Master - Slave system using 4 lines (3 common and 1 exclusive) as follows:

- MOSI (Master Out, Slave in)
- MISO (Master In, Slave Out)
- SCLK (Serial Clock)
- SS (Slave select)

The main features of the SPI interface, unlike many other serial interfaces, is that it allows for a full duplex serial communication. This is possible due to the presence of exclusive lines for data in both directions. The availability of another exclusive line used to select a particular Slave, reduces the overhead of address decoding in a multi-slave environment. These two features combine to add great speeds on the SPI bus (with clock speeds up to 70 MHz).



**Fig 1: SPI bus system**



**Fig 2: SPI variants and typical timing**

# Implementation

A typical SPI Bus with a Master and multiple Slaves is illustrated in Fig 1 above. Fig 2 illustrates a typical timing diagram of the SPI system. It also illustrates the different variants of the SPI that is resultant of an inverted clock polarity and shifted sense of clock phase from the SPI bus point of view.

Features of SPI Master Design Example described in this application note: 1. Generates active low chip-select for up-to 4 Slaves. 2. Operates in CPOL=0 and CPHA=0 mode. 3. Each Data Transaction length is 16-bits 4. Read and Write FIFO of 16x16 size.

The Fig 3 below shows the overall interface of SPI master to both host processor and SPI slave device. The controller FSM block monitors host side transaction requests to the SPI master through CPU interface logic. Depending on the host processor request, appropriate control signals are sent to the controller FSM along with the address and data. The controller FSM is the heart of the SPI master which makes sure that correct transactions take place in sequence and that the SPI devices timings are not violated in the process.

**sclk** is the SPI bus clock, derived from **clk\_sys** (system clock). Master's data on **mosi** is valid during the first rising edge of **sclk**. Similarly the slave posts its data when **sclk** is low so that the Master can sample **miso** during the subsequent rising edge of **sclk** (CPOL=0 and CPHA=0 mode).

The host processor selects the SPI master by holding the **cs\_n** signal LOW. When **cs\_n** is low, the address bus bits determine to which SPI register the host is pointing. The host processor reads a particular register when **oe\_n** is LOW and writes to a particular register when **we\_n** is LOW. The host uses reads (polling) status register contents to obtain the status of data transaction with SPI Slave. Read latency is 3 cycles. Fig 5, 6, and 7 illustrates timing diagram of SPI Master Read/Write Cycles.

Table 1 lists the SPI Master registers and the address mapping.

Table 3 lists the configuration register bit mapping.

- CLKDIV{3,2,1,0} bits configures **sclk** frequency. Table 2 lists relationship between **clk\_sys** frequency and **sclk**.

- START bit denotes SPI read/write transaction start. Must be set to HIGH after configuration to start data transaction.

- FIFORST clears Read/Write FIFO when HIGH

- LEN{3,2,1,0} represents block length of data transaction. Example: "0000" represents block length 1, "1111" represents block length 16.

- ADDR{1,0} denotes slave select address. Example: when "00", **s\_select(0)** goes low and when "11", **s\_select(3)** goes low.

Table 4 lists the status register bit mapping

- WFFULL, WFHFULL and WFEMPTY bits represent Write FIFO full, half full and empty

- RDFULL, RDHFULL and RDEEMPTY bits represent Read FIFO full, half full and empty

- BUSY bit HIGH indicates SPI Master is busy

SPI Master works in Read-only Mode when Write FIFO is empty. FIFORST bit of configuration register can be used to clear both the FIFO contents, which helps discard read data while write operation to SPI Slave.

Table 5 lists the pin description of SPI Master.

| Address | Mode | Register Description   |
|---------|------|------------------------|
| 00      | R/W  | Configuration Register |
| 01      | W    | Write Data FIFO(16x16) |
| 10      | R    | Status Register        |
| 11      | R    | Read Data FIFO(16x16)  |

**Table 1: SPI Master Registers**

| CLKDIV | SCLK Frequency |
|--------|----------------|
| 0000   | clk_sys/4      |
| 0001   | clk_sys/6      |
| 0010   | clk_sys/8      |
| ...    |                |
| 1111   | clk_sys/34     |

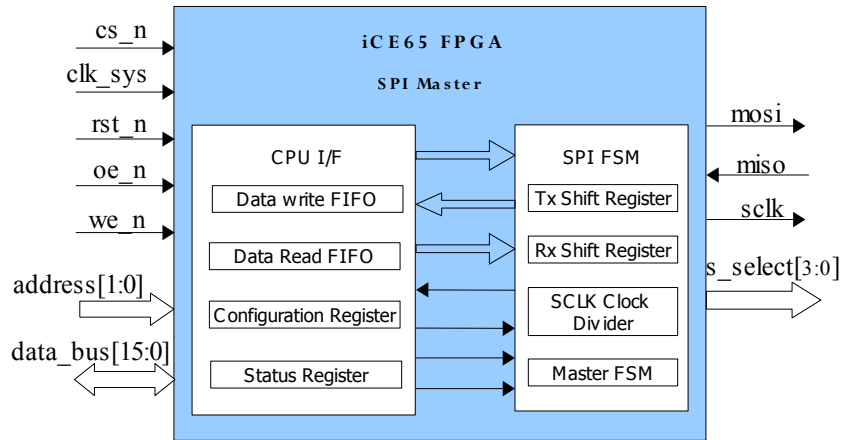
**Table 2: SPI SCLK Frequency**

| C11     | C10     | C9      | C8      | C7    | C6      | C5   | C4   | C3   | C2   | C1    | C0    |
|---------|---------|---------|---------|-------|---------|------|------|------|------|-------|-------|
| CLKDIV3 | CLKDIV2 | CLKDIV1 | CLKDIV0 | START | FIFORST | LEN3 | LEN2 | LEN1 | LEN0 | ADDR1 | ADDR0 |

**Table 3: Configuration Register**

| S7 | S6     | S5      | S4      | S3     | S2      | S1       | S0   |
|----|--------|---------|---------|--------|---------|----------|------|
| X  | WFFULL | WFHFULL | WFEMPTY | RDFULL | RDHFULL | RDEEMPTY | BUSY |

**Table 4: Status Register**



**Fig 3: Block Diagram of SPI Master**

| Port           | Direction | Description                   |
|----------------|-----------|-------------------------------|
| cs_n           | Input     | Active low chip select        |
| clk_sys        | Input     | System clock                  |
| rst_n          | Input     | System reset, Active Low      |
| oe_n           | Input     | Read Enable, Active low       |
| we_n           | Input     | Write Enable, Active low      |
| address[1:0]   | Input     | Processor Address bus         |
| data_bus[15:0] | Inout     | Processor data bus            |
| mosi           | Output    | SPI Master Transmit line      |
| miso           | Input     | SPI Master Receive line       |
| sclk           | Output    | Slave clock                   |
| s_select[3:0]  | Output    | Slave chip select, Active Low |

**Table 5: Pin Description**

Table 6 below shows the post P&R resource utilization summary of this design when implemented on an iCE65 FPGA.

| Device          | Logic Cells | IO Cells |
|-----------------|-------------|----------|
| iCE65L04-UCB284 | 400         | 30       |

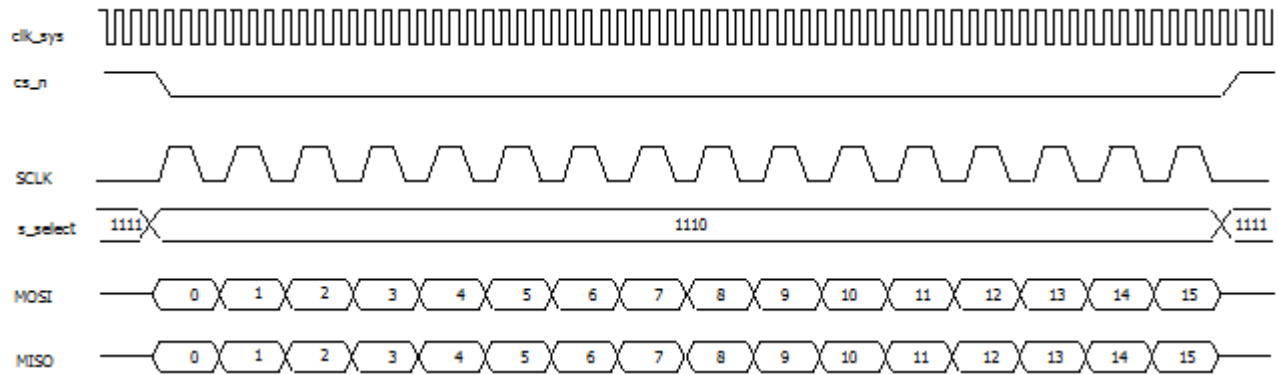
**Table 6: Resource Utilization**

## Conclusion

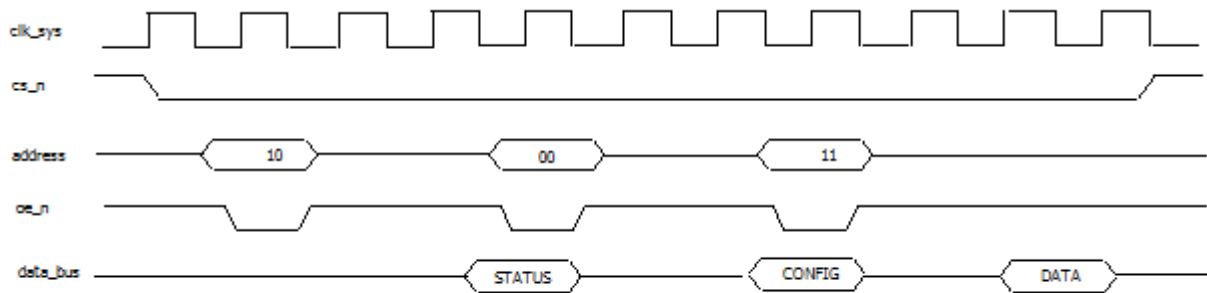
This design example demonstrates the implementation of a SPI Master using iCE FPGAs. The low pin count properties of the 4 wire SPI serial interface, and its high throughput characteristics due to full duplex communication are well complemented by iCE FPGA's very low power capabilities. This makes iCE FPGAs an obvious choice for SPI Master interface in all low power applications.

## Disclaimer

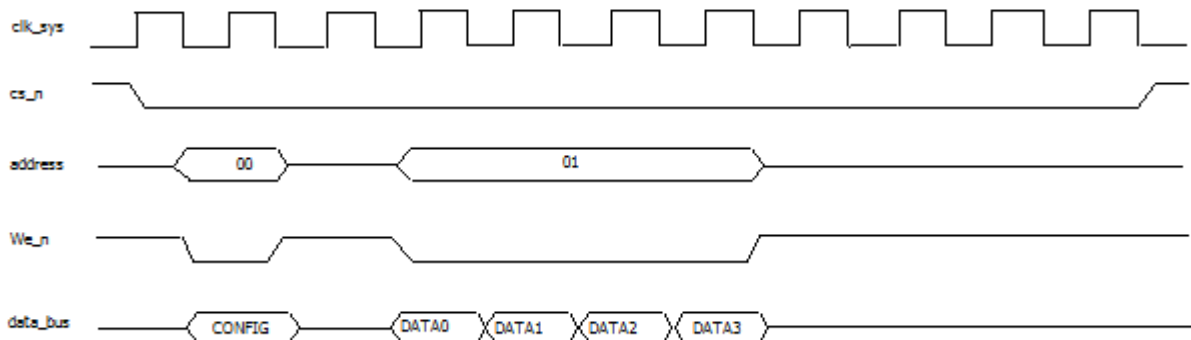
SiliconBlue is providing this document, design example, or information "as is". SiliconBlue does not ensure that this implementation or information is void of any claims of infringement. As a reader / implementer, you are responsible for obtaining any rights you may require for your implementation. SiliconBlue also does not warranty the fitness of this design example to be readily implemented in any specific context. It is your sole responsibility to validate this design example and its correctness while implementing it for your requirement.



**Fig 4. Timing Diagram: Read and Write on MOSI and MISO**



**Fig 5. Timing Diagram: Read status, configuration and data**



**Fig 6. Timing Diagram : Write configuration and data**