

CAST



SPI_MS Serial Peripheral Interface Master/Slave Core

The Serial Peripheral Interface (SPI) allows high-speed synchronous serial data transfers between microprocessors, microcontrollers and peripheral devices. The SPI_MS core implements the Serial Peripheral Interface, which can operate either as a master or as a slave.

When operating in master mode, the core generates the serial data clock (SCK) and selects the slave device, which will be addressed. When operating in slave mode, another master device generates the serial data clock and activates the slave select input of the core, in order to communicate.

The core was carefully designed to provide the most reliable communication possible, and to achieve very high bit rates.

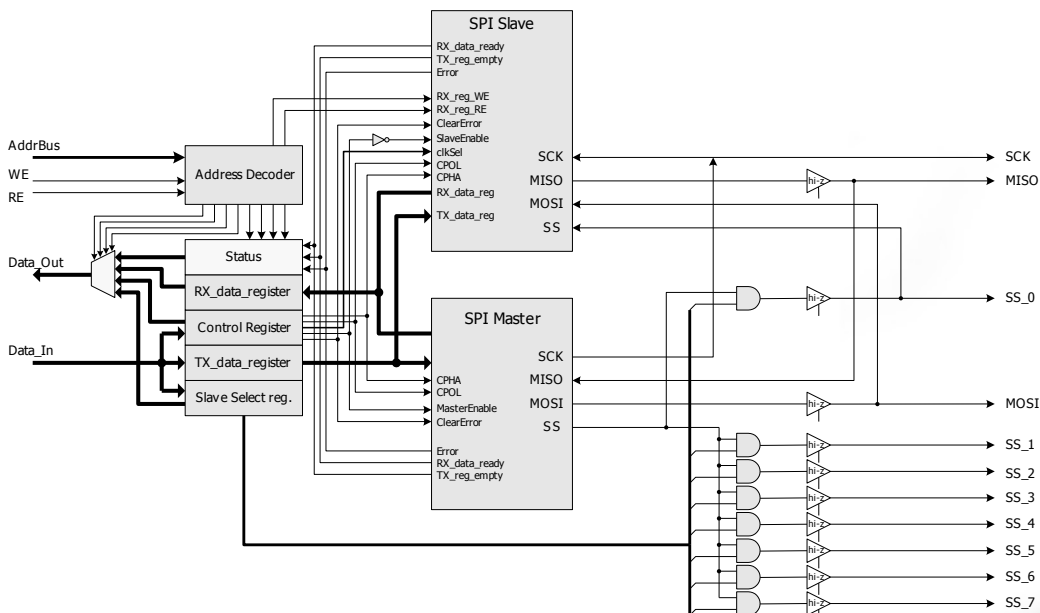
Developed for easy reuse, the SPI_MS is available optimized for several Lattice devices, with competitive utilization and performance characteristics.

Applications

The core is suitable for implementing serial interfaces in a wide range of applications, including:

- Embedded microprocessor boards and SOCs
- Consumer and professional audio/video
- Home and automotive radio

Block Diagram



Features

- Fully synchronous design with one clock domain
- Full duplex, synchronous, 8-bit serial data transfer
- High bit rates
- Master or slave mode
- Bit rates generated in Master mode:
 $\div 2, \div 4, \div 8, \div 16, \div 32, \div 64, \div 128, \div 256$ of the system clock
- Bit rates supported in Slave mode: $f_{SCK} \leq f_{SYSCLK} \div 2$
- Eight Slave Select lines
- MSB first or LSB first data transfer
- Post-synthesis EDIF netlist optimized for a specific Lattice device (HDL RTL source code also available)

Functional Description

The SPI slave was carefully designed to provide the most reliable communication possible. In order to avoid erroneous triggering of the internal shift register caused from noise or glitches on the serial data clock (SCK) line, the SCK line is sampled and synchronized with the system clock. Special design techniques were used to obtain a reception mechanism immune against metastability errors, which would cause reception faults when the serial data clock frequency is above 1/4 of the system clock frequency. Thus the reception mechanism of the SPI slave can operate reliably with a serial data clock frequency up to 1/2 of the system clock. This permits higher transfer rates when the core is used with a low frequency system clock.

The core was also designed to achieve very high bit rates. 22 Mbit/s were achieved in lab tests with the core implemented in an FPGA device and off-chip connection between master and slave. Higher bit rates are possible with faster devices and careful PCB design.

The design is fully synchronous and has one clock domain, the system clock. This leads to a more reliable and trouble-free synthesis and implementation of the core. No technology dependent features are used, so the VHDL code can be easily transferred to any technology.

Another feature incorporated in the core is the support for eight Slave Select lines used to access up to eight devices when working as a master. Also the ability to select the transfer order of the bits — MSB first or LSB first — saves valuable time by not implementing this function in software.

Implementation Results

The following are typical performance and utilization results using Lattice ispXPGA™ and ORCA™ devices.

Lattice Device	LUT-4s	Registers	PFUs	SysMEM EBRs	External I/Os	Speed (f _{max} , MHz)
LFX1200B-4	165	132	54	–	42	166
OR4E02-3	187	111	33	–	42	106

Lattice Device	LUT-4s	Registers	Slices	SysMEM EBRs	External I/Os	Speed (f _{max} , MHz)
LFE2-50-6	148	111	126	-	42	205
LFSC3GA25-6	143	111	95	-	42	250

Options and Modifications

The core can be easily customized to include receive/transmit FIFOs or specific microprocessor interfaces.

Support

The core as delivered is warranted against defects for ninety days from purchase. Thirty days of phone and email technical support are included, starting with the first interaction. Additional maintenance and support options are available.

Verification

The core has been verified through extensive simulation and rigorous code coverage measurements. It has also been successfully implemented in commercial and prototype systems.

Deliverables

The core includes everything required for successful implementation:

- Post-synthesis EDIF netlist (firm core) optimized for a specific Lattice device (HDL RTL source code (soft core) is also available)
- Sophisticated self-checking Testbench (Verilog versions use Verilog 2001)
- Simulation script, vectors, and expected results
- Synthesis (soft) or place and route (firm) script
- Comprehensive user documentation