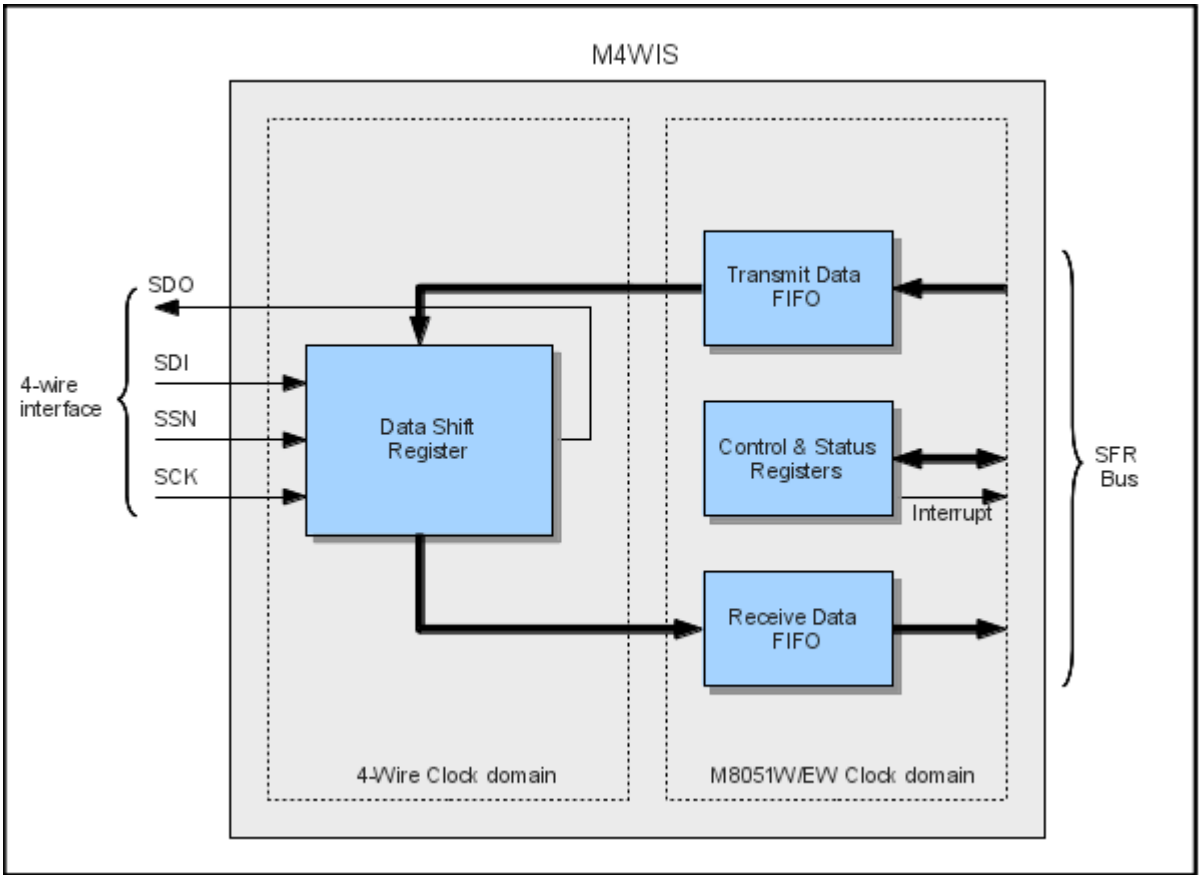
M4WIS
Four-Wire Slave Interface for M8051W and M8051EW Microcontrollers

Overview

The M4WIS adds four-wire slave capability to the M8051W and M8051EW designs. Use of standard synchronous design methodology makes this core simple to integrate into both ASIC SoC and FPGA designs.

M4WIS Four-Wire Slave Structure



- Implementation of a four-wire synchronous full-duplex slave interface, compatible with the Motorola SPI bus
- Supports various message protocols using byte granularity data
- Programmable clock phase and clock and select polarity
- Supports any four-wire clock rate, independent of microcontroller clock rate
- Integral transmit and receive data FIFOs minimise the processor overhead required to service the link
- Supports both interrupt-driven and polled transfers
- Optional clock prescaler minimises power consumption while active
- Rests in power saving mode when the interface is not enabled
- Slave select wake up feature can be used to cold start a M8051W and M8051EW microcontroller, in addition to interrupt driven wake ups
- Binds tightly to M8051W and M8051EW external interrupt and SFR buses with no additional glue logic required

M4WIS
Four-Wire Slave Interface for M8051W and M8051EW Microcontrollers

Supported Protocols

The M4WIS is compatible with the SPI serial bus protocol. Clock phase and polarity as well as select polarity are controlled by software.

Interfacing

The M4WIS is design to bind tightly to an M8051W or M8051EW SFR bus. Data, control and status information are exchanged with a host microcontroller using memory-mapped 8-bit words. A single interrupt signal is used to indicate that a matching address has been received, that data buffers require service or that an exception has been detected.

Structure

The M4WIS is comprised of two independent clock domains: The four-wire clock domain and the host peripheral clock domain.

An eight-bit shift register, clocked by the external four-wire clock, transfers data octets in and out of the core. Data for transmission over the two-wire interface is first written by the M8051W and M8051EW into the M4WIS transmit FIFO that is located in the SFR address space. When the external four-wire master executes a read transfer, data from the transmit FIFO is shifted out onto the two-wire interface. When the external four-wire master executes a write transfer, the data received by the M4WIS shift register is written to a receive FIFO. Data in the receive FIFO is read by the host M8051W and M8051EW via the SFR address space.

Power Management

The M4WIS is compatible with the standard M8051W and M8051EW three power saving states. In the M8051W and M8051EW idle state the M4WIS continues to function, transferring data in or out of data FIFOs and generating an interrupt when a transfer is complete or a FIFO requires attention.

A clock prescaler enables designs to select the minimum clock frequency for correct operation of M4WIS logic. This may be many times slower than the host microcontroller clock frequency.

When the M4WIS is not enabled, and there is no activity on the four-wire interface, power consumption drops to leakage levels only.

The M4WIS can be configured to generate a host wake-up signal whenever the Slave Select (SS) is detected. This enables the host logic to be left in a deep power saving state between four-wire messages.

Configuration Options

The core RTL is highly configurable at compile time allowing users to implement only the features required by their application.

Major configuration options include:

- Default clock polarity and phase parameters
- Default select polarity
- Data FIFO depths from 1 to 255 bytes
- Single clock design, or independent four-wire and microcontroller clock domains
- User defined SFR base address
- User defined interrupt channel
- Microcontroller clock domain prescaler
- Optional microcontroller wake-up signal

Deliverables

- VHDL '93 and Verilog 2001 RTL source code
- VHDL and Verilog functional demonstration testbench
- Demonstration assembler code
- Simulation scripts for Modelsim and Cadence
- Synopsys synthesis compile scripts and SDC timing constraint files
- Example Mentor DFT and ATPG scripts
- Example netlist implementation with SDF files
- Detailed product specification and a user guide containing implementation notes