The SPI slave protocol

For using the Netzer as Ethernet modem from i.e. another microcontroller there is a dedicated Netzer project “SPISlave”. This project implements an SPI slave mode. TCP/IP connections can be established and released (as client or server) and data can be received and sent.



Since version 1.4 the protocol and interface has significantly changed compared with prior versions. For older versions please check out the Elektor website.

Physical connection

At least six free pins on the master side are needed. See figure 1 diagramm for a typical connection.

All SPI pins are read only for the GPIO access (GPIO webpage). Other pins can be used as digital in or ouputs as needed.

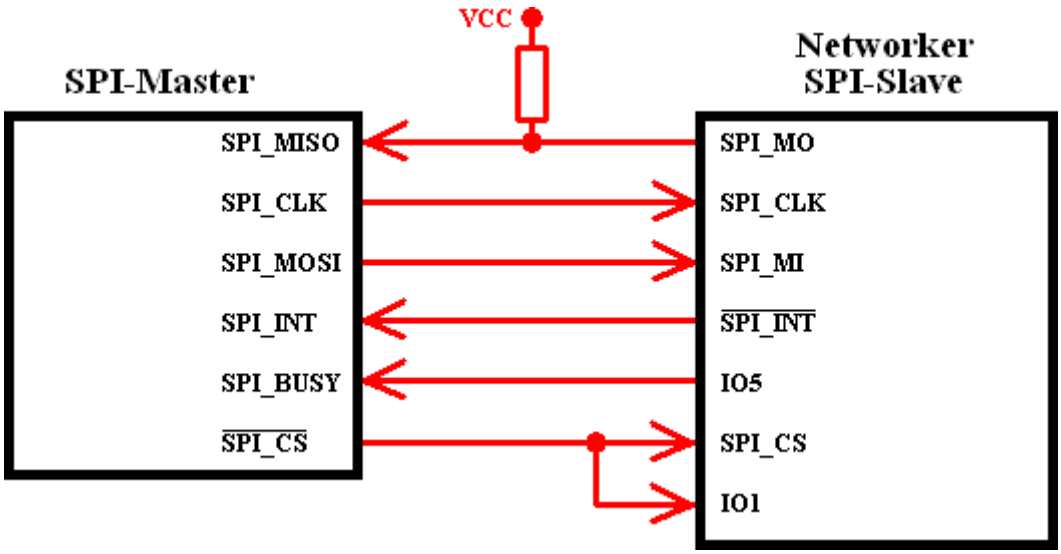


Fig. 1: Connection example

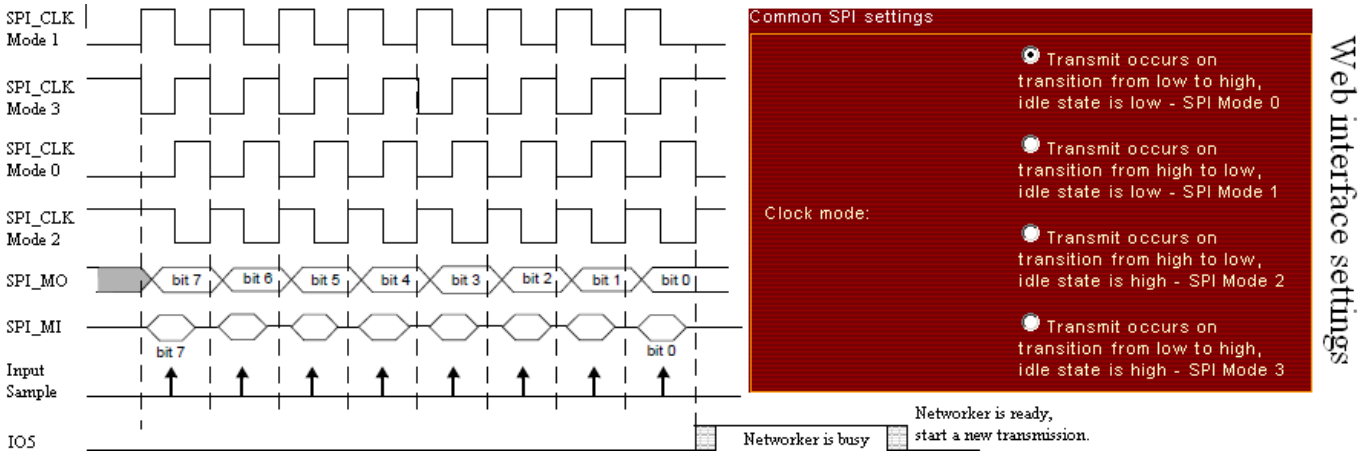


Fig. 2: Waveforms

Netzer pin	Direction	Description
SPI_CLK	Input	The master generates edges for data transmission here. One tranfer unit is eight edges and therefore eight bits. All clock polarities are supported by Netzer (settable via web interface). **Important Note:** The selected idle state on this pin is checked by Netzer on startup to ensure synchronicity with SPI master. Netzer blocks as long as the pin level does not match the idle state!
SPI_MO	Output / Open	Netzer shifts out its data on SPI_CLK edges.
SPI_MI	Input	Master shifts out its data on SPI_CLK edges.
SPI_CS	Input	The master **MUST** select Netzer with a low chip select line. During the entire transmission the chip select must stay low. Note: Releasing the chip select does not automatically reset the register interface (see below). A released chip select disables the internal SPI_MO driver. A pullup or pulldown resistor is needed therefore on this pin to prevent floating!
SPI_INT	Output	Netzer pulls this pin low, if internal states has changed. The master MUST check this line inbetween complete transmissions of eight bit. If the line is low the master MUST stop any pending register access immediately with a rising edge on IO1. The interrupt pin is released after a SPI master write to the interrupts flags register.
IO5	Output	IO5 is used as handshake pin for increasing data throughput. The pin is low during master transmissions. On finished master transmissions the pin is driven high as soon as the Netzer internal SPI module has detected pending data. During the processing of the received byte the line stays high. If processing is finished and possible data is provided by the slave for shifting out the line returns to low. The master is not allowed to start a new transmission during the high state of IO5.
IO1	Input	IO1 is used for controlling the register access. A falling edge on IO1 starts a new register access. Netzer awaits the very next byte from master after the edge as register address. A rising edge on the pin stops the register access. For easier connection the SPI_CS pin and IO1 can connected together.

From:
http://www.mobacon.de/dokuwiki/ - **MoBaCon**

Permanent link:
http://www.mobacon.de/dokuwiki/doku.php?id=en:netzer:spi-slave

Last update: **2025/06/11 20:42**