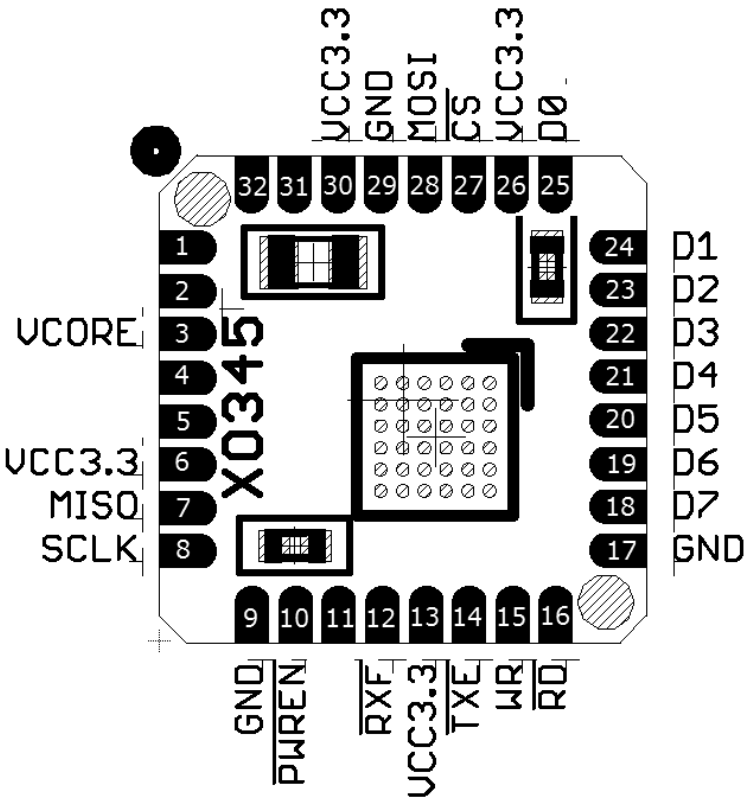
XO345 SPI-Parallel Interface

FT245BL pin compatible interface controller to use USB-enabled devices with SPI



Features:

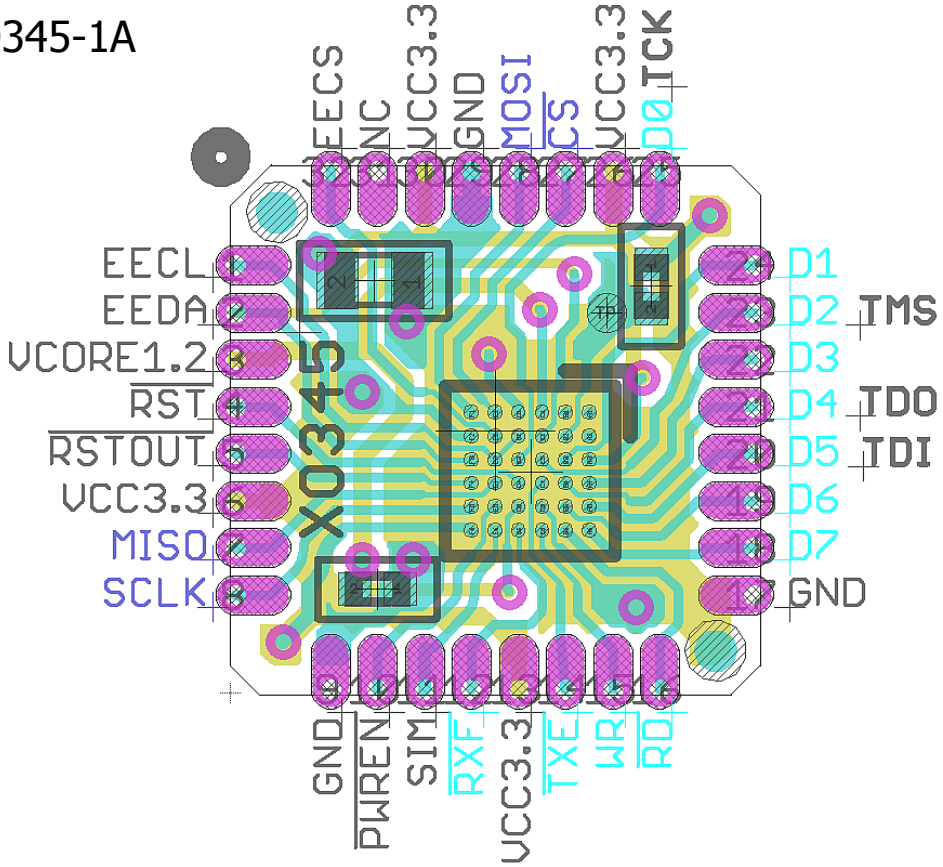
-QFP-32 0.8mm form factor
-drop-in replacement for FT245BL device
- on-board bypass caps
- multiple 3.3V pads
- SPI and VCORE 1.2V routed to D+,D-,XIN,XOUT and VUSB pads
- no bulky programming header
- 4-wire SPI Mode 0
up to 33 MHz
- EEPROM and control pins connected (dev option)
- deep 256 Byte RX and
256 Byte TX buffers
- low power consumption

Description:

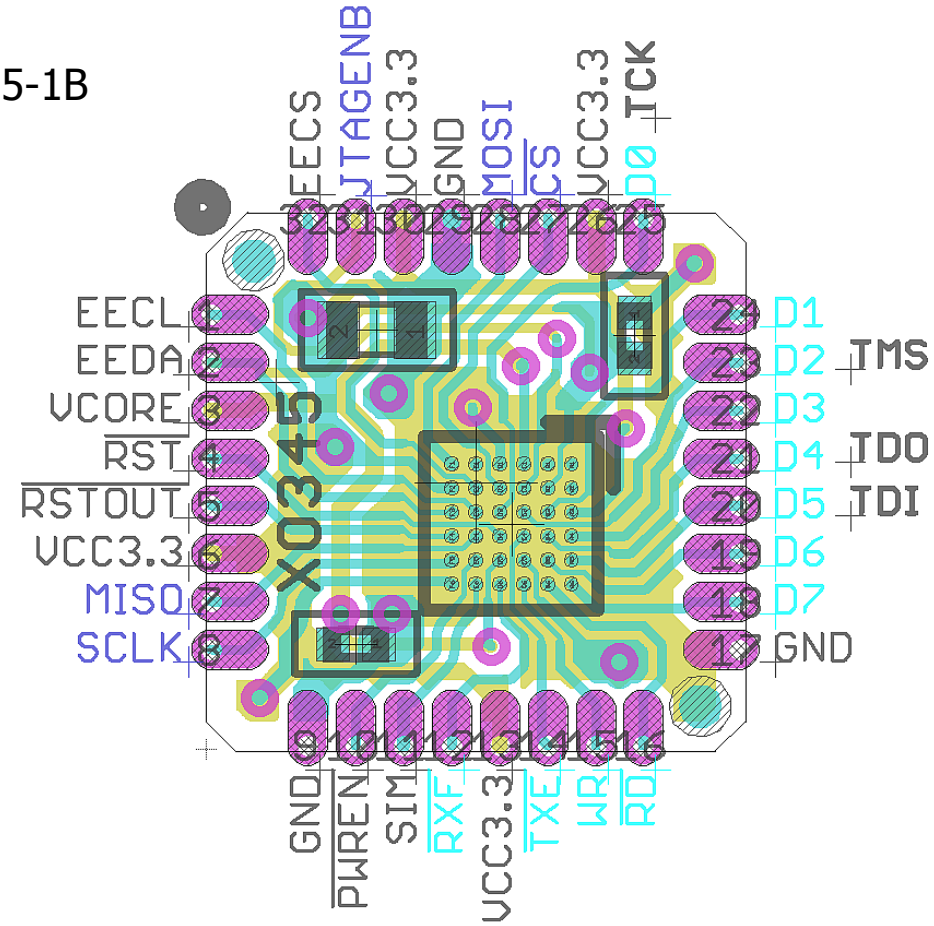
The XO345 is an FPGA-based FIFO interface hybrid designed as a drop-in replacement for the FTDI FT245BL chip. It has a „245" compatible parallel interface to the target board side and provides a 4-wire MODE 0 MSB-First slave SPI as an external interface.
SPI signals and VCORE are routed to easily accessible USB and XTAL pins. While VCCIO 3.3V is supplied via pad 13, only VCORE 1.2V needs to be supplied via the existing 5V VUSB line (USB powered implementation).
SPI signals are 3.3V.
Transmit and Receive buffers are 256 Byte wide, read and write timings are within 25ns of the original specs and compliant.

Pinouts (Rev 1A, 1B)

XO345-1A

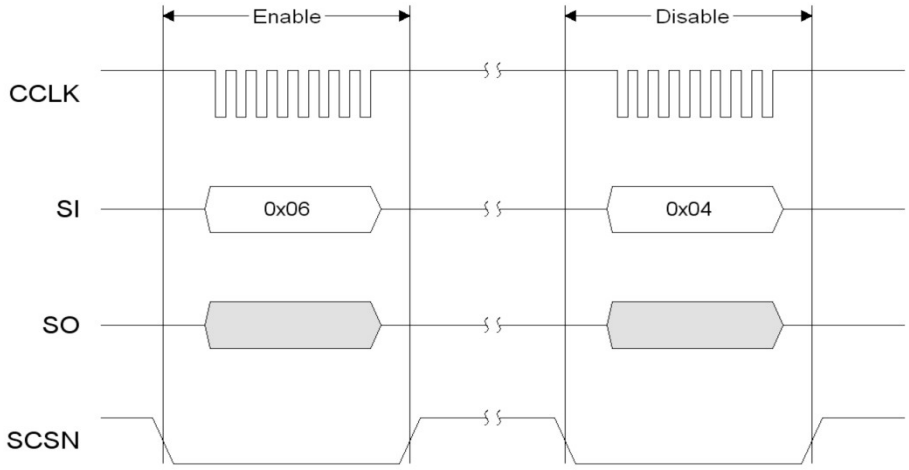


XO345-1B

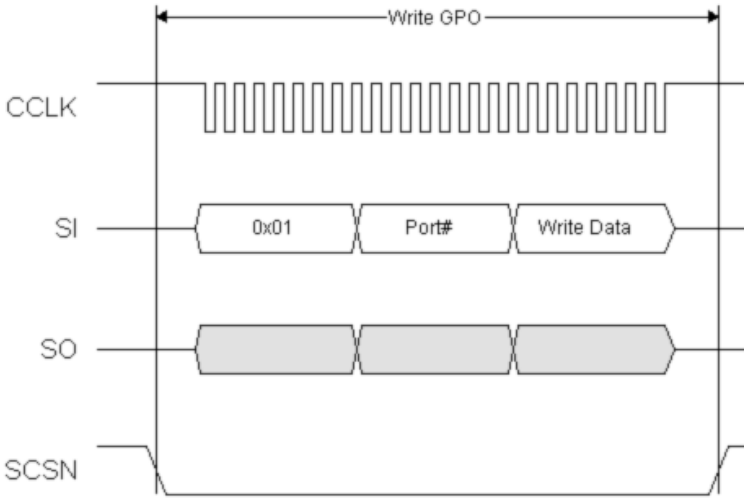


Command Set

!PWREN flag [default: disabled (high)]



GPO_0 control register [default: 0x00]



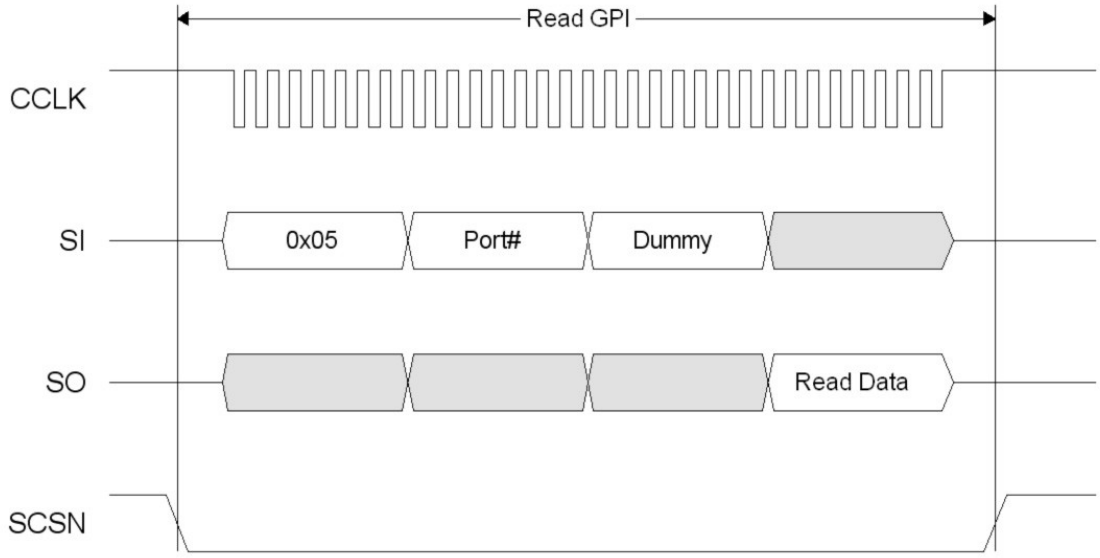
7	6	5	4	3	2	1	0
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	RXFN_STATE	RXFN_MODE

RXFN_MODE: when 0, !RXF is inhibited during write actions to the RX buffer. This simulates the bundling of bytes in a USB frame. After the write action !RXF is controlled by the level of the RX FIFO and the reaction to RDN pulses.

RXFN_STATE: (RXFN_STATE, RXFN_MODE) = 11: !RFX permanently inhibited, 01: !RXF immediately derived from the FIFO fill level and response to !RD pulses, x0: no effect

GPI_0 FIFO status register [default: 0x33]

Port#= 0x00



7	6	5	4	3	2	1	0
RX_FULL	RX_HI	RX_LO	RX_EMPTY	TX_FULL	TX_HI	TX_LO	TX_EMPTY

RX_FULL: RX FIFO is full at 256 Bytes (no more Bytes can be written)

RX_HI: RX FIFO is almost full (>=240 Bytes)

RX_LO: RX FIFO is not significantly filled (<= 16 Bytes)

RX_EMPTY: RX FIFO is completely empty at 0 Bytes

RX_FULL: TX FIFO is full at 256 Bytes (no more Bytes can be written)

RX_HI: TX FIFO is almost full (>=240 Bytes)

RX_LO: TX FIFO is not significantly filled (<= 16 Bytes)

RX_EMPTY: TX FIFO is completely empty at 0 Bytes

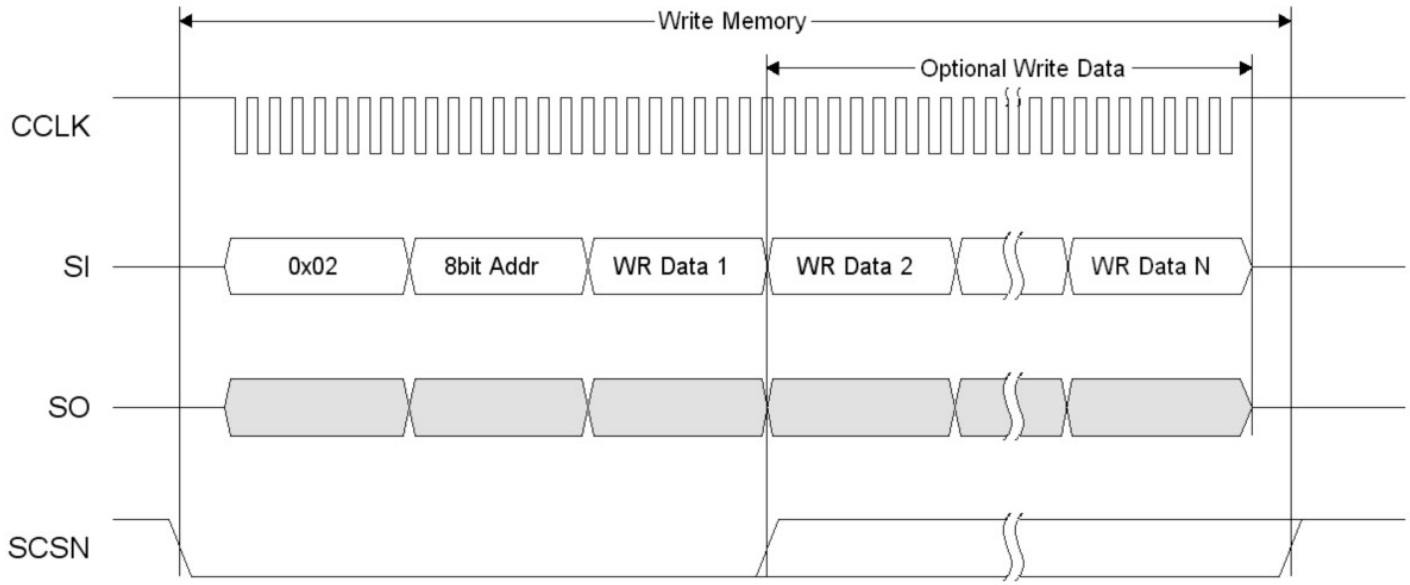
GPI_1 status register [reserved]

Port#= 0x01

7	6	5	4	3	2	1	0
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

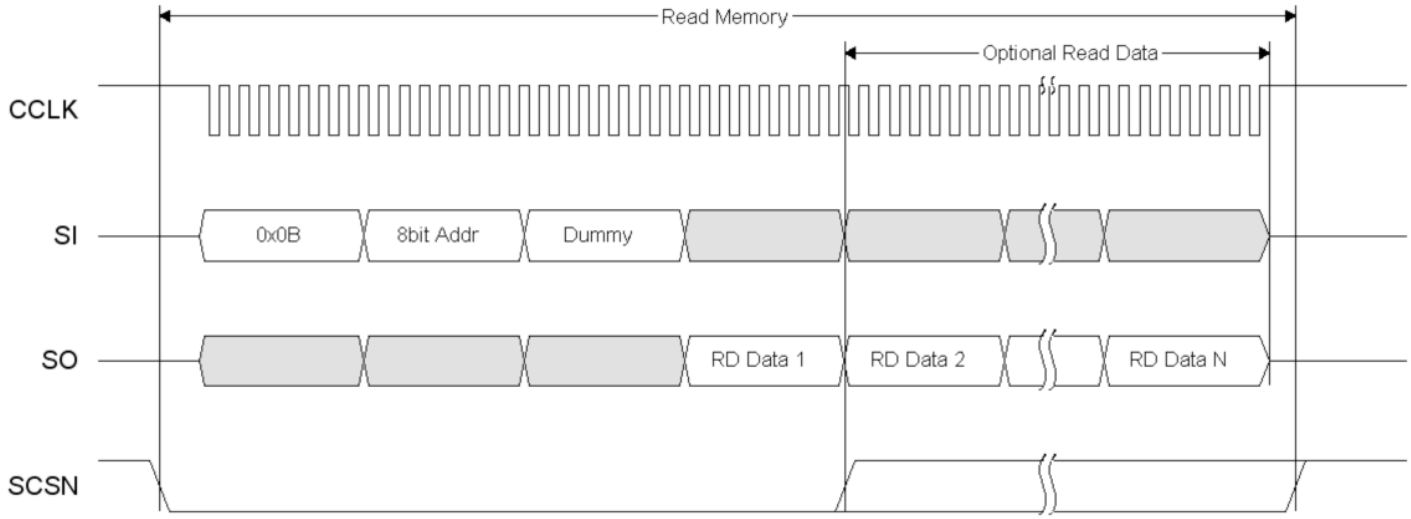
Writing to the RX FIFO

8bit Addr = 0x00 (ignored)



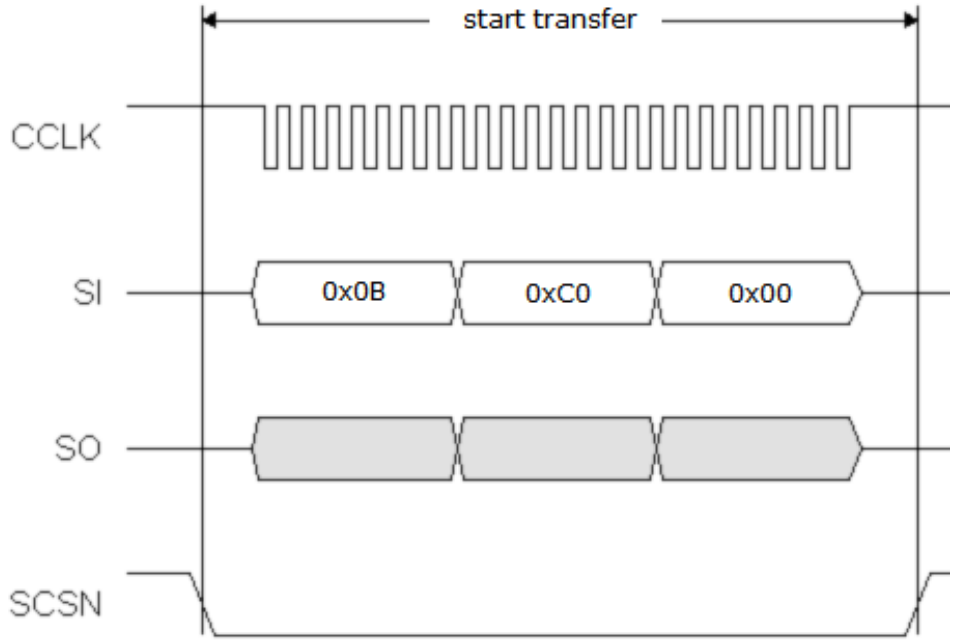
since WR Data 0..N is directly transferred to the RX FIFO, the write address is ignored. Up to 256 Bytes can be written at a time and the FIFO Status register needs to be checked to see how many bytes are still available.

Reading from the TX FIFO



Reading is done in two steps: a TX FIFO -> OUTPUT BUFFER transfer is issued. After completion, addresses 0x00 .. 0x3F can be read in a random and repeated way.

Initiating TX FIFO -> OUTPUT BUFFER transfer



The transfer process is started by reading from address 0xC0. It completes within 2.5µs.

Reading 0x00 yields a status byte of the OUTPUT BUFFER memory. Data can be read from addresses 0x01 .. 0x3F.

MEM(0x00): when 0x00, no valid data is inthe OUTPUT BUFFER memory.

MEM(0x00): when 0xFF, the transfer is ongoing.

MEM(0x00): when 0x01.. 0x3F, that number of Bytes can be read starting at 0x01.

Device Programming

XO345 is based on the Lattice MachXO3-1300E and is programmed via JTAG. The JTAG interface is mapped to the PARALLEL pins and accessible when the TP is grounded (Rev 1B: JTAGENB pad grounded).

During programming !TXEmpty and !RXFilled must be high level to disallow target system interference with D0..D7.