

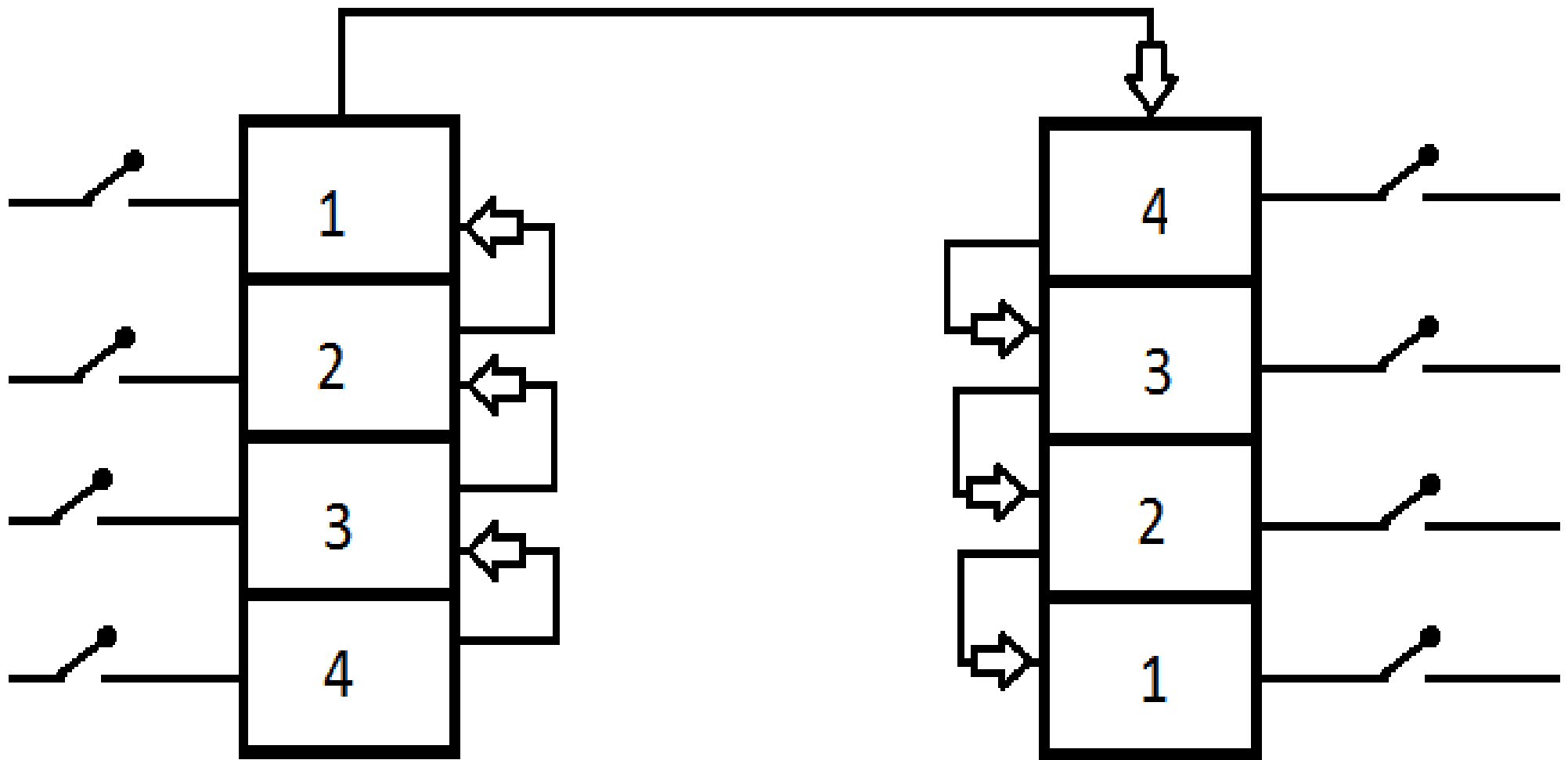
Serielle BUS-Systeme SPI und UART

Gliederung

- Shiftregister
- SPI
 - Grundlagen
 - Konfigurationen
 - Timing
 - Vor- / Nachteile
- UART
 - Nötige Voreinstellungen
 - Datenstrom
- Quellen

„Parallel“ zu Seriell

Shiftregister



SPI

Serial Peripheral Interface

Seriell

Synchron

Entwickelt von Motorola

SPI Pins

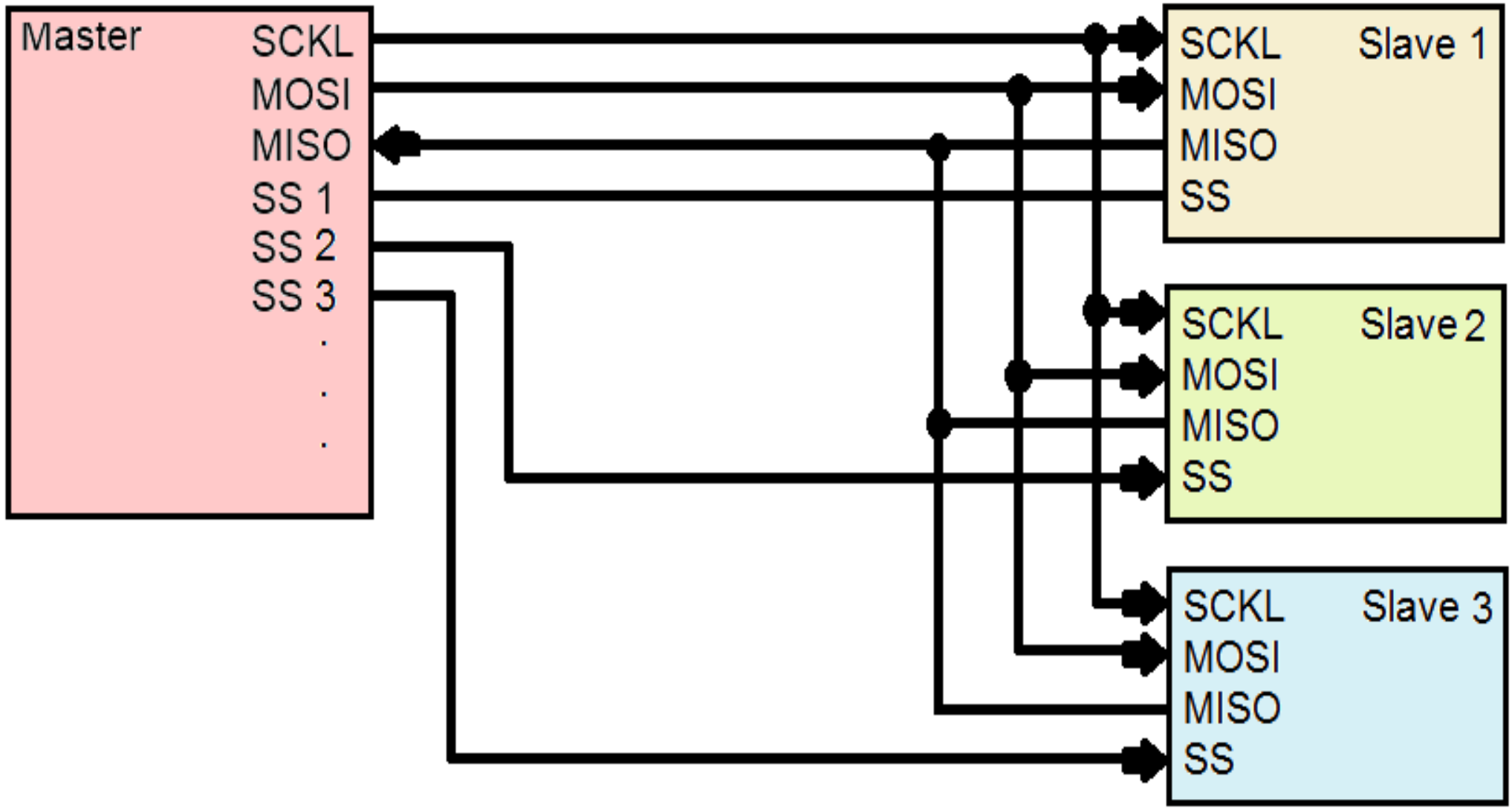
SCKL Serial Clock

MOSI Master out, Slave in (ausgehende Datenleitung)

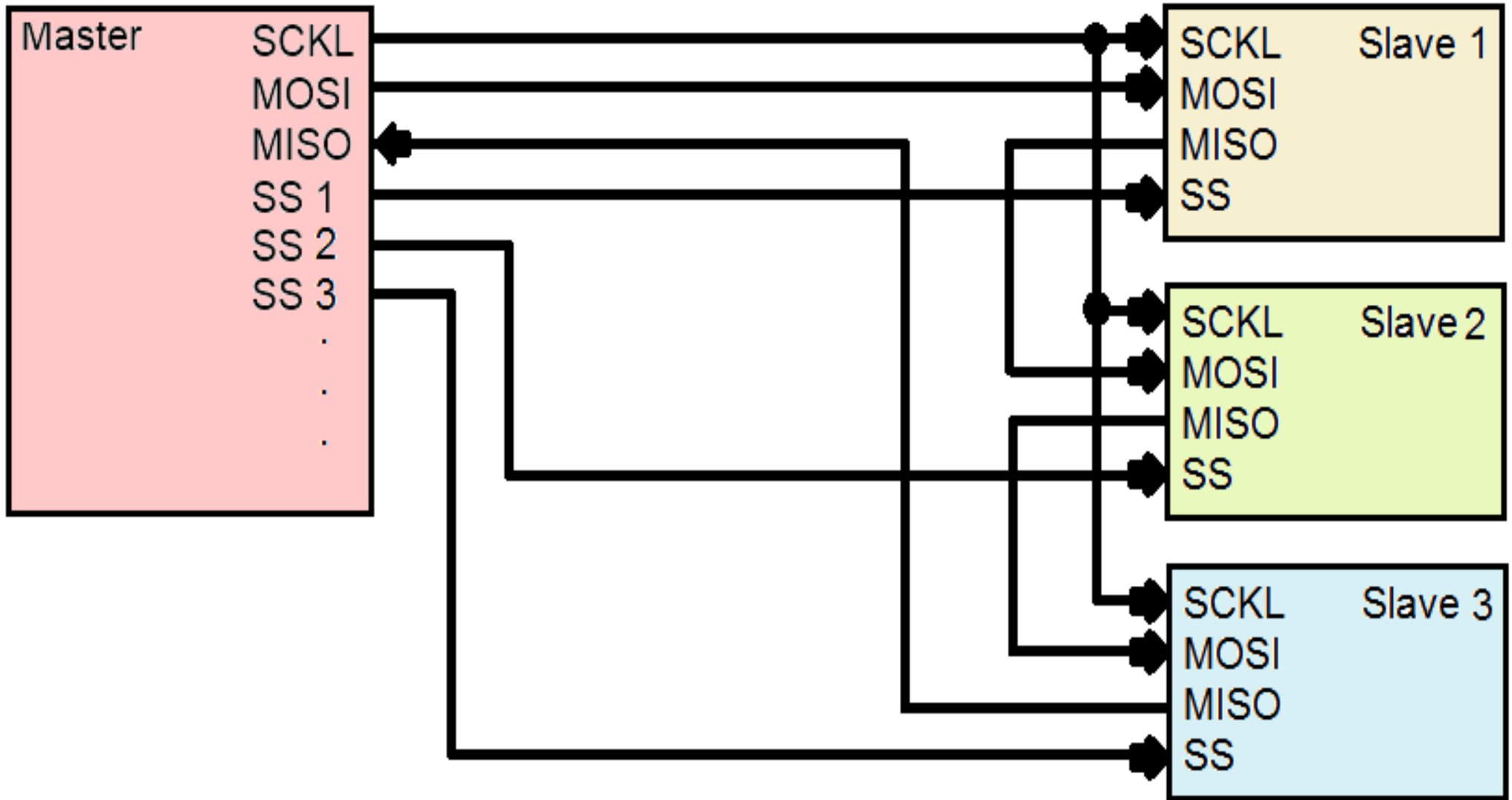
MISO Master in, Slave out (eingehende Datenleitung)

SS Slave Select

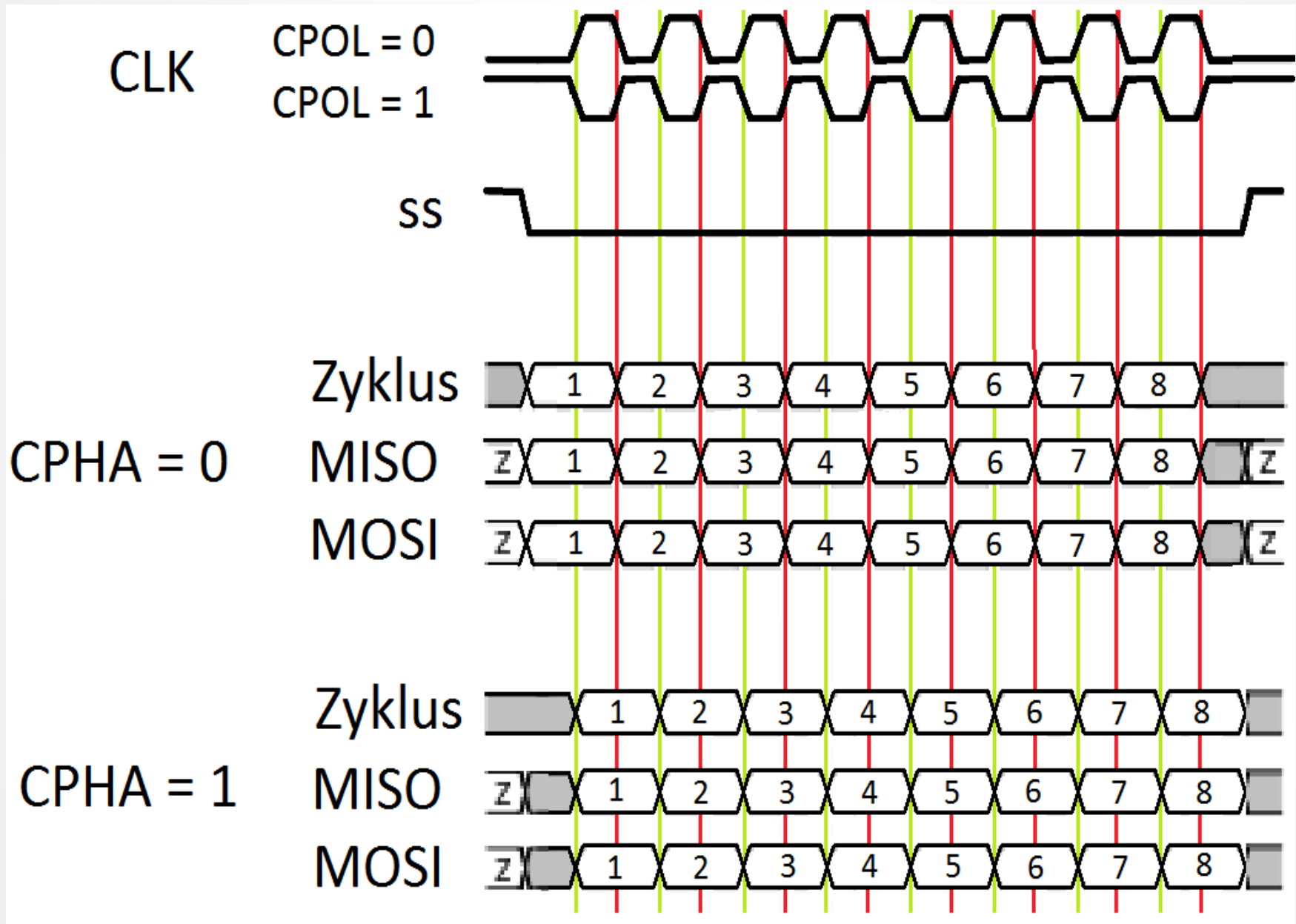
Independent Slave



Daisy Chain SPI



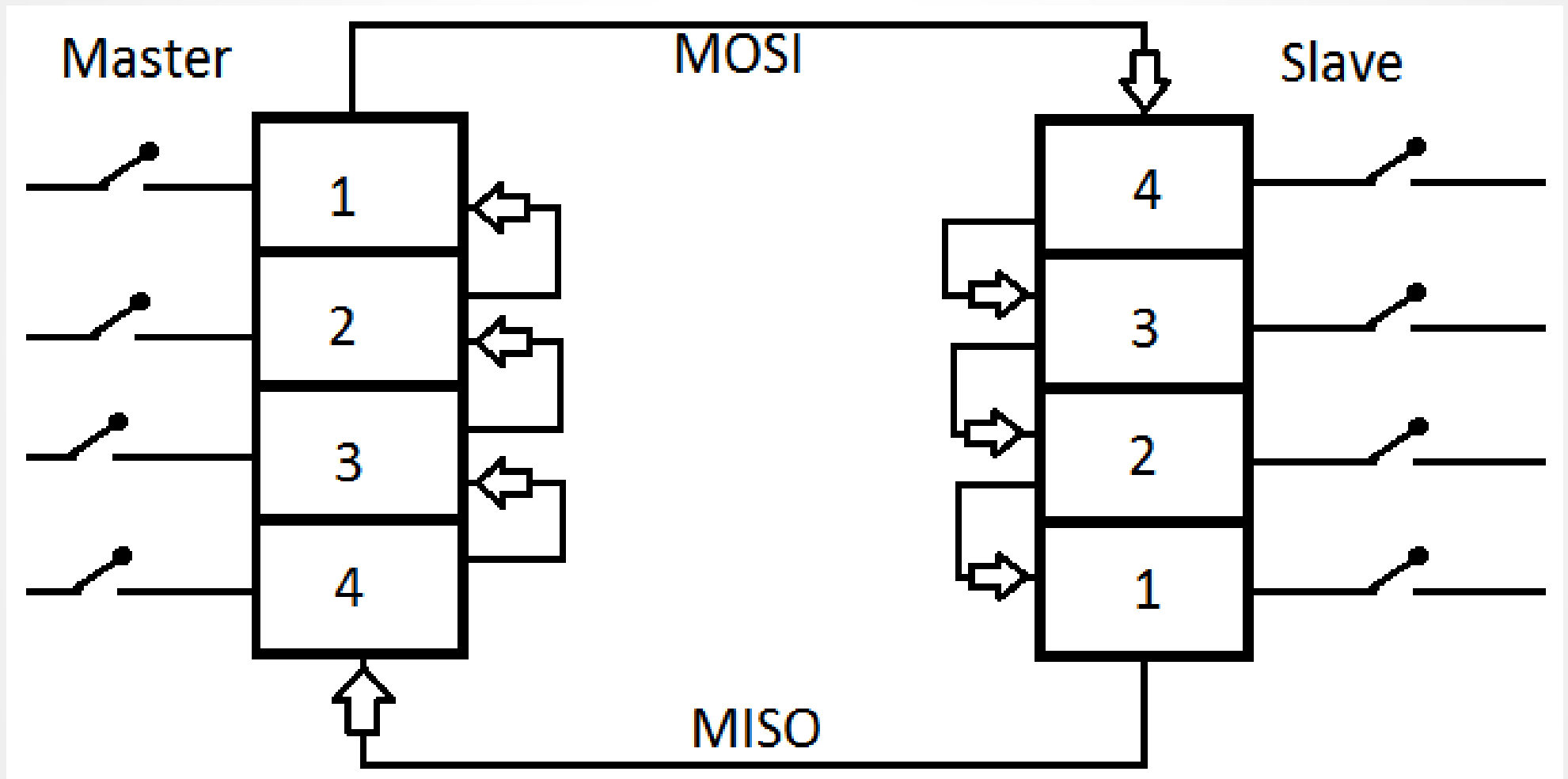
Timing



Mode Tabelle

Mode	CPOL	CPHA
0	0	0
1	0	1
2	1	0
3	1	1

Vollduplex



Vorteile

- Vollduplexfähig
- Protokoll ist flexibel
- einfache Hardware
- Belegt wenige Pins

Nachteile

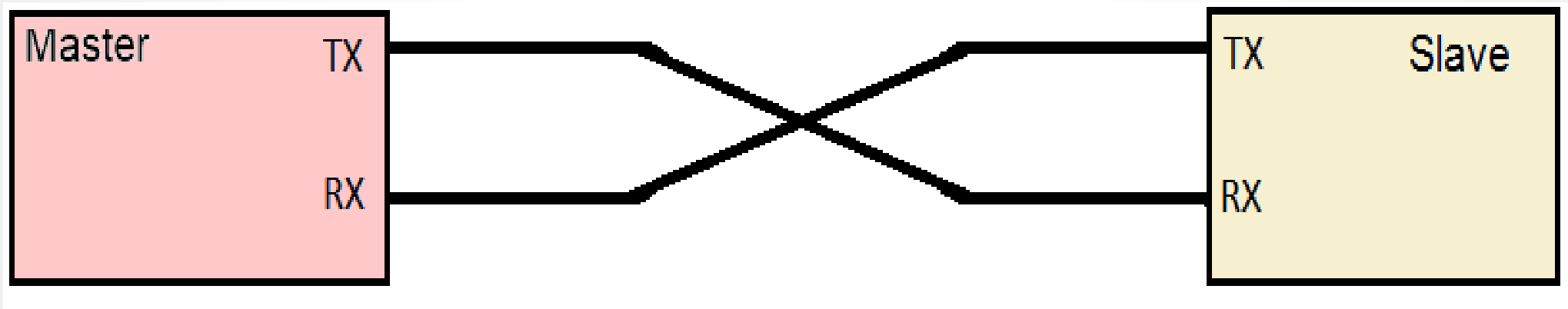
- „extra“ Slave Select nötig
- keine Slave Erkennung
- nur ein Master möglich
- kein Hot Plugin

UART

Seriell

Asynchron

Pinbelegung



UART Konfiguration

- Bitzahl pro Wort (5 – 9 Bits)
- Bitreihenfolge
- Bitgeschwindigkeit
- Verwendung von Checkbit

Baudrate einstellen

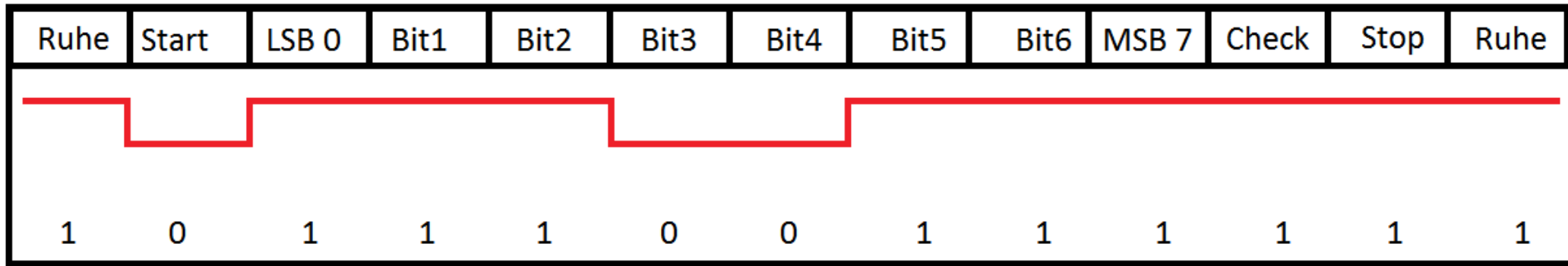
$$UBRRL/UBRRH = \frac{\textit{Taktfrequenz}}{16} \times \textit{Baudrate}$$

Code Beispiel

Senden

```
1  sbi    UCSRB, TXEN           ; TX aktivieren
2
3  ldi    zeichen, 'x'
4  call   serout               ; Unterprogramm aufrufen
5  ldi    zeichen, 'y'
6  rcall  serout               ; Unterprogramm aufrufen
7  ldi    zeichen, 'z'
8  rcall  serout
9
10 serout:
11 sbis   UCSRA, UDRE          ; Warten bis UDR für das nächste Byte bereit ist
12 rjmp   serout
13 out    UDR, zeichen
14 ret                                ; zurück zum Hauptprogramm
```

Datenstrom



Vorteile

- 2 x Halbduplex
- Protokoll ist flexibel (weniger als SPI)
- einfache Hardware
- belegt wenige Pins (weniger als SPI)
- Handshake leicht zu implementieren

Nachteile

- genauer Taktgeber nötig

Quellen

- <http://de.wikipedia.org/wiki/UART>
- http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus
- http://www.layadcircuits.com/layad_articles/UART_Basics.htm
- http://www.mikrocontroller.net/articles/AVR-Tutorial:_UART