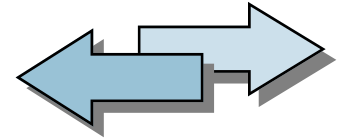


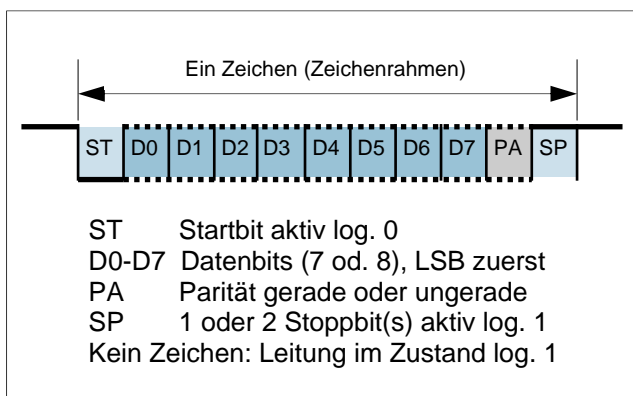
RS-232

Asynchrone serielle Schnittstelle



Asynchrone Datenübertragung

Bei der seriellen, asynchronen Datenübertragung nach RS232 wird die Information zur Synchronisierung jedem einzelnen Zeichen mitgegeben. Ein Startbit (log. 0) leitet die Zeichenübertragung ein. Danach folgt die eigentliche Information in Form von 7 oder 8 Datenbits. Dabei wird das LSB zuerst übertragen. Das Ende einer Zeichenübertragung wird durch ein oder zwei Stoppbits (log. 1) gekennzeichnet. Bei Bedarf kann nach den Datenbits zur Fehlererkennung noch ein Paritätsbit eingefügt werden. Für die Übertragung sind verschiedenen Geschwindigkeiten genormt. Werden keine Zeichen übertragen, so ist die Leitung im logischen Zustand 1 (Mark).



Beispiel: Bitfolge bei der seriellen Übertragung

Paritätsprüfung

Mit der Paritätsprüfung kann darauf geschlossen werden, ob bei der Übertragung ein Fehler aufgetreten ist. Es ist jedoch keine Fehlerkorrektur möglich. Die Prüfung erfolgt eines Paritätsbits das aus der Quersumme der log. 1 Zustände der Datenbits berechnet wird. Man unterscheidet zwischen gerader und ungerader Parität.

Parität	Quersumme der 1	Zustand PA
gerade	gerade	0
gerade	ungerade	1
ungerade	gerade	1
ungerade	ungerade	0

Es kann nur eine ungerade Fehleranzahl erkannt werden (1, 3, 5, usw.). Paritätsprüfung wird daher eher selten angewendet.

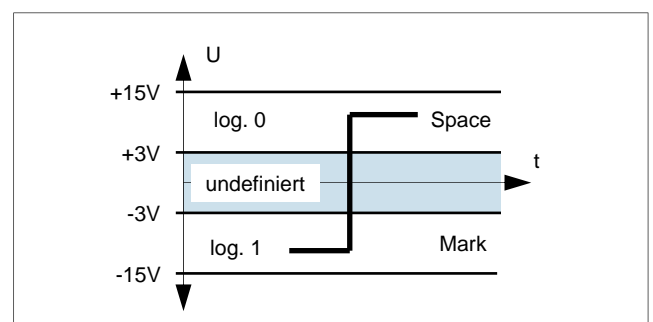
Übertragungsgeschwindigkeit

Zur Definition wird die Einheit Baud [Bd] verwendet. Sie gibt an, wie viele Bits/s übertragen werden. Falls die Übertragung mit $v = 9600\text{Bd}$, 8 Datenbits erfolgt und ein Start- und ein Stoppbit einbezogen werden, braucht es zur Übertragung von 1 Zeichen $10\text{Bit} = n$. Die Anzahl Zeichen pro Sekunde v_z ergibt sich aus: $v_z = v / n = 9600\text{Bd} / 10\text{Bit} = 960\text{Zeichen/s}$

Übliche Werte für Baudraten sind: 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400 [Bd] usw.

Eigenschaften nach RS232 Standard

Nach dem RS232 Standard gelten für die Spannungsschnittstelle folgende Eigenschaften:



Signalpegel der RS232 Schnittstelle

- Maximal darf U an der Last $\pm 25\text{V}$ betragen
- Kurzschlussfest I_k maximal 500mA
- Maximale Leitungslänge 15m
- 9 oder 25p SubD Stecker. DTE (m), DCE (w)

Steckerbelegungen für 9 - und 25 polige Stecker mit der entsprechenden Datenflussrichtung zwischen DCE (Modem) und DTE (PC).

Leitung	Bez.	9p	25p	DCE - DTE
Transmitted Data	TxD	3	2	←
Recived Data	RxD	2	3	→
Request to Send	RTS	7	4	←
Clear to Send	CTS	8	5	→
DataSet Ready	DSR	6	6	→
Signal Ground	GND	5	7	—
Line Signal Detector	DCD	1	8	→
Data Terminal ready	DTR	4	20	←
Ring Indicator	RI	9	22	→

I2C

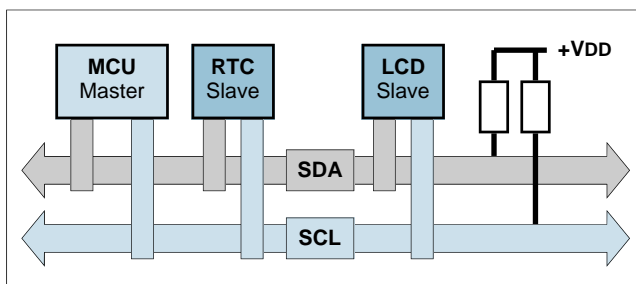
Inter - IC - BUS



I2C Datenübertragung

Philips entwickelte diese serielle bidirektionale 2-Draht Schnittstelle für die Kommunikation zwischen verschiedenen IC's in elektronischen Systemen. Das Interface besteht aus den beiden Signalleitungen SDA (serial data) und SCL (serial clock) sowie einem gemeinsamen GND (ground).

Am Bus können mehrere Teilnehmer angeschlossen werden. Jeder der Teilnehmer hat eine spezielle 7Bit Adresse. In den meisten Fällen kontrolliert ein Master die verschiedenen Slaves. Es ist aber auch möglich, mehrere Master an einen I2C Bus anzuschliessen. Ein Teilnehmer kann ein Nur-Empfänger



Prinzip I2C Bus

(LCD) oder ein Sender und Empfänger (RTC) sein. Der Master kontrolliert die Datenflussrichtung und gibt mit dem Clock-Signal die Übertragungsgeschwindigkeit vor.

Die SDA und die SCL Leitung werden mit PullUp-Widerständen versehen. Die Bus-Teilnehmer sind mit Open Drain Schaltungen an den Leitungen angeschlossen und ziehen im aktiven Zustand die Leitungen gegen GND.

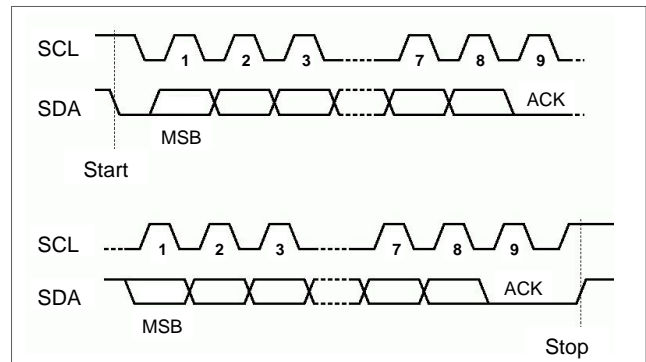
I2C Bus Protokoll

Der Bus-Master initialisiert eine Schrieb- oder Lese-sequenz mit einem bestimmten Teilnehmer. Jede Übertragungssequenz beginnt mit einer Start-Bedingung (SDA vor SCL auf 0) und endet mit einer Stop-Bedingung (SCL vor SDA auf 1).

Nach einem Start folgt ein Byte mit der 7Bit Teilnehmer-Adresse (b1-b7) und dem RW' Bit (b0), welches

7Bit Teilnehmer-Adresse							RW'
b7	b6	b5	b4	b3	b2	b1	b0

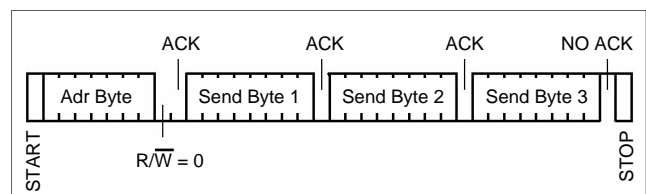
angibt, ob zum Teilnehmer geschrieben wird (b0=0) oder vom Teilnehmer gelesen werden soll (b0=1).



Start, Stop und Acknowledge

Master sendet Daten zu einem Slave

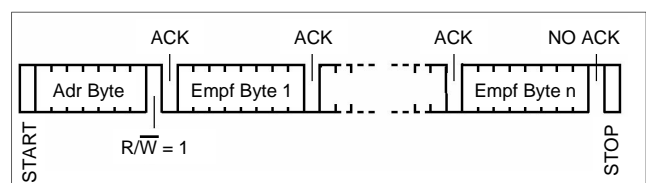
Mit einer Start-Bedingung beginnt die Schreibsequenz. Der Master gibt das Adressbyte mit dem RW' Bit =0 aus. Der Slave antwortet mit ACK (er zieht während Bit 9 SDA auf 0). Anschliessend sendet der Master ein oder mehrere Datenbytes. Nach jedem Datenbyte gibt der Slave ein ACK. Die Übertragung endet mit einer Stop-Bedingung.



Master sendet Adressbyte und Daten zu einem Slave

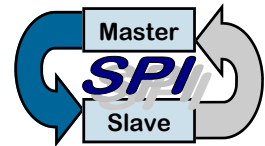
Master empfängt Daten von einem Slave

Eine Start-Bedingung leitet eine Lesesequenz ein. Der Master gibt das Adressbyte mit dem RW' Bit =1 aus. Der Slave antwortet mit ACK (er zieht während Bit 9 SDA auf 0). Anschliessend sendet der Slave ein oder mehrere Datenbytes. Nach jedem empfangenen Byte gibt der Master ein ACK. Die Übertragung endet mit einer Stop-Bedingung.



Master sendet Adressbyte und liest Daten von einem Slave

SPI Serial Peripheral Interface

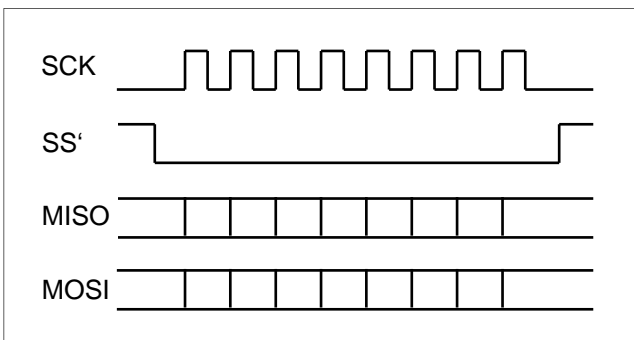


Übertragungsprinzip

Über die SPI Schnittstelle werden Daten synchron, seriell ausgegeben und gleichzeitig eingelesen. Ähnlich wie bei der I2C Schnittstelle kommuniziert ein Master (Microcontroller MCU) mit Peripherieschaltungen. Es ist jedoch auch möglich, dass über das SPI Daten zwischen zwei MCUs ausgetauscht werden. Diese Schnittstelle lässt auf kurzen Übertragungswegen hohe Übertragungsgeschwindigkeiten zu. Sender und Empfänger für die SPI Übertragung lassen sich mit geringem Aufwand (Schieberegister SR) aufbauen.

SPI Signale

Beim SPI wird zwischen Master (MCU) und Slave (Peripherieschaltung) unterschieden. Bei Datenübertragung zwischen zwei MCUs muss eine MCU auch als Slave geschaltet werden. Der Master startet eine Übertragung indem die SlaveSelect SS' Leitung auf den Zustand '0' gelegt wird. Anschliessend gibt der Master auf der Leitung des SystemClocks SCK acht Impulse aus. Auf der Datenleitung 'Master



Prinzip der SPI Signale

Out Slave In' MOSI gibt der Master bei jedem einzelnen Clocksignal ein Bit des zu übertragenden Bytes aus. Gleichzeitig liest der Master bei jedem Clocksignal ein Bit von der Leitung 'Master In Slave Out' ein. Die Slaveschaltung wird dabei vollumfänglich durch den Master gesteuert.

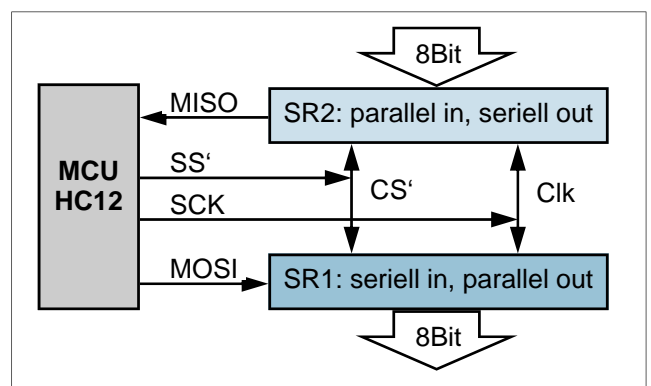
Übertragungsparameter

Für die Übertragungsparameter gibt es keine Normen. So kann die Übertragungsgeschwindigkeit beliebig gewählt und die Bitreihenfolge (MSB first, LSB first) kann je nach Anwendung eingestellt werden. Für die Synchronisierung der Daten mit dem Clock-

Signal sind mehrere Möglichkeiten vorgesehen (pos, neg Flanke, Polarität usw.). Bei der HC12 MCU sind alle diese Parameter programmierbar.

Prinzip einer Slaveschaltung

Eine einfache IO-Schaltung kann mit zwei Schieberegistern (SR) aufgebaut werden. Das Empfangs-SR liest die Daten seriell von der MCU ein und gibt sie parallel aus, während ein zweites SR die Daten



Porterweiterung mit Schieberegistern über das SPI

parallel einliest und sie seriell an die MCU weiter gibt. Mit dem Signal SS' selektiert die MCU zuerst die beiden Schieberegister. Anschliessend werden durch das Clocksignal SCK über die Leitung MOSI Daten in das Ausgabe-SR1 geschrieben und gleichzeitig über die MISO Leitung Daten aus dem Eingabe-SR2 in die MCU gelesen.

Anschluss mehrerer SPI Slaves

Parallel geschaltete Slaves:

Mehrere Slaves können parallel an die Leitungen SCK, MOSI, MISO angeschlossen werden. Jeder Slave wird jedoch über eine separate SS' Leitung von der MCU selektiert. So kann die MCU den Teilnehmer auswählen. Dabei ist zu berücksichtigen, dass die SS' Leitung die MISO Ausgänge der nicht selektierten Slaves in den TriState schaltet.

Seriell geschaltete Slaves:

Mehrere Slave SR können in Serie geschaltet werden. In dieser Konfiguration werden mehrere Datenbytes nacheinander ausgegeben dabei muss die SS' Selectleitung dauernd aktiv bleiben ('0'). Sobald die Daten in den SR an den richtigen Positionen sind, wird SS' deaktiviert. Diese Schaltungsart benötigt keine zusätzlichen Signalleitungen.