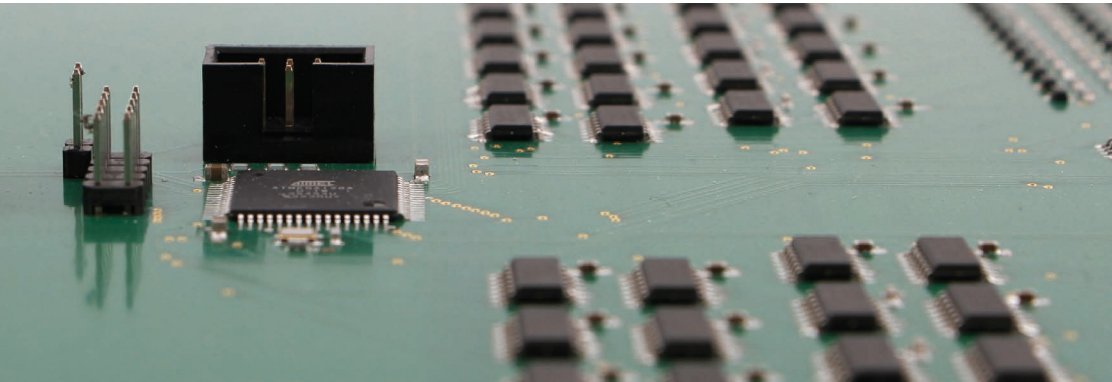


## Digitale Kommunikation mit modernen Sensoren über serielle Bussysteme



Mit dem Siegeszug der Digitaltechnik in nahezu allen Bereichen der Elektronik sind serielle Bussysteme ein unverzichtbarer Bestandteil in elektronischen Geräten geworden. Für die Kommunikation mit modernen Sensoren haben sich vor allem die Bussysteme I<sup>2</sup>C und SPI etabliert. Die Frage, welches Bussystem für die jeweilige Applikation geeignet ist, lässt sich oft nicht eindeutig beantworten. Im Folgenden wird ein kurzer Überblick über beide Bussysteme gegeben und die Vor- und Nachteile diskutiert.

Seit ihrer Entwicklung in den 1980er Jahren haben sich die seriellen Bussysteme I<sup>2</sup>C [1] und SPI [2,3] in nahezu allen Bereichen der Elektronik etabliert. Sie finden sich u.a. in Geräten der Robotik, der Medizintechnik, der Mess- und Regelungstechnik, der Haustechnik und der Automobilindustrie. Diese seriellen Bussysteme bieten eine schnelle und sichere bidirektionale Datenübertragung zwischen Mikrocontrollern und weiteren peripheren Systemkomponenten (z. B. Sensoren, Aktoren, Speicher, ADCs...) und sind vor allem beim Informationsmanagement in Systemen mit kurzen Übertragungslängen das Mittel der Wahl.

Autorin:  
Carola Kiehn  
Entwicklungsingenieurin  
Analog Microelectronics GmbH  
www.analogmicro.de

### Gemeinsamkeiten

Dem I<sup>2</sup>C- und SPI-Bussystem ist gemeinsam, dass die Datenübertragung seriell, bidirektional und synchron abläuft. Sowohl der I<sup>2</sup>C- als auch der SPI-Bus arbeiten nach dem Controller-Target/Master-Slave Prinzip: Ein Mikrocontroller (Controller/Master) kann über den jeweiligen Bus mit mehreren Teilnehmern (Target/Slave) kommunizieren. Dabei wird die Kommunikation vom Controller initiiert und gesteuert, während die peripheren Komponenten auf den Controller antworten (Bild 1). Die Datenübertragung erfolgt dabei auf dem Bussystem in beide Richtungen synchron zu vom Controller generierten Taktimpulsen. Die Information wird in Form von Datenpaketen von jeweils acht Bits (1 Byte) verpackt und seriell (nacheinander) über den Datenbus geschickt. Eine „0“ wird als niedriger Spannungspegel (LOW) und eine „1“ als hoher Spannungspegel (HIGH) der Datenleitung übertragen.

### Der I<sup>2</sup>C-Bus

Der grundlegende Aufbau zur Kommunikation via I<sup>2</sup>C-Bus ist in Bild 2 gezeigt. Es werden lediglich zwei Busleitungen benötigt, die Serial-Data-Leitung (SDA-Leitung) und die Serial-Clock-Leitung (SCL-Leitung). Die Datenübertragung findet bidirektional über die SDA-Leitung statt, d. h. die Informationen werden auf dieser Leitung sowohl vom Controller zum Target als auch umgekehrt vom Target zum Controller übertragen. Die SCL-Leitung dient der Synchronisation der Datenübertragung. Darüber hinaus sind zwei weitere Leitungen, Versorgungs-

spannung (VCC) und Masse (GND), zur Versorgung der Busteilnehmer erforderlich. Zusätzlich werden die SDA- und die SCL-Leitung mit der VCC-Leitung über Pull-Up Widerstände verbunden. Diese Widerstände garantieren, dass die Busleitungen im ungenutzten Zustand auf dem Pegel der Versorgungsspannung (HIGH) liegen.

### Das I<sup>2</sup>C-Protokoll

Die Kommunikation erfolgt nach dem I<sup>2</sup>C-Protokoll, welches den Ablauf genau vorgibt (Bild 3). Zu Beginn der Kommunikation befinden sich sowohl die SCL- als auch die SDA-Leitung im Leerlauf, d. h. auf HIGH-Pegel. Der Controller initiiert die Datenübertragung mit einer Startbedingung, indem er die SDA-Leitung von HIGH auf LOW zieht, während die SCL-Leitung auf HIGH steht. Anschließend wird mit jedem Clockimpuls ein Datenbit übertragen. Dabei darf die Änderung des Zustandes auf der SDA-Leitung nur dann erfolgen, wenn die SCL-Leitung auf LOW steht. Der Controller beendet die Datenübertragung, indem er eine Stoppbedingung schickt. Dazu setzt der Controller die SDA-Leitung von LOW auf HIGH, während die SCL-Leitung auf HIGH steht.

SDA und SCL sind nun wieder auf HIGH-Pegel und der Bus ist frei für einen neuen Datentransfer.

### Datentransfer im Detail

Im Detail läuft der Datentransfer wie folgt ab: Der Controller sendet nach der Startbedingung die Adresse des Busteilnehmers, den er ansprechen möchte. Die Adresse wird in Form von 8 bit (1 Byte) gesendet und beinhaltet die 7-bit Adresse des Targets und ein Direction Bit, das festlegt, ob das Target Daten empfangen (8. Bit = 0 / LOW, WRITE) oder Daten senden soll (8. Bit = 1 / HIGH, READ). Eine 10-bit-Adresse wird von manchen Komponenten ebenfalls unterstützt, hier wird nur die 7-bit-Adresse betrachtet.

Hat das Target das Adressbyte erfolgreich interpretiert, bestätigt es dies mit der für das I<sup>2</sup>C-Protokoll typischen Empfangsbestätigung, dem Acknowledge-Bit (ACK). Dafür zieht es die SDA-Leitung im zusätzlichen Takt auf LOW. Hat der Controller ein WRITE-Bit geschickt, so sendet der Controller anschließend weitere Datenbytes an das Target, beginnend mit dem Most Significant Bit (MSB). Das Target bestätigt den Erhalt jedes Bytes mit ACK, bis der Controller die Datenübertragung mit einer Stoppbedingung beendet. Hat der Controller mit dem Adressbyte ein READ-Bit gesendet, fordert er Daten vom Target an. Nachdem das Target das Adressbyte mit einem ACK bestätigt hat, sendet es Daten in Form von Bytes, ebenfalls beginnend mit dem MSB, wobei jedes Byte vom Controller mit einem ACK bestätigt wird. Der Controller beendet die Kommunikation mit einer Stoppbedingung. Während der gesamten Kommunikation erzeugt der Controller die notwendigen Clockimpulse.

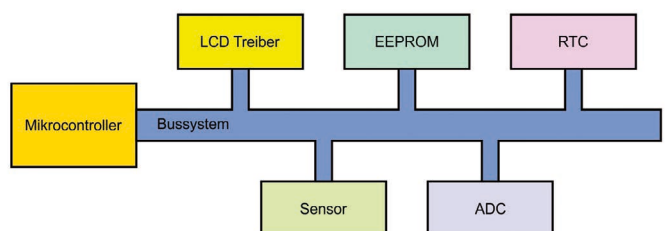


Bild 1: Schematische Darstellung eines Bussystems

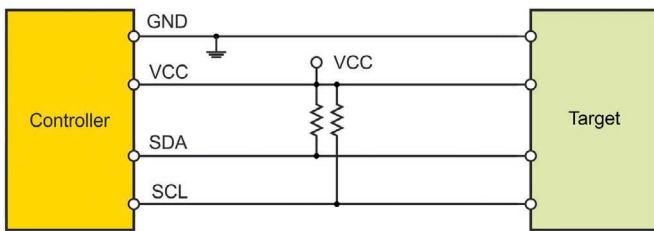


Bild 2: Aufbau des I<sup>2</sup>C-Bussystems

Target-Select (TS)-Leitung wird das anzusprechende Target ausgewählt.

## Kommunikationsprotokoll

Die Kommunikation auf dem SPI-Bus folgt dem Kommunikationsprotokoll, das in Bild 5 dargestellt ist. Um den Datentransfer mit einem Target zu initiieren, zieht der Controller die TS-Leitung des Targets auf

Zustand auf der jeweiligen Datenleitung bei einer Flanke des Clockimpulses vom Sender festgelegt und bei der zweiten Flanke vom Empfänger abgetastet (üblicherweise wird der Zustand bei der fallenden Flanke festgelegt und bei der steigenden Flanke des Clockimpulses abgetastet).

## Einstellmöglichkeiten

Obwohl vom SPI-Protokoll nicht vorgegeben, ist eine Datenübertragung in Form von 8-bit großen Datenpaketen in der Praxis etabliert. Die Reihenfolge der Daten, das heißt, ob zuerst das Most-Significant-Bit (MSB) oder das Least-Significant-Bit (LSB) gesendet wird, ist nicht prinzipiell festgelegt und hängt von den Spezifikationen und Einstellmöglichkeiten der Busteilnehmer ab. Durch die gleichzeitige Übertragung von Daten auf COTI- und CITO-Leitung bietet der SPI-Bus die Voraussetzungen für eine voll-duplex Kommunikation, die jedoch nur ausgenutzt werden kann, wenn bei Initiierung der Kommunikation durch den Controller bereits Daten am Target vorliegen.

## Verschiedene Übertragungsmodi

Das SPI-Protokoll erlaubt verschiedene Übertragungsmodi zwischen Controller und Target, die festlegen, ob die Clock-Polarität (CPOL) im Ruhezustand HIGH oder LOW ist und auf welcher Flanke die Daten (CPHA) übertragen werden. Durch die Kombination der beiden Einstellmöglichkeiten CPOL und CPHA ergeben sich insgesamt vier Modi (Bild 6). Im Modus 0 (Standardmodus) ist die SCLK-Leitung im Ruhezustand LOW (CPOL = 0). Das jeweilige Datenbit wird bei fallender Taktflanke des ersten Clockimpulses vom Sender gesetzt (Umschaltflanke) und

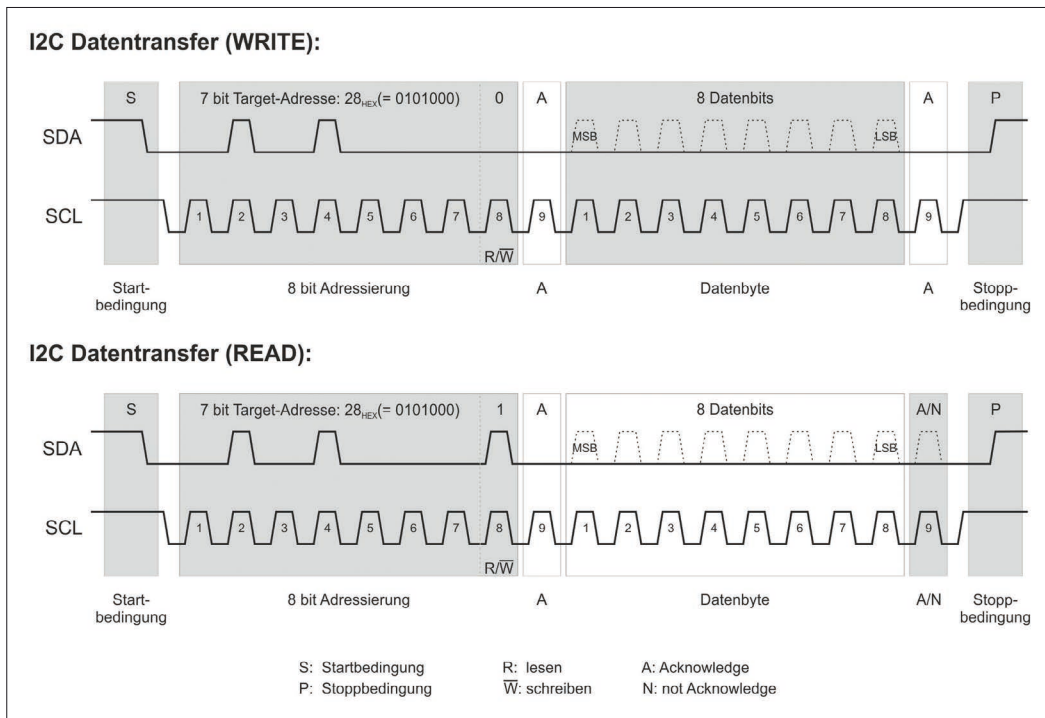


Bild 3: Ablauf der I<sup>2</sup>C-Kommunikation

An einem I<sup>2</sup>C-Bus können mehrere Teilnehmer parallel an die SDA- und SCL-Leitung angeschlossen werden, solange die Teilnehmer unterschiedliche I<sup>2</sup>C-Adressen haben. Grundsätzlich ist auch ein Multi-Controller-Betrieb an einem I<sup>2</sup>C-Bus möglich.

## Anzahl der Busteilnehmer

Die maximale Anzahl der Busteilnehmer ist formal durch die Anzahl der verfügbaren Adressen begrenzt (7-bit-Adresse: 127 mögliche Adressen). Das I<sup>2</sup>C-Protokoll reserviert einige wenige Adressen für besondere Zwecke, diese dürfen nicht vergeben werden. Die maximale Übertragungsgeschwindigkeit ist durch die Taktrate auf der SCL-Leitung beschränkt, im Standardmodus auf 100 kHz, im Fast-Mode auf 400 kHz und im High-Speed-Mode auf 3,4 MHz.

In der Praxis beschränken parasitäre Kapazitäten und Widerstände der Teilnehmer und der Busleitungen die Länge des Busses, die Anzahl der

Teilnehmer und die Übertragungsgeschwindigkeit. Üblicherweise lassen sich Buslängen von 2 m ohne Einschränkungen realisieren.

## Der SPI-Bus

Der grundlegende Aufbau eines SPI-Bussystems ist in Bild 4 dargestellt. Für den Datentransfer zwischen Controller und Target über den SPI-Bus werden neben den zwei Leitungen für die Versorgung der Busteilnehmer vier weitere Busleitungen zur Kommunikation benötigt. Die Datenübertragung erfolgt über die Controller-Output-Target-Input-Leitung (COTI-Leitung) für die Datenübertragung vom Controller zum Target und die Controller-Input-Target-Output-Leitung (CITO-Leitung) für die Datenübertragung vom Target zum Controller. Über die Serial-Clock (SCLK)-Leitung findet die Synchronisation der Kommunikation zwischen Controller und Target statt und über die

LOW. Dieses Target ist nun solange für den Datentransfer ausgewählt, bis der Controller die TS-Leitung zurück auf HIGH setzt.

Die Datenübertragung erfolgt synchron zu den vom Controller erzeugten Clockimpulsen auf der SCLK-Leitung. Mit jedem Clockimpuls können sowohl Daten über COTI-Leitung (vom Controller zum Target) als auch über die CITO-Leitung (vom Target zum Controller) übertragen werden. Dazu wird der

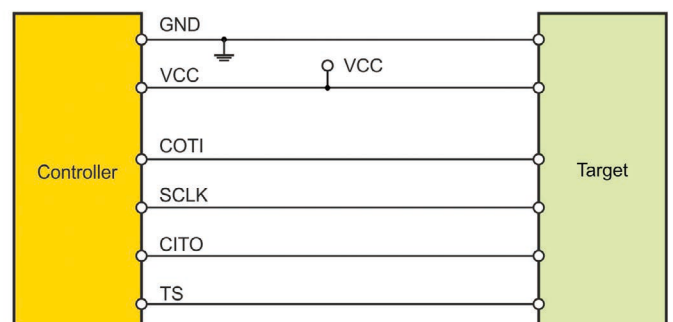
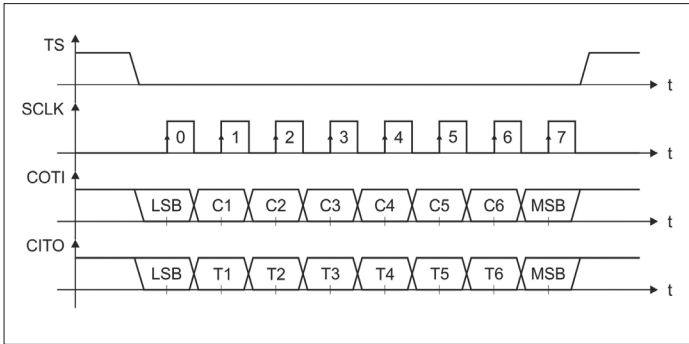


Bild 4: Aufbau des SPI-Bussystems



**Bild 5: Ablauf der SPI-Kommunikation**

die Daten werden mit dem nachfolgenden Clockimpuls bei steigender Flanke abgetastet (Samplingflanke) (CPHA=0). Im Modus 1 ist ebenfalls CPOL = 0. Die Daten werden bei steigender Taktflanke gesetzt und bei fallender Flanke abgetastet (CPHA = 1). Im Modus 2 ist CPOL = 1, d. h. die SCLK ist im Ruhezustand HIGH und CPHA = 0, im Modus 3 ist CPOL = 1 und CPHA = 1. Die Auswahl des SPI-Modus hängt von den Spezifikationen der beteiligten Targets ab. Die meisten SPI-kompatiblen Komponenten unterstützen mehrere Modi, um eine Interoperabilität zu gewährleisten.

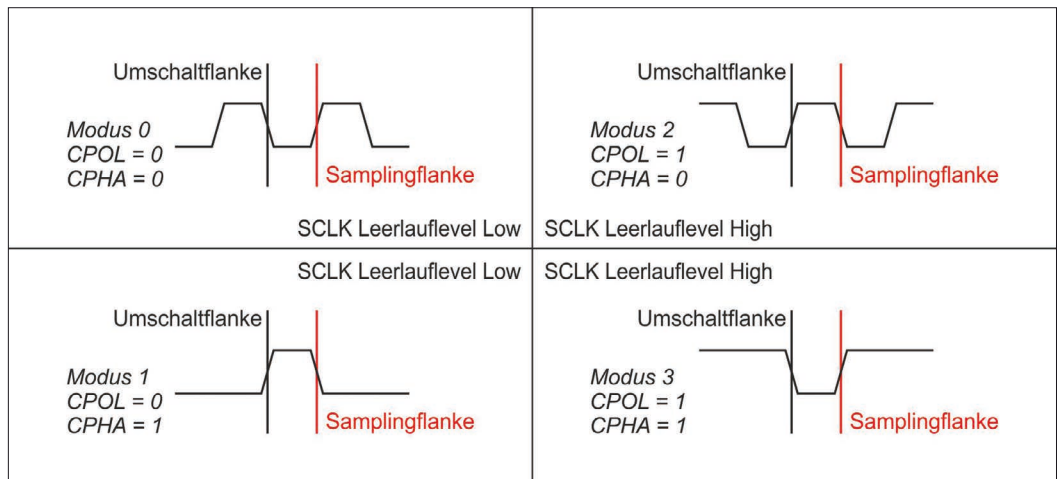
## Mehrere Targets – ein SPI-Bus

Prinzipiell ist es möglich mehrere Targets an einem SPI-Bus anzuschließen. Dafür muss der Controller jedem Target eine eigene TS-Leitung zur Verfügung stellen. Die maximale Anzahl der Targets ist grundsätzlich nicht beschränkt, ergibt sich jedoch in der Praxis aus der Anzahl der dem Controller zur Verfügung stehenden TS-Pins.

Die maximale Geschwindigkeit der Datenübertragung ist nach dem SPI-Protokoll ebenfalls nicht festgelegt. In der Praxis wird die maximale Übertragungsgeschwindigkeit durch die Reaktionsgeschwindigkeit der Busteilnehmer und die Signalintegrität auf den Busleitungen begrenzt. Typischerweise lassen sich Übertragungsraten von 10 - 20 MHz erreichen.

## I<sup>2</sup>C- und SPI-Bus gegenübergestellt

Im Vergleich zum SPI-Bus bietet sich der I<sup>2</sup>C-Bus durch seine einfache Topologie, die bidirektionale Datenübertragung und die Adressierung über die Datenleitung für platzkritische Anforderungen an, insbesondere wenn mehrere Targets



**Bild 6: SPI-Modi**

auf geringen Raum und mit wenigen Leitungen platziert werden sollen. Zudem ermöglicht der I<sup>2</sup>C-Bus den Einsatz von mehreren Controllern an einem Bus. Ein weiteres Plus ist die Empfangsbestätigung ACK, die die Sicherheit der Datenübertragung erhöht. Nachteilig ist die Notwendigkeit von Pull-Up Widerständen, die die erreichbaren Anstiegszeiten der Flanken und somit die Übertragungsraten begrenzen.

Der SPI-Bus wird aufgrund der Schaltungstechnik und der voll-duplex Übertragung für Anwendungen bevorzugt, die eine hohe Datenrate und eine präzise Timing-Steuerung erfordern. Pull-Up Widerstände sind hier grundsätzlich nicht erforderlich. Das SPI-Protokoll ist weniger stark reglementiert und hängt in seiner Umsetzung von den Spezifikationen der Busteilnehmer ab. Der SPI-Bus eignet sich aufgrund der benötigten TS-Leitungen nur für eine begrenzte Anzahl an Teilnehmern.

Letztendlich muss die Entscheidung für eines der Bussysteme aber immer anhand der spezifischen Anforderungen der Anwendung gewählt werden.

## I<sup>2</sup>C und SPI kompatible Sensoren

Heutzutage sind Komponenten/Sensoren am Markt verfügbar, die sowohl I<sup>2</sup>C- als auch SPI-kompatibel sind. Damit kann die Auswahl von Sensoren rein auf Basis von Spezifikationen und unabhängig vom Bussystem getroffen werden. Nach der erfolgreichen Erprobung eines Sensors kann dieser leicht in

aktiviert wird. Die max. Übertragungsraten im I<sup>2</sup>C-Betrieb beträgt 3,4 MHz und im SPI-Betrieb 20 MHz. Den AMS 5935 gibt es für Druckbereiche von 1,25 mbar bis 2 bar für alle Druckarten. Mit der geringen Stromaufnahme und dem Versorgungsspannungsbereich von 1,8 bis 3,3 V ist er ideal für Mikrocontrolleranwendungen im Batteriebetrieb geeignet.

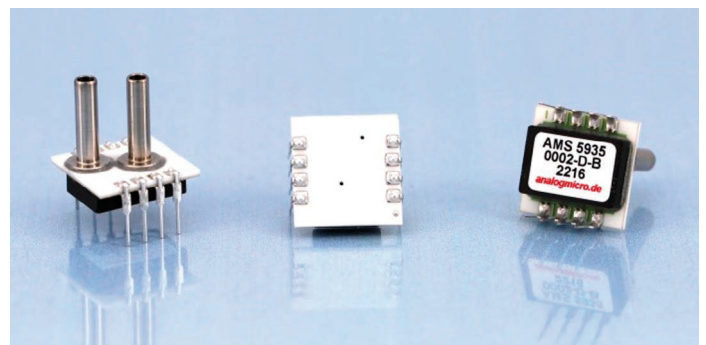
unterschiedliche Bussysteme eingesigned werden, sodass die bus-spezifischen Vorteile in der jeweiligen Applikation ausgenutzt werden können.

## Beispiel

Als Beispiel soll hier der Drucksensor AMS 5935 [4] von Analog Microelectronics GmbH vorgestellt werden (Bild 7). Dieser leiterplattenmontierbare Sensor mit 18-bit Auflösung gibt kalibrierte Druck- und Temperaturmesswerte über eine universelle I<sup>2</sup>C/SPI-Schnittstelle aus. Die Kommunikationsart kann nach jedem Power-Up neu eingestellt werden, indem entweder ein I<sup>2</sup>C-Befehl folgt oder die TS-Line

## Quellen:

- [1] „UM10204: I<sup>2</sup>C-bus specification and user manual, Rev. 7.0-1 October 2021“, User Manual NXP
- [2] Patent US4816996: Queued serial peripheral interface for use in a data processing system., Motorola; Hill, Jelemensky, Heene
- [3] F. Leens, „An introduction to I<sup>2</sup>C and SPI protocols“ in IEEE Instrumentation & Measurement Magazine, February 2009, doi: 10.1109/MIM.2009.4762946
- [4] Datenblatt AMS5935: <https://www.analog-micro.com/de/produkte/drucksensoren/board-mount-drucksensoren/ams5935/> ◀



**Bild 7: Board-Mount Drucksensor AMS 5935 mit I<sup>2</sup>C- und SPI-Interface**