

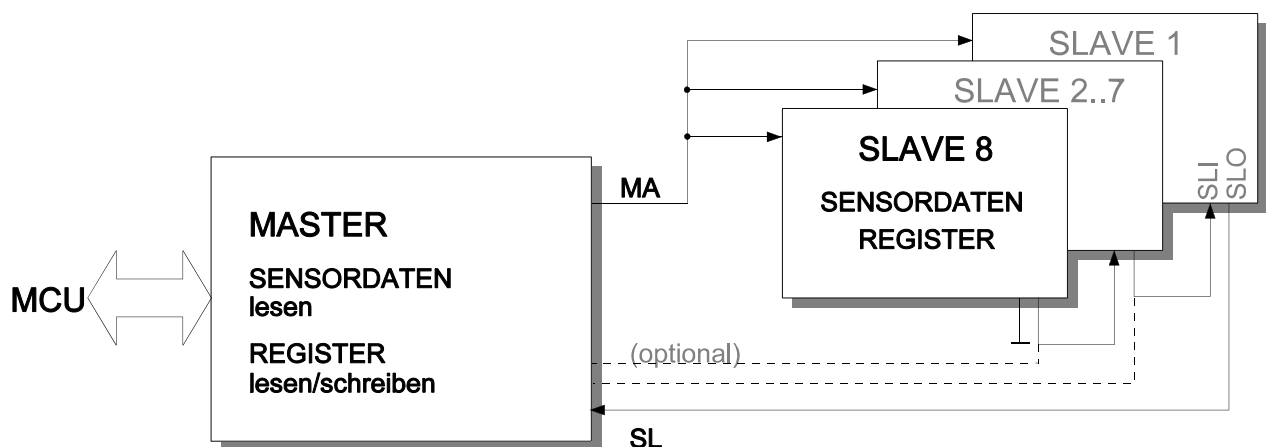
### EIGENSCHAFTEN

- ◆ Datenübertragung über zwei unidirektionale Leitungen
- ◆ Hardware-kompatibel zu SSI
- ◆ Bis zu 8 Teilnehmer (Single Master, Multi Slave)
- ◆ Serielle oder parallele Teilnehmer-Anbindung an den Master
- ◆ Zyklisches Lesen von Sensordaten beliebiger (hardwarebestimmter) Länge
- ◆ Hohe Lesegeschwindigkeit durch Laufzeitkompensation durch den Master
- ◆ Anforderung von Verarbeitungszeit durch den Sensor möglich (verzögerte Übertragung)
- ◆ Synchrone Speicherung der Daten in allen Teilnehmern durch Triggerung vom Master
- ◆ Sensor-Parametrisierung durch bidirektionalen Zugriff auf Teilnehmer-Register
- ◆ Hohe Sicherheit der Register-Kommunikation durch Datenprüfung (CRC); je nach Hardware-Implementierung auch für Sensordaten

### ANWENDUNGEN

- ◆ Industrieller Sensor-Bus, z.B. zur Positionserfassung an mehreren Achsen
- ◆ Motor-Feedback
- ◆ Chip-Kommunikation in Multi-Sensor-Systemen

### KONFIGURATION



### KURZBESCHREIBUNG KONFIGURATION

Diese Spezifikation beschreibt ein serielles Sensor-Interface zum synchronen, schnellen und sicheren Auslesen von Sensordaten sowie zum bidirektionalen Zugriff auf die Sensorregister. Das Interface ist hardwarekompatibel zur SSI-Schnittstelle und besitzt zwei oder mehr unidirektionale Leitungen.

Ein System besteht aus einem Controller (Master) und ein bis acht Sensoren (Slaves), die seriell oder semiparallel geschaltet werden können. Bei beiden Verschaltungsarten wird die vom Master kommende Leitung 'MA' an alle Slaves angeschlossen.

Bei der seriellen Teilnehmer-Anbindung ermöglicht ein Eingang (SLI) am Sensor - verbunden mit dem Ausgang (SLO) des Vorgängers - das Einlesen der Daten dieses Sensors und so den Busbetrieb mit mehreren Teilnehmern (Bild 1). Der Master erhält die Daten der Slaves nacheinander.

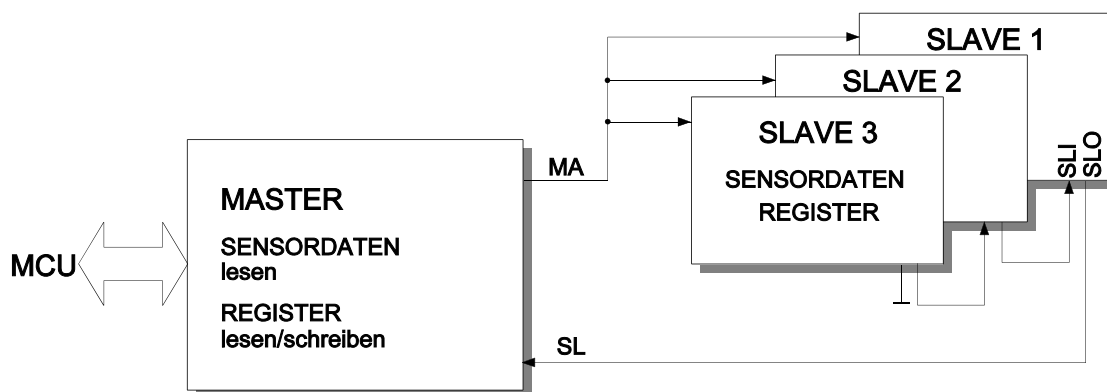


Bild 1: serielle Teilnehmer-Anbindung

Prinzipiell besteht die Möglichkeit die Ausgänge der Teilnehmer auch parallel zum Master zurückzuführen (Leitungen SL0..SL2 in Bild 2), so dass die Antworten aller Slaves gleichzeitig eingelesen werden. Dieses Verfahren bedeutet mehr Aufwand für den Master, bietet aber den Vorteil kürzerer Übertragungszeiten.

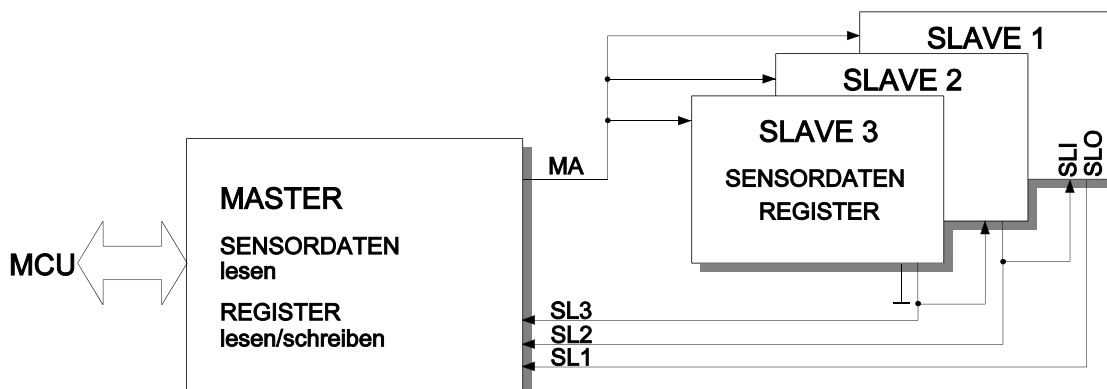


Bild 2: semiparallele Teilnehmer-Anbindung

### KURZBESCHREIBUNG KOMMUNIKATION

Das Busprotokoll ermöglicht zwei Kommunikationsarten, den "Sensormodus" und den "Registermodus". Es stellt sicher, dass der Slave im Fehlerfall immer in einen definierten Zustand zurückkehrt.

Im Sensormodus können Sensordaten ohne Adressierungsnotwendigkeit zyklisch schnell ausgelesen werden. Nach der Kommunikationsinitialisierung durch den Master, bei der auch Leitungslaufzeiten ausgemessen und zur Erhöhung der Übertragungsgeschwindigkeit vom Master kompensiert werden können, gibt dieser nur noch den Takt vor, mit dem die Sensordaten aus den Slaves ausgegeben werden (Bild 3). Für die Fehlerfreiheit der übertragenen Daten ist eine entsprechende hardwaremässige Überprüfung empfehlenswert.

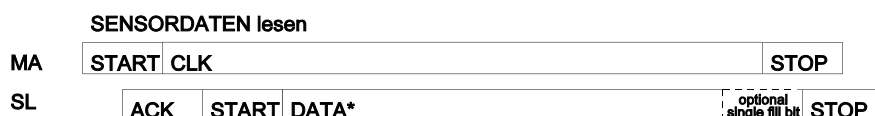


Bild 3: Signale im Sensormodus

Im Registermodus können die Register in den einzeln adressierbaren Sensoren beschrieben bzw. ausgelesen werden. Die BiSS sieht hierfür eine spezielle Adressierungssequenz (Bild 4) vor, in der 3 Bits zur Baustein- und 7 Bits zur Registeradressierung eingesetzt werden; zur Erhöhung der Übertragungssicherheit werden die Daten der Adressierungssequenz mit einem 4-Bit-CRC gesichert.



Bild 4: Signale im Registermodus - Adressierungssequenz

Mit den 7 Bit zur Registeradressierung können pro Slave 128 Register von jeweils 8 Bit Länge adressiert werden. Sollte dieser Bereich nicht ausreichen, so besteht die Möglichkeit, einem Slave weitere Blöcke zu je 128 Registern zuzuordnen, wobei jeder dieser Blöcke die Anzahl der möglichen Slaves im System um einen reduziert.

Sollen Daten aus einem Register ausgelesen werden, so muss der Master nach der Adressierungssequenz nur noch die entsprechende Anzahl von Takten liefern. Beim Beschreiben der Register werden die einzuschreibenden Daten vom Master PWM-kodiert übertragen.



Bild 5: Signale im Registermodus - Register lesen



Bild 6: Signale im Registermodus - Register schreiben

### FUNKTIONSBESCHREIBUNG

#### Die Kommunikationsarten

Mit der *BiSS* Sensor-Schnittstelle können wahlweise Daten sehr rasch ausgelesen oder Register beschrieben sowie ausgelesen werden. Die Umschaltung zwischen diesen beiden Kommunikationsarten erfolgt durch eine Zeitbedingung am Anfang jedes Kommunikationszyklus: dauert der erste low-Pegel des Master-Signals länger als die (einstellbare) Zeit 'timeout<sub>SENS</sub>', so werden die Sensoren in den Registermodus versetzt, bei Unterschreitung dieser Zeit in den Sensormodus.

#### Kommunikationsart 'SENSORMODUS'

##### *EIN-SENSOR-BETRIEB* oder *SEMIPARALLELE TEILNEHMER-ANBINDUNG*

Zum Auslesen der Sensordaten muss der Controller (Master) nach der ersten gesendeten fallenden Flanke innerhalb von 'timeout<sub>SENS</sub>' eine steigende Flanke erzeugen. Auf diese Flanke hin speichert der Sensor (Slave) die aktuellen Daten oder beginnt mit der Messwertkonvertierung. Die zweite steigende Flanke veranlasst den Sensor dazu, an seinem Ausgang ebenfalls einen low-Pegel zu erzeugen (Bild 7, Teil S1).

#### Sensordaten lesen #1

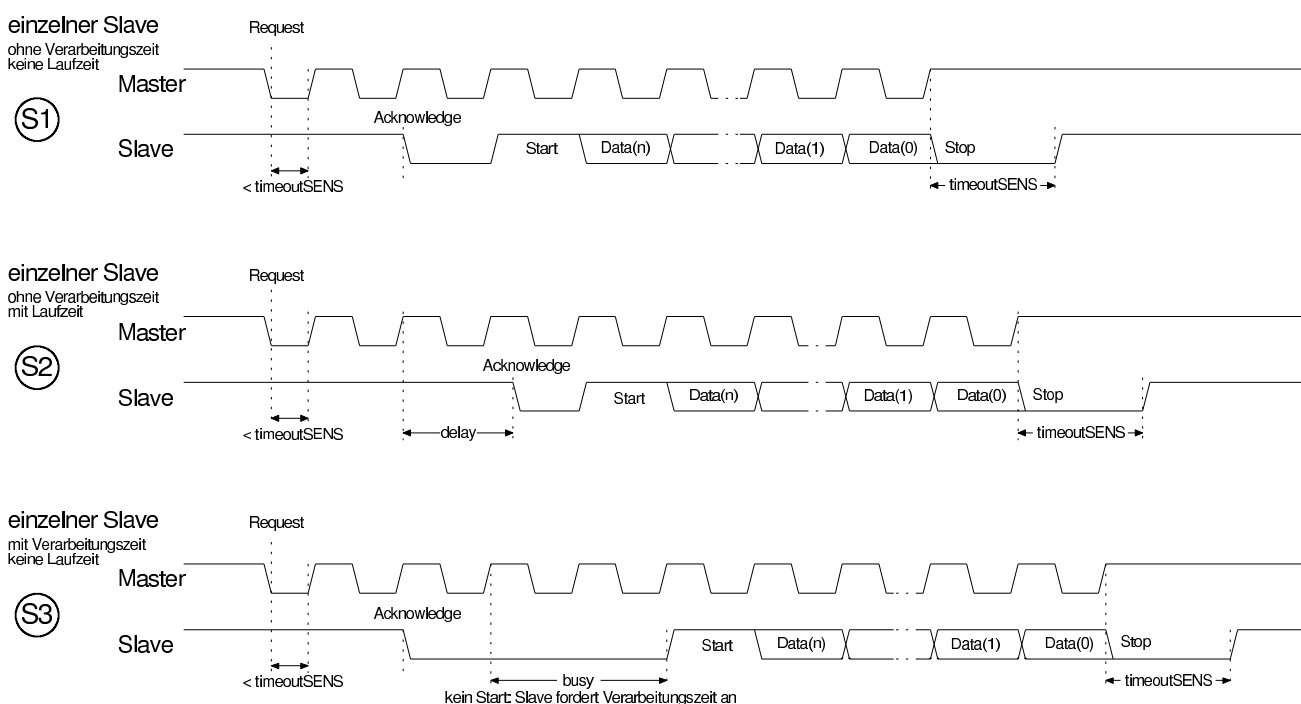


Bild 7: Einlesen der Sensordaten eines einzelnen Slaves;  
 S1: ohne Laufzeit; ohne Verarbeitungszeit  
 S2: mit Laufzeit, ohne Verarbeitungszeit  
 S3: ohne Laufzeit, mit Verarbeitungszeit

Aus der zeitlichen Differenz zwischen der zweiten steigenden Master-Flanke und der Slave-Antwort kann der Master die Leitungslaufzeit bestimmen und diese durch entsprechende Verschiebung der Abtastung des Slavesignals kompensieren. Hierdurch wird eine beschleunigte Kommunikation ermöglicht (Bild 7, Teil S2). Nach dieser Kommunikationsinitialisierung wird das Ausgangssignal des Masters verwendet, um die Sensordaten mit den steigenden Flanken aus dem Slave auszutakten. Hierbei beginnt die Slave-Antwort mit einem Startbit ('1') und endet mit einem Stopbit ('0'). Dies ermöglicht dem Slave durch Verzögerung des Startbits entsprechend Verarbeitungszeit anzufordern (Bild 7, Teil S3).

### SERIELLE TEILNEHMER-ANBINDUNG

Bei der seriellen Anbindung mehrerer Slaves werden die Daten der 'hinteren' Slaves durch die 'vorderen' (näher am Eingang des Masters befindlichen) Slaves hindurchgeschleust. Hierbei folgen die Datenbits der einzelnen Slaves ohne Trennung durch Start- oder Stopbits direkt aufeinander. In Bild 8 sind die entsprechenden Signalverläufe bei der Serienschaltung zweier Slaves dargestellt, wobei Slave1 der 'vordere' und Slave2 der 'hintere' ist.

#### Sensordaten lesen #2

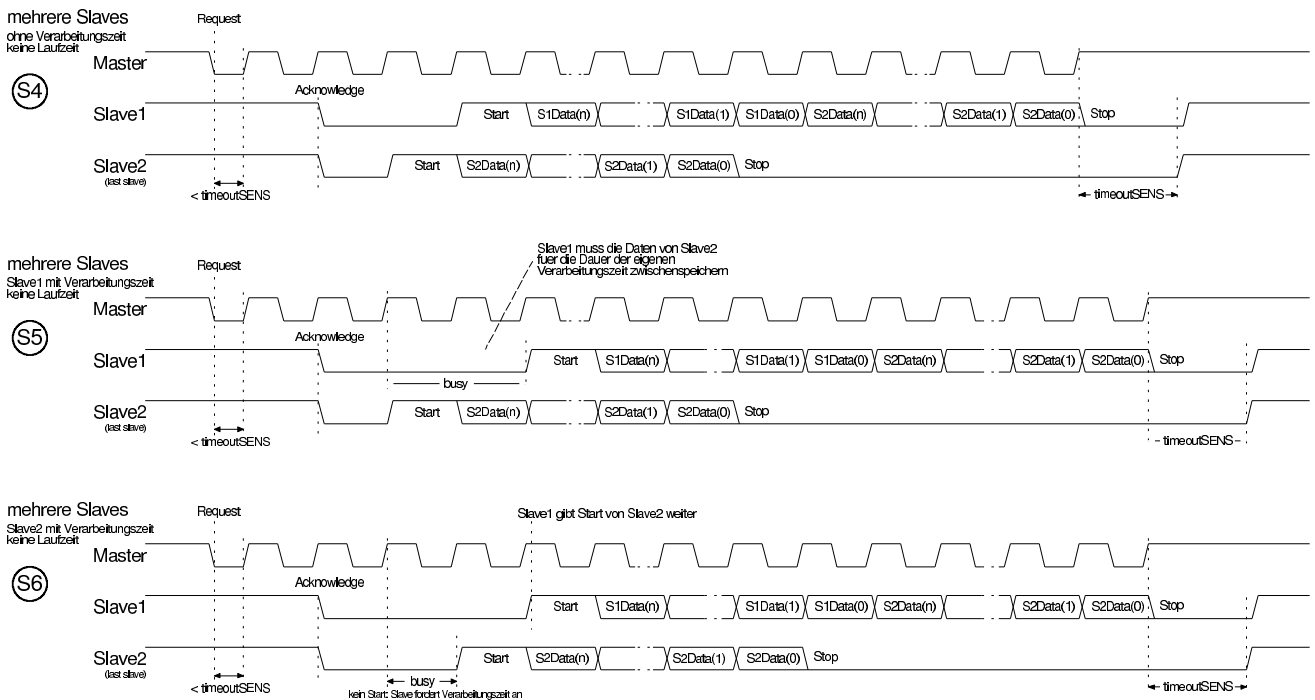


Bild 8: Einlesen der Sensordaten zweier in Serie geschalteter Slaves (ohne Laufzeit);  
 S4: ohne Verarbeitungszeit  
 S5: mit Verarbeitungszeit von Slave1  
 S6: mit Verarbeitungszeit von Slave2

Benötigt ein 'vorne' liegender Slave Verarbeitungszeit, so muss er die während dieser Zeit eingehenden Signale der hinteren Slaves zwischenspeichern (Bild 8, Teil S5). Verzögert ein 'hinten' liegender Slave sein Startsignal, weil er Verarbeitungszeit benötigt, so wird diese Verzögerung von den vorderen Slaves an den Master weiter gegeben (Bild 8, Teil S6).

### Kommunikationsart 'REGISTERMODUS'

In den Registermodus gelangt man, indem der Master den ersten low-Pegel eines Kommunikationszyklus länger als  $\text{timeout}_{\text{SENS}}$  anstehen lässt. Der Slave antwortet nach  $\text{timeout}_{\text{SENS}}$  auf die fallende Flanke des Masters ebenfalls mit einer fallenden Flanke an seinem Ausgang - für den Master das Signal, dass die eigentliche Datenübertragung beginnen kann. Diese Datenübertragung erfolgt PWM-kodiert, was eine einfache Übertragung von Takt und Daten auf der Master-Leitung ermöglicht. Die Leitungslaufzeiten müssen allerdings klein gegenüber der verwendeten Taktrate sein.

Erst erfolgt die Baustein- und Registeradressierung, danach das Auslesen der Registerdaten des Slaves oder das Einschreiben neuer Daten.

Die Adressierung erfolgt durch das Senden eines Startbits ('1'), gefolgt von 3 Bit Slave-ID, 7 Bit Registeradresse, 1 Bit ('WNR') zur Read/Write-Umschaltung, 4 Bit CRC (CRC-Polynom '10011b', invertiert ausgegeben) und einem Stopbit ('0'). Während der Adressierungssequenz nimmt jeder Slave - beginnend mit dem 'hintersten' - so viele Slave-IDs in Besitz wie notwendig sind, um alle seine Register adressieren zu können. Dies geschieht dadurch, dass jeder Slave mit einer entsprechenden Verzögerung seiner steigenden Signalfanke reagiert (siehe Bild 9 und Applikationshinweise).

#### Bausteinadressierung (Registermodus)

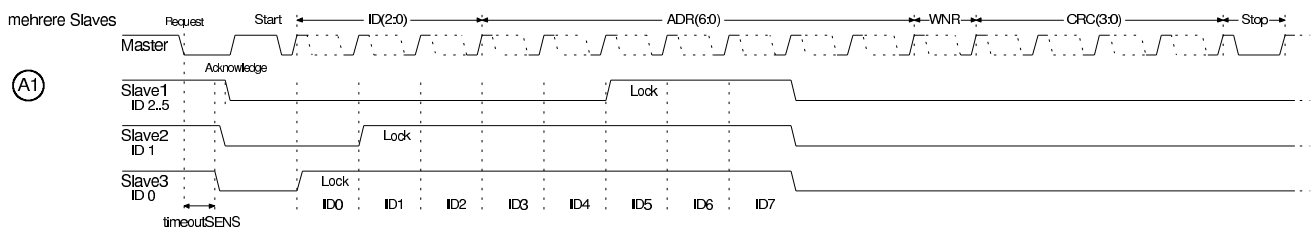


Bild 9: Bausteinadressierung im Registermodus

### REGISTER LESEN

Ist das übertragene WNR-Bit eine '0', so wird nach Abschluss der Adressierungssequenz ein oder mehrere Register ausgelesen; der Slave-Ausgang befindet sich zu diesem Zeitpunkt auf low-Pegel. Stimmt die übertragene Slave-ID mit der eigenen Slave-ID überein, so reagiert der Slave mit einem Startbit ('1') (Bild 10, Teil R1). Sollte der Slave noch Verarbeitungszeit benötigen, so verzögert er einfach die spätere Startbit-Ausgabe entsprechend (Bild 10, Teil R2).

Das Protokoll sieht die Möglichkeit vor, dass mit einer Adressierung gleich mehrere aufeinander folgende Register gelesen (oder auch beschrieben) werden können. Hierzu taktet der Master weiter, und der Slave zählt intern die Adresse hoch (Autoinkrement) und antwortet mit einem neuen Datenwort bestehend aus einem Startbit ('1'), den eigentlichen Datenbits, den vier CRC-Bits (CRC-Polynom '10011b', invertiert ausgegeben) und einem Stopbit ('0') (Bild 10, Teil R3).

Hierbei kann es vorkommen, dass verbotene oder nicht vorhandene Register ausgelesen werden sollen; in diesem Falle reagiert der Slave, indem er an seinem Ausgang eine Dauer-'0' ausgibt (Bild 10, Teil R4).

Benötigt ein Slave bei dem Auslesen eines durch automatische Inkrementierung angesprochenen Registers noch Verarbeitungszeit, so kann er diese durch Verzögerung des Startbits des entsprechenden Datenwortes anfordern (Bild 10, Teil R5).

Bild 10, Teil R6 zeigt das Verhalten mehrerer Slaves beim Registerrauslesen. Ein 'hinter' dem adressierten Slave liegender Sensor liefert eine Dauer-'0' (im Beispiel Slave3), der adressierte Baustein (Slave2) gibt seine Registerdaten aus und der 'vor' dem adressierten Slave liegende Sensor (Slave1) gibt die vom Vorgänger übergebenen Daten transparent weiter, generiert zusätzlich aber ein eigenes Stopbit.



### *REGISTER SCHREIBEN*

Ist das in der Adressierungssequenz übertragene WNR-Bit eine '1', so werden nach dieser Sequenz vom Master die 8 Datenbits zuzüglich der 4 CRC-Bits - eingerahmt von jeweils einem Startbit ('1') und einem Stopbit ('0') - PWM-kodiert an den adressierten Slave übertragen (Bild 11, Teil W1). Der Slave schickt die eingeschriebenen Daten unkodiert an den Master zurück, so dass dieser die Kommunikation überwachen und im Falle eines Fehlers reagieren kann.

Auch beim Schreiben von Registerwerten kann es vorkommen, dass der Slave Verarbeitungszeit benötigt; in diesem Falle verzögert der Slave sein Startbit um die benötigte Zeit. Der Master kann während dieser Zeit keine Daten in das Slave-Register übertragen und muss deshalb das gesendete Bit bis zur Antwort des Slaves wiederholen (Bild 11, Teil W2, Teil W5 für mehrere Datensätze).

Wie schon beim Lesen ist es auch beim Schreiben eines Slaves möglich, mehrere aufeinander folgende Register in einem Schreibzyklus zu beschreiben, indem aufeinander folgende Datenworte - jeweils eingerahmt von einem Start- ('1') und einem Stopbit ('0') - hintereinander übertragen werden (Bild 11, Teil W3). Sollte es hierbei passieren, dass verbotene oder nicht vorhandene Register adressiert werden, so bricht der Slave durch eine Dauer-'0' anstelle der Rücksendung des eingegangenen Startbits die Übertragung ab (Bild 11, Teil W4).

Bild 11, Teil W6 zeigt die Signalverläufe beim Beschreiben eines Slaves in einer Serienschaltung von Sensoren: 'nach' dem adressierten Slave liegende Sensoren (Slave3) liefern am Ausgang eine Dauer-'0', der adressierte Slave (Slave2) gibt die eingelesenen PWM-Daten unkodiert wieder aus, vorausliegende Sensoren (Slave1) geben die Daten der Vorgänger transparent weiter, generieren aber zusätzlich ein eigenes Stopbit.

### Register schreiben

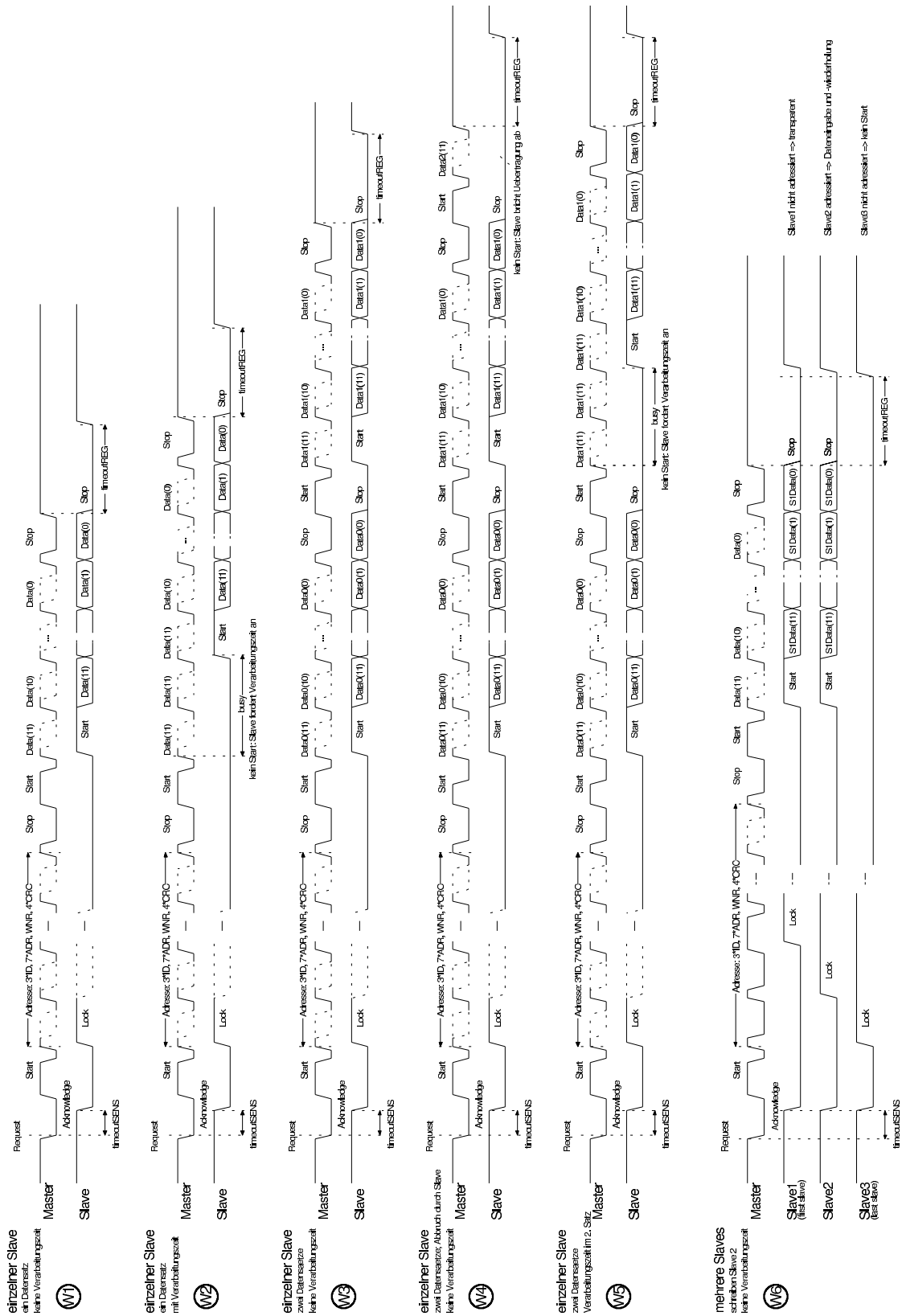


Bild 11: Register schreiben -

R1..R2: Schreiben eines einzelnen Datenwortes

R3..R5: Schreiben mehrerer Datenworte

R6: Schreiben bei mehreren Slaves in Serienschaltung

Data(11) bis Data(0): 8 bit Registerdaten zzgl. 4 bit CRC

### APPLIKATIONSHINWEISE

#### Vereinbarung einer Slave-Typenkennung

Jeder Bus-Teilnehmer, integrierte Bausteine oder andere Schaltungen, stellen dem System eine eindeutige Typenkennung zur Verfügung. Diese Kennung erwartet der BiSS-Master unter den Registeradressen '78h' und '7Fh' (6 Byte für die Baugruppe, 2 Byte für den Hersteller) und konfiguriert sich dann selbst für die angeschlossenen Teilnehmer. iC-Haus unterstützt bei der Vergabe der Typenkennungen.

#### Erweiterung des Register-Adressbereichs

Standardmäßig sieht das Protokoll der Sensor-Schnittstelle 3 Bit zur Slave-Identifizierung und 7 Bit für die Registerauswahl vor. Sollte ein Sensor mehr als die mit 7 Bit adressierbaren 128 Register beinhalten, so kann er während der Adressierungssequenz zwei oder mehr Slave-IDs in Besitz nehmen, indem er die Ausgabe des Lockbits um entsprechend viele Takte verzögert (siehe Bild 9, Slave1). Er reagiert dann nicht nur auf eine Slave-ID, sondern entsprechend auf mehrere. Die Anzahl der möglichen Slaves im System verringert sich durch die Inbesitznahme mehrerer Slave-IDs durch einen Sensor entsprechend.

Es empfiehlt sich, Sensoren mit zwei oder vier nötigen Slave-IDs so in der Gesamtschaltung einzusetzen, dass sich die in Besitz genommenen Slave-IDs nur im LSB / in den LSBs unterscheiden. In diesem Fall können diese Bits der Slave-ID direkt als hochwertigste Registeradressierungsbits eingesetzt werden, ohne dass ein Adressmapping erfolgen muss (siehe Bild 12).

#### Registermodus Adressierungssequenz (max. 8 Slaves, bei 3-Bit Slave-ID und 7-Bit Registeradresse)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
START	ID(2:0)			ADR (6:0)							WNR	CRC			STOP	

#### Registermodus Adressierungssequenz (Bei 2-Bit Slave-ID, 8-Bit Registeradresse)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
START	ID(2:1)		ADR (7:0)					WNR	CRC			STOP				

Bei Verwendung von zwei Slave-IDs (z.B: 100 und 101) kann das LSB der Slave-ID als achties Bit der Register Adresse verwendet werden

Bild 12: Bitverteilung bei 7- und 8-Bit Registeradresse

**Beispiel:** Ein Sensor mit mehr als 128, aber weniger als 256 Registern wird so platziert, dass er die Adressen '100b' und '101b' in Besitz nimmt. Das LSB der Slave-ID entspricht dann dem MSB der Register-Adresse. Der Sensor erhält damit die 2-Bit Slave-ID '10b'.

#### Fehlerüberprüfung

Für die Adressierungssequenz und für Registerdaten ist eine Fehlerüberprüfung durch einen 4-bit-CRC bereits durch das BiSS-Protokoll vereinbart. Für die Übertragung der Sensordaten empfiehlt sich eine ähnliche Überprüfung; so sollte beispielsweise zur Fehlerüberprüfung eines 20-Bit Sensorwertes ein 6-Bit-CRC gebildet und mit übertragen werden.

Ein Nachdruck dieser Spezifikation – auch auszugsweise – ist nur mit unserer schriftlichen Zustimmung und unter genauer Quellenangabe zulässig. Die angegebenen Daten dienen ausschließlich der Produktbeschreibung. Dies gilt insbesondere auch für die angegebenen Verwendungsmöglichkeiten/ Einsatzbereiche des Produktes.

Eine Garantie hinsichtlich der Eignung oder Zuverlässigkeit des Produktes für die konkret vorgesehene Verwendung wird von iC-Haus nicht übernommen. iC-Haus überträgt an dem Produkt kein Patent, Copyright oder sonstiges Schutzrecht.

Für die Verletzung etwaiger Patent- und/oder sonstiger Schutzrechte Dritter, die aus der Ver- oder Bearbeitung des Produktes und/oder der sonstigen konkreten Verwendung des Produktes resultieren, übernimmt iC-Haus keine Haftung.

Unsere Entwicklungen, IPs, Schaltungsprinzipien und angebotenen Integrierten Schaltkreise sind grundsätzlich geeignet, naheliegend und vorgesehen für einen zweckentsprechenden Einsatz in technischen Applikationen, z. B. in Geräten und Systemen und in beliebigen technischen Einrichtungen, soweit sie nicht bestehende Schutzrechte verletzen. Prinzipiell sind die Verwendungsmöglichkeiten technisch nicht beschränkt und beziehen sich beispielsweise auf Produkte des Warenverzeichnisses für die Außenhandelsstatistik, Ausgabe 2007, herausgegeben vom Statistischen Bundesamt, Wiesbaden, Dezember 2006, oder auf ein beliebiges Produkt des Produktkatalogs der Hannover-Messe 2007.

Eine zweckentsprechende Applikation unserer veröffentlichten Entwicklungen verstehen wir als Stand der Technik, die nicht mehr als erfinderisch im Sinne des Patentgesetzes gelten kann. Unsere expliziten Applikationshinweise sind nur als Ausschnitt der möglichen, besonders vorteilhaften Anwendungen zu verstehen.