BiSS Interface

PROTOKOLL-BESCHREIBUNG (C-Mode)

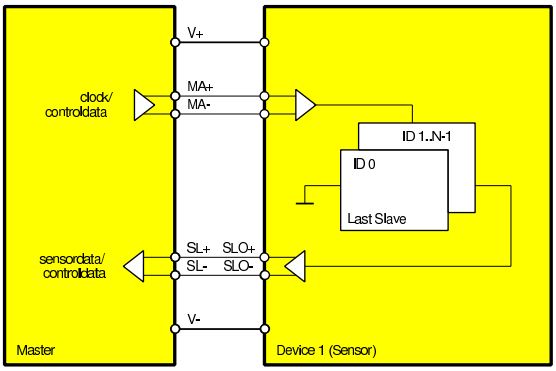
EIGENSCHAFTEN

- Sensor/Aktor-Interface
- Isochrone, echtzeitfähige Datenübertragung
- Schnell, seriell, sicher
- Permanent bidirektional
- Punkt-zu-Punkt oder Multislave Netzwerke
- Kompakt und kosteneffektiv
- Offener Standard
- IP-Module verfügbar (kostenfreie BDM-Lizenz)

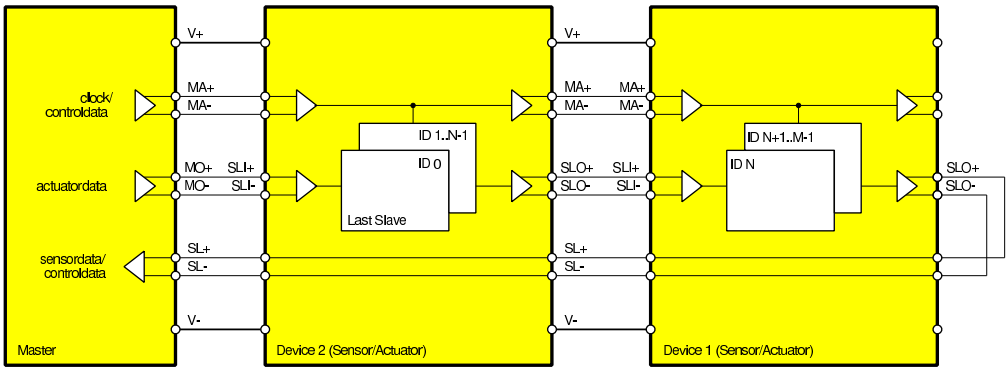
ANWENDUNGEN

- Antriebsregelungen
- Smart Sensors
- Sichere Aktoren

BLOCKSCHALTBILD



Sensorbus bzw. Punkt-zu-Punkt Netzwerk



Sensor/Aktor-Bus

Copyright © 2008 iC-Haus

KURZBESCHREIBUNG

Diese Spezifikation beschreibt ein serielles Schnittstellenprotokoll zum isochronen, schnellen und sicheren Auslesen von Sensordaten, Schreiben von Aktordaten in Echtzeit sowie zum zeitgleichen Zugriff auf die Register der Teilnehmer (Slaves).

In der Punkt-zu-Punkt Konfiguration wird nur ein Gerät (Device) mit einem oder mehreren Slaves (nur Sensoren) am Master betrieben. Die Leitung MO entfällt, die Leitung SL wird vom ersten Slave direkt zurückgeführt. Die *BiSS*-Schnittstelle ist in der Punkt-zu-Punkt Konfiguration hardware-kompatibel zur SSI-Schnittstelle und benötigt lediglich zwei unidirektionale Leitungen.

In der Bus-Konfiguration werden alle Geräte in Kette geschaltet, jedes Gerät besitzt daher zwei Steckverbinder, *BiSS*-IN und *BiSS*-OUT, deren Belegung für differentielle Signale vorgesehen sind. Die Leitung MA liefert den Takt des Masters parallel an alle Slaves, die Leitungen SLI und SLO verbinden den Master und alle Slaves zu einer Kette (MO $\rightarrow$ SLI_N, SLO_N $\rightarrow$ SLI_{N-1}, SLO_1 $\rightarrow$ SL).

An eine *BiSS* C-Mode Schnittstelle können beliebig viele Teilnehmer (Slaves) angeschlossen werden, die sowohl als Aktor als auch als Sensor arbeiten und parallel über mehrere logische Kanäle Daten übertragen können. Jeder Slave besitzt einen Speicher, der neben seiner Konfiguration auch seine Identifikation (Hersteller- und Gerätekennung) und gegebenenfalls eine Profilkennung sowie sein elektronisches Datenblatt (EDS) beinhalten kann.

Zur Datenübertragung verwendet *BiSS* C-Mode Protokoll zwei Arten von Daten mit unterschiedlichem Durchsatz:

- **Single-Cycle-Daten (SCD)** sind die primären Daten und werden in jedem Zyklus neu erzeugt und vollständig übertragen. Unterschieden werden Sensordaten, welche vom Slave zum Master übertragen werden, und Aktordaten für die Gegenrichtung. Sie finden Anwendung bei der Übermittlung sich schnell ändernder Werte wie z. B. Position, Beschleunigung oder bei der Positionsregelung von Antrieben.
- **Control-Daten (CD)** werden mit einem Bit in der fixierten Startsequenz bzw. am Taktende pro Zyklus übertragen, nutzen aber nur ein Bit je Richtung für alle Slaves. Sie erlauben das Lesen und Schreiben der Slave-Register und das Senden von Kommandos an ausgewählte oder an alle Slaves.

Die Parameter der einzelnen Datenkanäle, darunter Bit-Anzahl und CRC-Format, werden vom Slave festgelegt und sind in seinem elektronischen Datenblatt gespeichert. Dieses ist entweder eine über die Slave-Kennung referenzierte und in der Steuerung hinterlegte Datei (im XML-Format) oder es ist direkt im Speicher des Slaves hinterlegt. Alternativ können die Übertragungsparameter über applikationsspezifische Profile gewonnen werden, welche ebenfalls über eine im Slave gespeicherte Kennung referenziert werden. Die Steuerung liest mittels Registerzugriffen die Profilkennung oder die Identifikation und das EDS des Slaves und programmiert den Master entsprechend der Parameter der Slaves.

Neben der Nutzung eines “Full-*BiSS*-Masters“, der den Anschluss beliebiger Slaves erlaubt, können “Custom-*BiSS*-Master“ mit eingeschränktem Datenkanal-Parameter arbeiten und somit nur mit einem oder wenigen Slaves zusammenarbeiten. Diese Möglichkeit erlaubt Implementierungen von *BiSS*-Mastern mit geringen Ressourcen in kleinen FPGAs und mit nur wenig RAM.

Als “erster Slave“ (= “First Slave“) wird derjenige Slave bezeichnet, dessen Daten zuerst an den Master übertragen werden. Sein Ausgang SLO ist mit der Rückleitung SL direkt verbunden. Der Eingang SLI des “letzten Slaves“ (= “Last Slave“) ist mit der Leitung MO des Masters verbunden oder in der Punkt-zu-Punkt-Konfiguration auf “0“ gelegt.

FUNKTIONSBESCHREIBUNG

Das *BiSS* C-Mode Protokoll ermöglicht die gleichzeitige Übertragung von Sensordaten (SD, Sensor Data) von allen Slaves zum Master, Aktordaten (AD, Actuator Data) vom Master zu den Slaves und Steuerungsdaten (CD, Control Data) an einzelne oder mehrere Slaves.

Der *BiSS*-Frame

Die isochrone Übertragung der *BiSS*-Frames wird typisch für zyklisch abtastende Systeme verwendet. Jeder Zyklus beginnt dabei mit der Übertragung eines *BiSS*-Frames, anschließend bleibt die Schnittstelle bis zum nächsten Zyklusbeginn im Ruhezustand (MA = MO = SLO_X = “1“). Die Zyklusdauer entspricht somit mindestens der Dauer eines *BiSS*-Frames, und darf beliebig groß werden.

Der *BiSS*-Frame (Übertragungsrahmen) wird vom Master mit dem Takt MA gestartet, getaktet und beendet. Die erste steigende Flanke an MA dient dabei der Synchronisation aller Slaves, sie ermöglicht die isochrone Abtastung von Sensordaten und die isochrone Ausgabe von Aktordaten. Mit der 2. steigenden Flanke von MA setzen alle Slaves ihre SLO-Leitung auf “0“ und erzeugen damit ihr “Ack“-Signal (Acknowledge); es bleibt solange aktiv (SLO = “0“), bis das Startbit am Eingang SLI des jeweiligen Slaves eintrifft. Das Startbit wird dann synchron mit dem Takt MA von jedem Slave um einen Takt verzögert weitergereicht, während das CDS-Bit (Control Data Slave) entweder vom Slave weitergegeben oder nach den Regeln des Control-Frames gesetzt wird.

Beginnend mit dem 2. Bit nach dem Startbit und bis zum Stoppbit des *BiSS*-Frames folgt der Datenbereich (“Data Range“), der die Sensordaten von den Slaves zum Master und die Aktordaten vom Master zu den Slaves überträgt.

Der *BiSS*-Frame endet mit dem *BiSS*-Timeout. In dieser Zeit werden vom Master keine weiteren Takte an MA gesendet. Der invertierte Zustand der MA-Leitung während des *BiSS*-Timeouts ist der Zustand des CDM-Bits (Control Data Master). Am Ende der Datenübertragung setzt der Master seinen Ausgang MO auf den Ruhezustand “1“, die Slaves reichen diese an SLI empfangene “1“ an ihren Ausgang SLO weiter, sobald sie selbst den Ablauf des *BiSS*-Timeouts erkannt haben. Dies stellt sicher, dass dem Master der *BiSS*-Timeout an der Leitung SL nur dann signalisiert wird, wenn alle angeschlossenen Slaves den *BiSS*-Timeout erkannt haben. Mit Ablauf des *BiSS*-Timeouts kehren alle Slaves in den Ruhezustand zurück; alle Leitungen liegen dabei auf High-Pegel (“1“).

In der Punkt-zu-Punkt-Konfiguration (siehe Bild 1) wird das Startbit von dem letzten Slave erzeugt; er erkennt die Punkt-zu-Punkt-Konfiguration dadurch, dass sein Eingang SLI zu Beginn eines Frames bereits “0“ ist. Ebenso signalisiert der Slave den Ablauf des *BiSS*-Timeouts ohne auf einen Vorgänger zu warten. Der Ausgang des einzigen Geräts (SLO) ist in diesem Fall direkt mit dem SL Eingang des Masters verbunden.

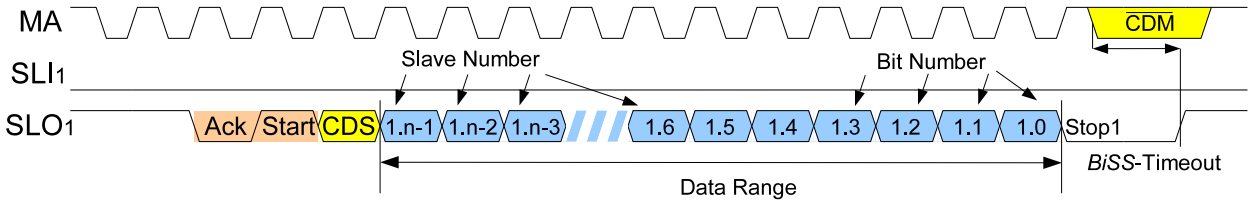


Bild 1: *BiSS*-Frame (Punkt-zu-Punkt-Konfiguration)

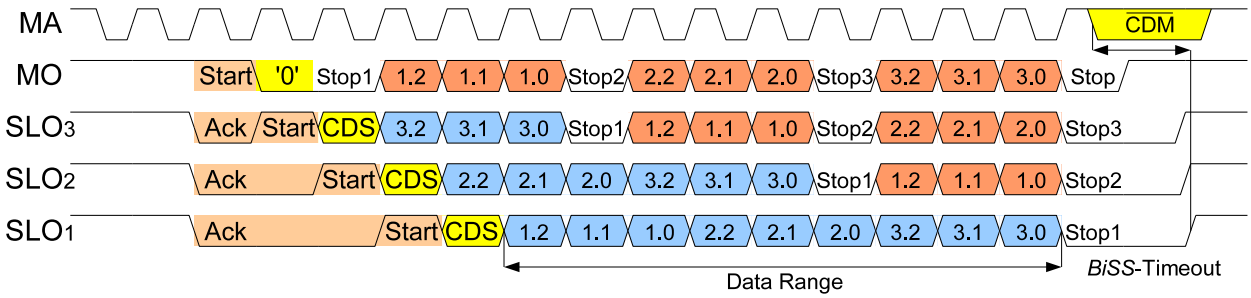


Bild 2: *BiSS*-Frame mit 3 Slaves (Bus-Konfiguration)

In jedem *BiSS*-Frame werden je Richtung ein Bit Steuerdaten (DC, Control Data) zur Kommando- bzw. Registerkommunikation übertragen. Das Control-Data-Bit des Masters (CDM, Control Data Master) wird über die Leitung MA als inverser Pegel des *BiSS*-Timeouts an alle Slaves gesendet. Der angesprochene Slave antwortet mit dem CDS-Bit (CDS, Control Data Slave), das immer im ersten Bit nach dem Startbit übertragen wird. Der Master sendet das CDS-Bit stets mit Null (MO: CDS = “0“). Die Steuerdaten-Bits mehrerer aufeinander folgender *BiSS*-Frames werden vom Master und von den Slaves zu einem Control-Frame zusammengesetzt (siehe Steuerungs-Kommunikation). Er erlaubt das Lesen und Schreiben der Slave-Register sowie das Senden von Kommandos an ausgewählte oder alle Slaves.

Verarbeitungszeit per Anforderung
Benötigt ein Slave vor der Ausgabe seiner Sensordaten zusätzliche Verarbeitungszeit, z. B. zur A/D-Wandlung oder zum Speicherzugriff, kann er diese durch Verzögerung des Startbits anfordern. Der Master erkennt das verzögerte Startbit und erzeugt die zusätzlich erforderlichen MA-Takte.

Besteht in der Punkt-zu-Punkt-Konfiguration ein Gerät (Device) aus mehreren Slaves, müssen alle außer dem letzen Slave die an SLI empfangenen Daten ihres Vorgängers zwischenspeichern und im Anschluss an die eigenen Daten an SLO senden. Der Slave mit der längsten Verarbeitungszeit bestimmt die gesamte Verarbeitungszeit; es empfiehlt sich, ihn als letzten Slave zu platzieren.

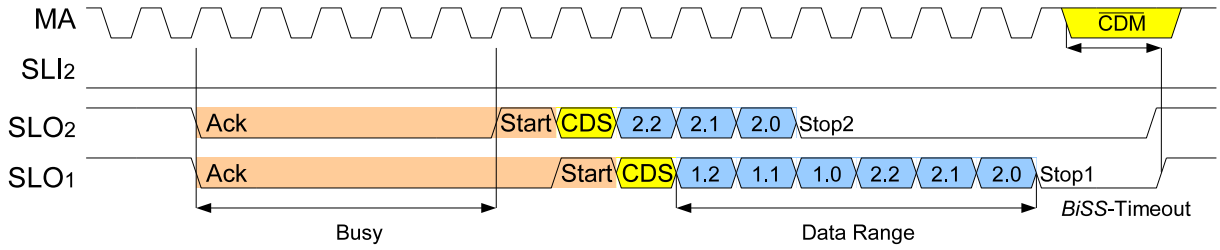


Bild 3: Anforderungen von Verarbeitungszeit (Punkt-zu-Punkt-Konfiguration)

Verarbeitungszeit per Parameter
In der Bus-Konfiguration verzögert der Master die Ausgabe des Startbits an MO. Dazu wird der Master beim Busaufbau auf die maximale Verzögerungszeit aller angeschlossenen Slaves konfiguriert. Ist die vom Slave benötigte Verarbeitungszeit variabel, muss die maximal benötigte Verarbeitungszeit eingestellt werden. Zeitangaben werden von der Steuerung mit dem aktuell eingestellten Bustakt in Takte umgerechnet, aufgerundet und im Master konfiguriert.

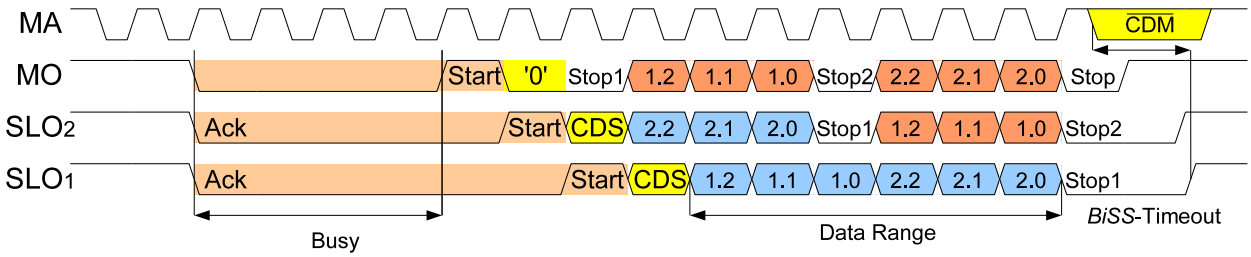


Bild 4: Parametrierte Verarbeitungszeit (Bus-Konfiguration)

Laufzeitkompensation
Die gesamte Signallaufzeit der Kette vom Takt MA bis zu dem Ausgang des ersten Slaves (Signal SL) kann vom Master ausgemessen und durch entsprechende Verschiebung der Abtastung des Slavesignals kompensiert werden. Dazu muss die Leitung MA die gleiche Leitungstopologie besitzen und mit den gleichen Leitungstreibern versehen sein wie die Kette SLI-SLO. Damit erhalten MA und SLI die gleiche laufzeitbedingte Verzögerung und bleiben synchron. Zusätzlich wird die Datenausgabe an MO verzögert, falls die Leitungslaufzeit größer als eine Periode ist. Zur Ermittlung der Laufzeit wird die Verzögerung von der zweiten steigenden MA-Flanke bis zur fallenden Flanke des Ack-Bits der Slave-Antwort (SL: “Ack“) verwendet; sie ist im Idealfall Null.

Die Laufzeitkompensation ermöglicht eine beschleunigte Kommunikation mit hohe Datenraten von typisch 10 MBit/s. Sie wird mit jedem *BiSS*-Frame erneut durchgeführt und berücksichtigt so auch alterungs- oder temperaturbedingte Drifteffekte.

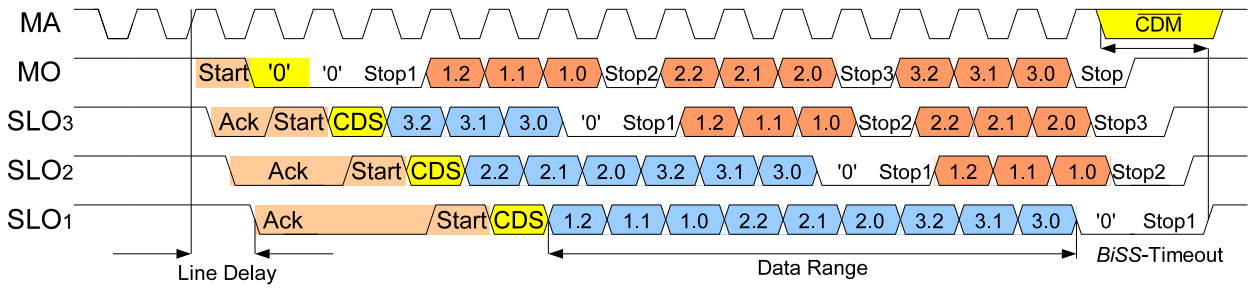


Bild 5: Laufzeitkompensation durch den Master

Bus-Reset
Nach dem Einschalten oder einem Fehler muss der Master vor der Datenübertragung eine Pause von 40 µs einhalten. Dies stellt sicher, dass der *BiSS*-Timeout abgelaufen ist und alle Slaves für die Datenübertragung bereit sind.

In der Punkt-zu-Punkt Konfiguration ist der letzte Slave vor dem ersten MA-Impuls nicht definiert, was dazu führt, dass die SL Leitung gegebenenfalls auf Null bleibt. Der Master muss entweder einen Puls an MA erzeugen oder aber den ersten Zyklus ohne Berücksichtigung von SL starten. Slaves die eine längere Konfigurationsphase benötigen, lehnen innerhalb dieser Phase alle Registerzugriffe ab.

SENSOR- UND AKTORDATEN-KOMMUNIKATION

Der Datenbereich
Der Datenbereich dient der Übertragung der Sensordaten von den Slaves zum Master und der Aktordaten vom Master zu den Slaves. Der gesamte Datenbereich ist in logische Datenkanäle aufgeteilt. Die Position und Länge der einzelnen Datenkanäle wird für jeden Slave in seinen Parametern beschrieben. Ein Slave kann keinen, einen oder mehrere Datenkanäle für Sensor- und/oder Aktordaten besitzen.

Der Master muss mit den Parametern der einzelnen Datenkanäle in deren Reihenfolge und der Reihenfolge der angeschlossenen Slaves konfiguriert werden, um ihre Bits korrekt zuordnen zu können. Nur bei korrekter Konfiguration der Datenkanäle im Master lassen sich gültige Daten übertragen; die je Datenkanal enthaltenen Sicherungsbits dienen der Erkennung von Übertragungsfehlern und Bit-Verschiebungen.

Die Daten des ersten Slaves erreichen den Master unmittelbar nach dem CDS-Bit. Die weiteren Datenkanäle folgen ohne Trennung durch Start- oder Stoppbits direkt aufeinander, so dass sich die Länge des Datenbereichs aus der Summe der Längen aller Datenkanäle ergibt. Die Slave-Nummerierung erfolgt in der Reihenfolge der Datenübertragung, sie wird deshalb in umgekehrter Reihenfolge zur Signalrichtung SLI → SLO gezählt.

Die Slaves können im *BiSS*-Frame den *BiSS*-Timeout nur dann korrekt signalisieren, wenn zu Beginn des *BiSS*-Timeouts alle SLO-Leitungen den Pegel “0“ besitzen. Dazu sendet der Master vor jedem Datenkanal eine führende Null, welche bei Zyklusende als Stoppbit am Slaveausgang anliegt. Im Fehlerfall kann der *BiSS*-Frame zu jedem Zeitpunkt mit einer Taktpause von 40 µs abgebrochen werden.

Die empfangenen Single-Cycle Aktordaten Daten werden von den Aktoren im folgenden *BiSS*-Frame als Sensordaten zurückgesendet. Bei Sensordatenkanälen sendet der Master an MO den Wert “0“.

Die Sensordaten werden mit der ersten steigenden Flanke des *BiSS*-Frames erfasst und, im Fall von Single-Cycle-Daten, noch im selben Zyklus gesendet. Der Master steuert die Ausgabe der Aktordaten so, dass diese mit dem letzten MA-Takt korrekt in den Schieberegistern der Slaves stehen; sie werden mit dem Ablauf des *BiSS*-Timeouts gültig. Für die isochrone Arbeitsweise ist es empfehlenswert, die Aktordaten ebenfalls mit der ersten steigenden Taktflanke des folgenden Zyklus an den Ausgängen auszugeben.

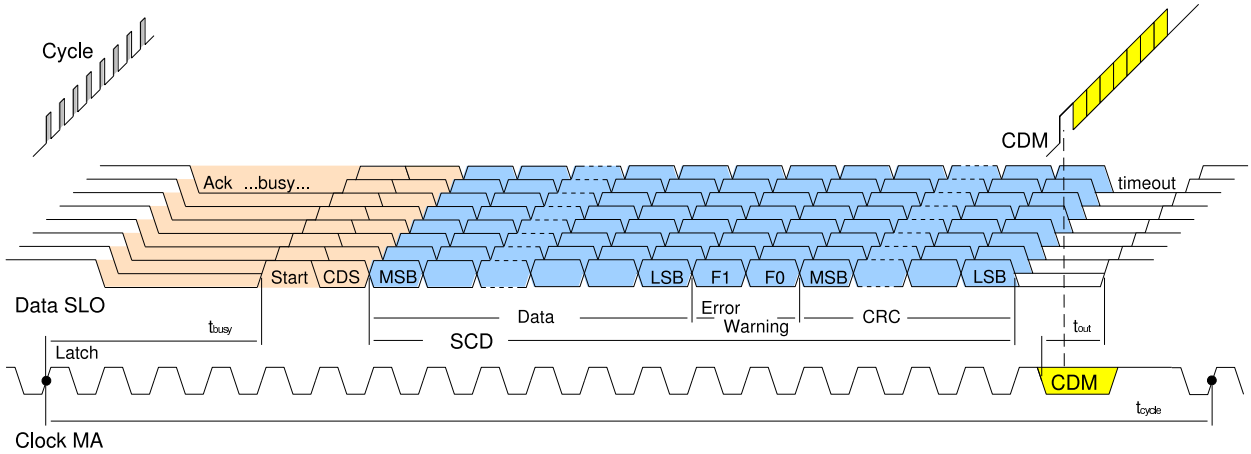


Bild 6: Der Datenbereich und die Datenkanäle

Die Datenkanäle
Ein Datenkanal ist eine logische Einheit, die der gesicherten Datenübertragung dient und die zugehörigen Parameter und Dateninhalte beschreibt. Jeder Kanal ist entweder ein Sensordatenkanal (Slave → Master) oder ein Aktordatenkanal (Master → Slave) und enthält schnelle (SCD) Daten. Ein Datenkanal belegt im Datenbereich des *BiSS*-Frames und im Speicher des Masters die parametrierte Länge (Anzahl der Daten- und Sicherungsbits). Im elektronischen Datenblatt besitzt jeder Datenkanal einen Abschnitt, der die zur gesicherten Datenübertragung notwendigen Parameter und die Beschreibung der Dateninhalte enthält. Entsprechend dieser Parameter konfiguriert die Steuerung einen Datenkanal im Master.

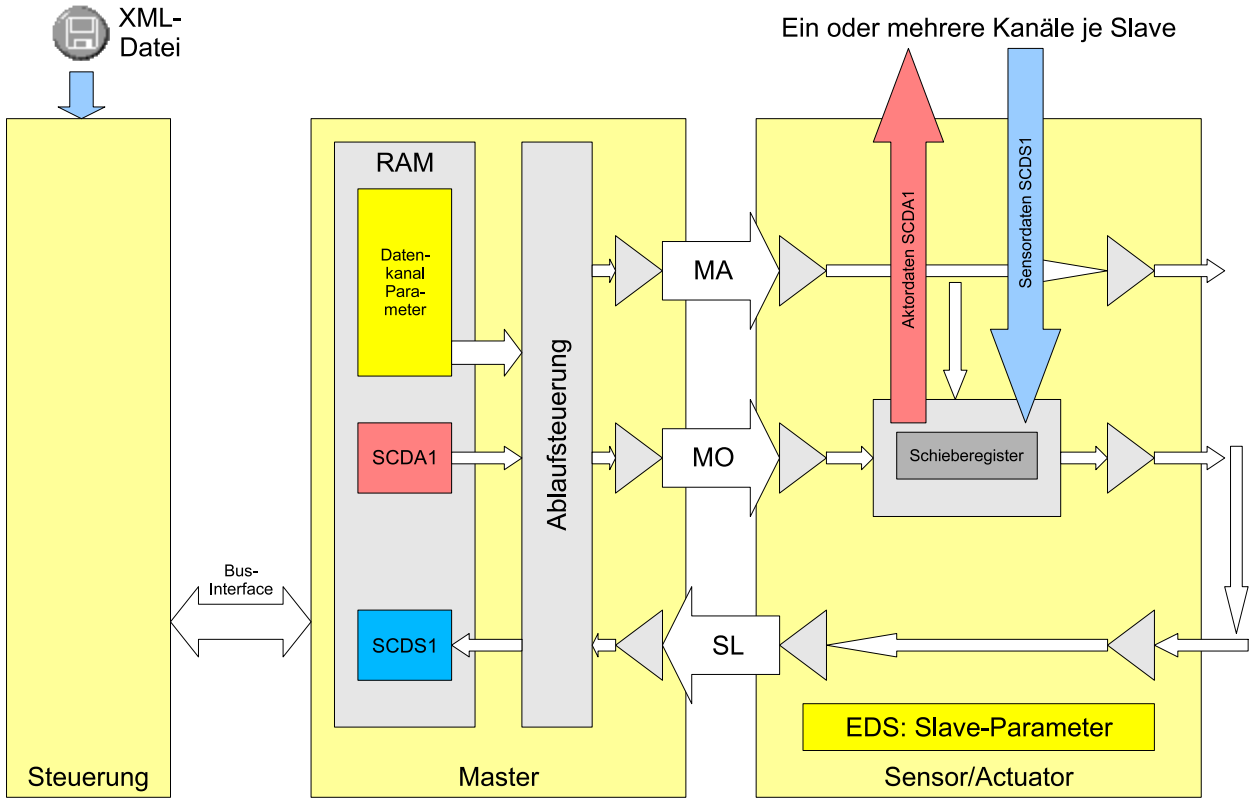


Bild 7: Konfiguration und Übertragung von Sensor- und Aktordaten

Die Datenkanal-Parameter
Folgende Parameter für Datenkanäle sind definiert:

- Übertragungsrichtung und Typ
  1. SCDS (Single-Cycle-Sensor-Data)
  2. SCDA (Single-Cycle-Actuator-Data)
- Bitanzahl (1. . . 64)
- Verarbeitungszeit (0 µs. . . t_{busy_s} bzw. 0 µs. . . t_{busy_m})
- Datenausrichtung (links- oder rechtsbündig)
- CRC-Polynom (für 0. . . 8 CRC-Bits)
- CRC-Startwert (für 0. . . 8 CRC-Bits)

Single-Cycle-Daten (SCD)
Ein Datenkanal mit Single-Cycle-Daten wird für schnelle und zyklische Sensor- bzw. Aktordaten verwendet, welche komplett in einem Zyklus übertragen werden. SCD benötigen keine Adressierung, besitzen eine parametrierbare Länge von 1 bis 64 Datenbits und eine CRC-Sicherung von 0 bis 8 Bits.

Bild 8: Ein SCD-Datenkanal

Verarbeitungszeit für Single-Cycle-Daten
Die Verarbeitungszeit für SCD beginnt mit der ersten steigenden MA-Flanke zeitgleich in allen Slaves. Die Framelänge wird daher nur von der größten Verarbeitungzeit aller Slaves bestimmt. Nur diese Zeit ist für den Master sichtbar. In den Parametern wird die Verarbeitungszeit wahlweise als Zeiteinheit oder in MA-Takten angegeben; eine Abhängigkeit von der Framelänge ist zulässig. Die maximale Verarbeitungszeit für SCD ist t_{busy_s} (siehe Kenndaten).

Der Datenwert
Alle Datenwerte werden mit dem höchstwertigen Bit zuerst übertragen (“MSB first“). Ein Datenwert selbst kann aus mehreren Bitgruppen bestehen, z. B. einem Messwert und mehreren Fehlerflags. Zusammensetzung und Ausrichtung sind im elektronischen Datenblatt des Slave definiert. Bild 11 zeigt beide Varianten anhand eines Datenwerts von 13 Bit, der in einem 16 Bit breiten Wort im Master abgelegt wird.

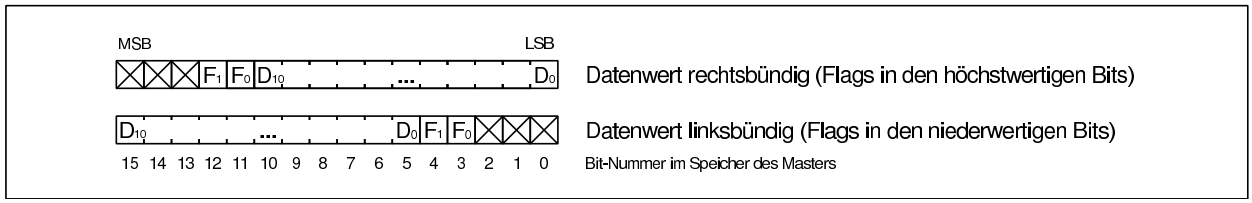


Bild 9: Ausrichtung des Datenwerts (Alignment)

Ungültige Werte
Es wird empfohlen, dass zur Erkennung einer fehlerhaft konfigurierten Übertragungsrichtung eines Datenkanals für Single-Cycle-Daten der Datenwert Null (“0“) als ungültig reserviert wird. Den Datenwert Null überträgt der Master auch, wenn für einen Aktordatenkanal beim Beginn des *BiSS*-Frames keine neuen Daten vorliegen. Üblicherweise wird der Wert Null (“0“) in den gültigen Daten vermieden, indem mindestens ein Bit des Datenwerts als “0“-aktives Fehlerbit verwendet wird.

CRC-Sicherung
Jeder Datenkanal kann zusätzlich zu seinem Datenwert eine Übertragungssicherung mittels CRC verwenden. Die Eigenschaften der CRC-Sicherung werden in den Parametern des Datenkanals angegeben. Das CRC-Polynom gibt auch die Anzahl der übertragenen CRC-Bits an; 0 bis 8 Bit sind möglich. Die Berechnung erfolgt über die Datenbits beginnend mit dem höchstwertigen Bit. Ein Startbit gibt es nicht. Die CRC-Sicherungsbits werden immer invers mit dem höchstwertigen Bit zuerst übertragen.

Der Startwert für die CRC-Berechnung ist in der Regel Null, kann aber mit einem konfigurierbaren Startwert versehen werden, falls Master und Slave diese Funktion unterstützen. Die Konfiguration des CRC-Startwerts erlaubt die eindeutige Zuordnung eines Datenwerts zu einem Slave, da bei fehlerhafter Konfiguration des Masters oder vertauschter Reihenfolge die CRC-Prüfung fehlschlägt. Die Steuerung vergibt einen Startwert für jeden Datenkanal und schreibt diesen mittels Registerzugriff in den Slave; die entsprechende Registeradresse wird im elektronischem Datenblatt angegeben.

Werden zur Sicherung eines Datenkanals andere Prüfverfahren verwendet oder ist die maximale Anzahl der Bits für die CRC-Prüfung nicht ausreichend, so kann die CRC-Prüfung bzw. CRC-Erzeugung per Parametrierung deaktiviert werden (CRC-Polynom = 0). Die Sicherungsbits werden dann innerhalb der (maximal 64 Bit pro Datenkanal) normalen Datenbits übertragen und im Speicher des Masters abgelegt. Die Prüfung der Sicherungsbits kann nun mittels Software erfolgen.

STEUERUNGS-KOMMUNIKATION

Der Control-Frame
Der Control-Frame erlaubt das gesicherte und bestätigte Lesen und Schreiben der Register eines Slaves sowie das gesicherte und bestätigte Senden von Kommandos an ausgewählte oder alle Slaves. Der Control-Frame ergibt sich aus einer Anzahl von *BiSS*-Frames, die Sensor- oder Aktordaten erzeugen und übertragen.

Die Ausführung des Registerzugriffs oder des Kommandos erfolgt immer am Ende des Zyklus des letzten CDM-Bits, und zwar mit Ablauf des *BiSS*-Timeouts im Slave. Der Control-Frame kann zu jedem Zeitpunkt mit der Übertragung von 14 “0“-Bits abgebrochen werden. Vor dem Startbit eines Control-Frames müssen mindestens 14 Zyklen mit CDM = “0“ liegen.

Slave-Adressierung mittels ID-Belegung
Im Gegensatz zur Sensor- und Aktordatenkommunikation erfordert die Steuerungskommunikation eine eindeutige Adressierung. Die Adresse der Slaves (“ID“) wird entsprechend der Reihenfolge in der Kette dynamisch zugeordnet. Die ID-Belegung erfolgt automatisch bei jedem Control-Frame für die ersten 8 IDs durch setzen der ID-Lock-Bits (IDL).

Die ID-Belegung beginnt in jedem Control-Frame nach dem Startbit des Masters (CDM = “1“). Dazu wartet jeder Slave das erste freie CDS-Bit (SLI = “0“) ab und setzt es (SLO = “1“).
Durch die Reihenfolge in der Kettenschaltung können die CDS-Bits von dem letzen Slave zuerst belegt werden, was dazu führt, dass die IDs in umgekehrter Reihenfolge der Slave-Nummern vergeben werden.

Das Beispiel in Bild 12 zeigt die ID-Belegung von vier Devices, die jeweils einen Slave (Device 3 + 4) bzw. vier Slaves (Device 1+2) enthalten. Da die verfügbaren acht IDs nicht für zehn Slaves ausreichen, signalisieren die vorderen Slaves mit IDL8 = “1“, dass sie keine ID bekommen haben.

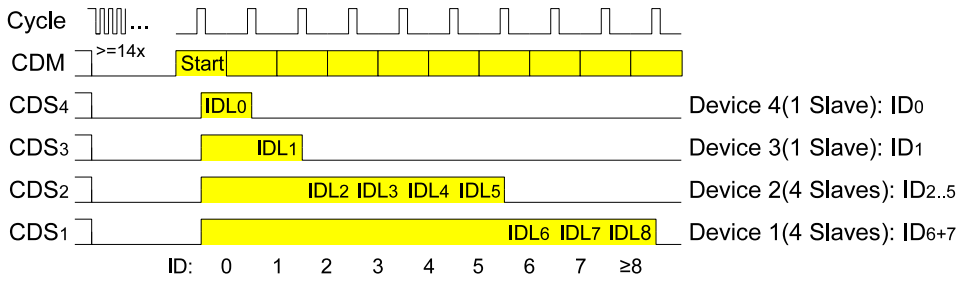


Bild 10: Beispiel einer ID-Belegung

CRC-Sicherung
Auch die Steuerungs-Kommunikation nutzt zur Übertragungssicherung eine Prüfsumme. Das verwendete CRC-Polynom ist:
$X^4 + X^1 + X^0$
Es stehen damit 4 CRC-Bits zur Verfügung, welche invertiert übertragen werden. Die Berechnung erfolgt mit dem Startwert Null über die Adressierungssequenz bzw. die Datenbits beginnend mit dem höchstwertigen Bit immer ausschließlich des Startbits.

Hinweis:
Vor jedem Control-Frame müssen mindestens 14 Bits mit CDM = “0“ übertragen werden.

Die Kommandos
Kommandos können gleichzeitig an 8 Slaves in beliebiger Kombination (Adressiert) oder an alle Slaves am Bus gemeinsam (Broadcast) gesendet werden.

Ein Kommando-Frame beginnt mit einem Startbit und dem Steuerungsbit CTS (Control Select), welches auf “0“ ist (CTS = “0“: Kommando). Der Master wählt mit den folgenden IDS-Bits (ID-Select) den oder die angesprochenen Slaves aus, wobei die bitweise Kodierung eine beliebige Kombination der ersten 8 IDs erlaubt (z.B. $IDS_{0..7}$ = “1000 0100“ entspricht ID = “0“ und “5“). Anschließend sendet der Master eines der vier Kommandos (CMD = “00“ / “01“ / “10“ / “11“) und schließt die Adressierungssequenz mit einer 4-Bit CRC ab. Gleichzeitig mit der Adressierungssequenz erfolgt die ID-Belegung der Slaves mittels der IDL-Bits (ID-Lock). Die angesprochenen Slaves bestätigen auf das folgende Startbit (17. Bit) den korrekten Empfang des Kommandos, indem jeder adressierte Slave das entsprechende IDA-Bit (ID-Acknowledge) setzt. Der Master oder die Steuerung kann die IDA-Bits mit den IDS-Bits bitweise vergleichen (IDS = IDA) und abhängig vom Ergebnis das Kommando durch Senden eines EX-Bits (Execute, CDM=“1“) ausführen lassen oder durch Senden von 14 “0“-Bits die Ausführung abbrechen.

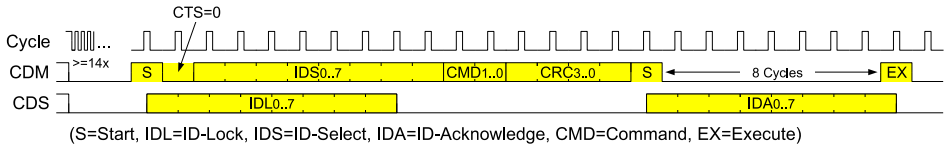


Bild 11: Der Kommando-Frame (Adressiert)

Des weiteren ist es möglich, Kommandos an alle Slaves zu senden (Broadcast), indem kein Slave direkt angesprochen wird, d.h. keines der IDS-Bits gesetzt ist. Broadcast-Kommandos werden auch dann von den Slaves ausgeführt, wenn sie aktuell keine ID belegen. Als Besonderheit des Broadcast-Kommandos kann der *BiSS*-Frame hierbei auf zwei Takte und *BiSS*-Timeout verkürzt werden.

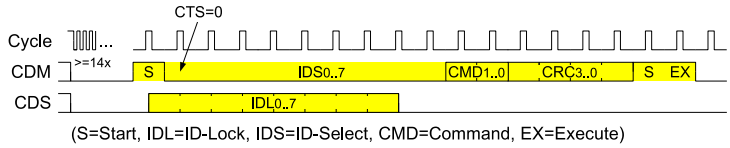


Bild 12: Kommando-Frame (Broadcast)

Aus den zwei Kommandobits und den zwei Adressierungsmöglichkeiten ergeben sich somit acht Kommandos. Einige dieser Kommandos sind für *BiSS* C-Mode festgelegt und müssen von jedem *BiSS* C-Mode Slave interpretiert werden.

Reserviert
(CMD = “00“, Broadcast):
Der Broadcastbefehl “00“ ist für zukünftige Erweiterungen reserviert.

Reserviert
(CMD = “01“, Broadcast):
Der Broadcastbefehl “01“ ist für zukünftige Erweiterungen reserviert.

Buskoppler abschalten
(CMD = “10“, Broadcast):
Alle Buskoppler werden auf “bypass“ geschaltet. Dieses Kommando ist nur für Buskoppler relevant. Alle anderen Slaves ignorieren es.

Reserviert
(CMD = “11“, Broadcast):
Der Broadcastbefehl “11“ ist für zukünftige Erweiterungen reserviert.

Reserviert
(CMD = “00“, Adressiert):
Der adressierte Befehl “00“ ist für zukünftige Erweiterungen reserviert.

Reserviert
(CMD = “01“, Adressiert):
Der adressierte Befehl “01“ ist für zukünftige Erweiterungen reserviert.

Buskoppler einschalten bzw. frei
(CMD = “10“, Adressiert):
Ein adressierter Buskoppler schaltet von “bypass“ auf “Leitungsbetrieb“ um. Für andere Slaves ist dieses Kommando frei definierbar.

Frei
(CMD = “11“, Adressiert):
Dieses Kommando ist für jeden Slave frei definierbar.

Die Kommandos sind für den Busaufbau erforderlich. Mit normalen *BiSS*-Frames werden die Datenkanal-Parameter der Slaves aus dem jeweiligen EDS gelesen.

Mit dem Broadcastbefehl “10“ werden alle Buskoppler auf “bypass“ geschaltet. Etwaige Störungen wie Kurzschlüsse oder Unterbrechungen können somit ab dem ersten Slave erkannt und isoliert werden. Die Slaves eines Busses werden dann nacheinander folgend, einzeln, von “bypass“ auf “Leitungsbetrieb“ umgeschaltet und auf eine Antwort der nachfolgenden Instanz/Verbindung hin kontrolliert.

Die freien Kommandos können genutzt werden, um beliebige Aktionen (z.B. Nullsetzen bei Drehgebern) zeitgleich bei ausgewählten Slaves auszuführen.
Werden mehr als zwei freie Kommandos benötigt, empfielt es sich, das Kommando erst über jeweils einen Registerzugriff in den entsprechenden Slaves zu selektieren und dann das Kommando den beteiligten Slaves zur zeitgleichen Ausführung zu senden.

Mehr als 8 IDs
Werden mehr als die 8 möglichen IDs benötigt, setzen die Slaves das 9. Bit (ID-Lock, IDL8). Es signalisiert der Steuerung, dass es weitere Slaves gibt, die nicht adressiert werden können.

Hinweis: Der Master kann anhand der ID-Lock-Bits (IDL0..8) nur erkennen, wie viele IDs aktuell belegt sind, nicht die Zuordnung zu den Slaves.

Die Register-Kommunikation
Die Lese- und Schreibzugriffe auf die Register eines Slaves erfolgen mit einem Control-Frame, dessen Steuerungsbit CTS gesetzt ist (CTS = “1“: Registerzugriff).

Der Register-Frame beginnt mit der Adressierungssequenz. Hier sendet der Master die Slave-ID (3 Bit), gefolgt von der Registeradresse (7 Bit) und einer 4-Bit-CRC. Durch die binäre Kodierung können damit 8 Slaves mit je 128 Register (= 128 Byte) adressiert werden. Gleichzeitig mit der Adressierungssequenz erfolgt die Belegung der IDs durch die Slaves (siehe ID-Belegung).

Die beiden nächsten CDM-Bits, R- und W-Bit, bestimmen, ob es sich um einen Lesezugriff (RW = “10“) oder Schreibzugriff (RW = “01“) handelt. Beide Bits müssen invers zueinander sein und werden zur Bestätigung von dem adressierten Slave an den Master zurückgesandt. Sie werden bei der Berechnung des CRCs nicht verwendet.

Der Schreibzugriff
Bei dem Schreibzugriff haben die beiden Lese-/Schreib-Bits den Wert RW = “01“. Danach folgen Startbit, 8 Datenbits, 4-Bit CRC und ein Stopbit.
Die 8 Datenbits werden beim Schreiben mit einer 4-Bit CRC gesichert, zudem werden die übertragenen Registerdaten zurückgesandt.

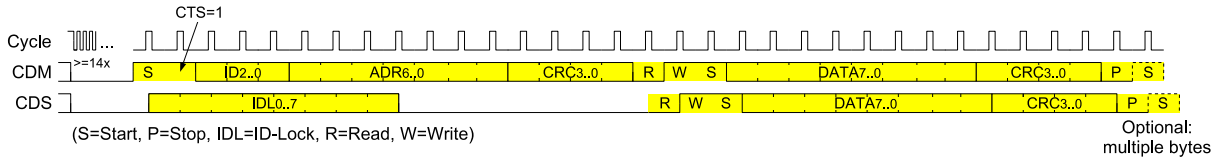


Bild 13: Register-Schreibzugriff

Der Lesezugriff
Bei dem Lesezugriff haben die beiden Lese-/Schreib-Bits den Wert RW = “10“. Danach folgen Startbit, 12 “0“-Bits und ein Stoppbit. Die Registerdaten werden beim Lesen mit einer 4-Bit CRC gesichert.

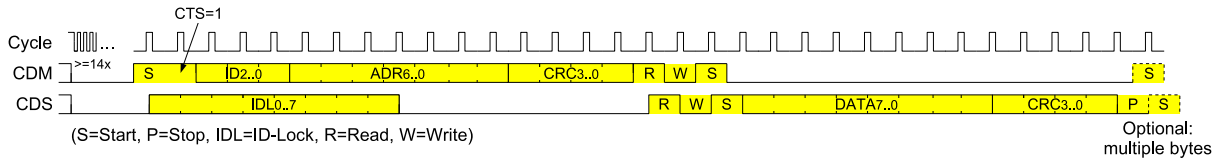


Bild 14: Register-Lesezugriff

Sequentieller Registerzugriff
Es ist möglich, in einem Zugriff mehrere aufeinander folgende Register zu lesen oder zu schreiben. Der Master sendet dazu unmittelbar nach dem Stoppbit des ersten Datenwerts ein erneutes Startbit (CDM = “1“). Beim Schreibzugriff folgen das Datenbyte, die 4 CRC-Prüfbits und das Stoppbit. Beim Lesezugriff werden lediglich 13 “0“-Bits – inklusive Stoppbit – gesendet. Der Slave erhöht intern mit jedem Schreib- oder Lesezugriff die Registeradresse um 1 (Auto-Inkrement). Mit einem Zugriff können maximal 64 Register in Folge gelesen oder geschrieben werden. Sequentielle Zugriffe über die Registeradresse 63 (0x3F) bzw. 127 (0x7F) hinaus sind nicht erlaubt. Der sequentielle Zugriff endet, wenn an CDM kein weiteres Startbit folgt.

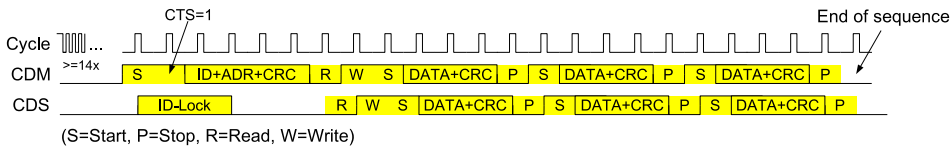


Bild 15: Mehrere Register schreiben

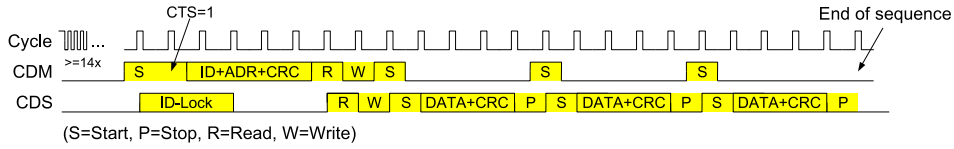


Bild 16: Mehrere Register lesen

Nicht implementierte Register
Die Register eines *BiSS* C-Mode Slaves können "verboten" oder "nicht implementiert" sein. In diesem Fall lehnt der Slaves den Zugriff auf das Register ab, indem er das über CDS zurückgesandte W-Bit invertiert. Beim Schreibzugriff führt dies zu RW = "00", beim Lesezugriff zu RW = "11".

Werden mehrere Register in Folge geschrieben oder gelesen und ist das folgende Register nicht implementiert oder im Auto-Inkrement nicht ansprechbar, wird das Stoppbit über CDS invertiert, d.h. als "1"-Bit, zurückgegeben. Der Zugriff auf ein nicht implementiertes Register beendet den sequentiellen Zugriff.

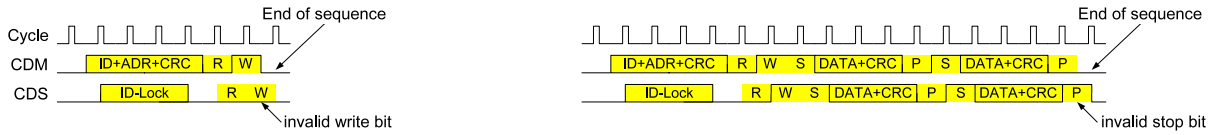


Bild 17: Zugriffe auf nicht implementierte Register (hier Schreiben)
a) erstes Register b) weiteres Register einer Schreibsequenz

Verarbeitungszeit beim Registerzugriff
Benötigt der Slave beim Lesen oder Schreiben von Registern zusätzliche Verarbeitungszeit, kann er diese für jedes Byte individuell dadurch anfordern, dass er den Beginn seines Startbits verzögert. Der Master wiederholt während dieser Zeit das Startbit. Wird das Startbit nicht innerhalb der Zeit t_{busy_r} übertragen, wird der Registerzugriff als ungültig abgebrochen, indem der Master das Wiederholen des Startbits abbricht.

Ein Registerzugriff benötigt üblicherweise Verarbeitungszeit, wenn ein externer Speicher adressiert wird. Wird nach dem letzten übertragenen Register Verarbeitungszeit benötigt, z.B. zur Speicherung des Werts, kann dies nicht über *BiSS* C-Mode signalisiert werden. Der Slave kann aber am Beginn des folgenden Registerzugriffs die noch benötigte Verarbeitungszeit anfordern. Die maximale Verarbeitungszeit beim Registerzugriff beträgt t_{busy_r} (siehe Kenndaten).

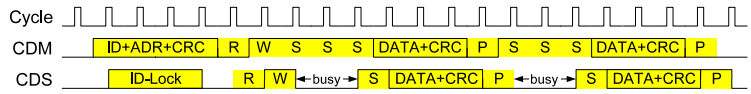


Bild 18: Schreibzugriff mit Verzögerungszeit

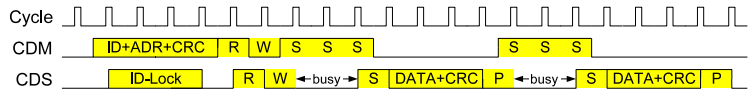


Bild 19: Lesezugriff mit Verzögerungszeit

Hinweis:
Das CDS-Bit wird im *BiSS*-Frame vor dem CDM-Bit übertragen. Beim Registerzugriff muss der Master das über CDS empfangene Startbit unverzüglich auswerten und im gleichen *BiSS*-Frame mit dem CDM-Bit antworten. Dieses ist entweder "1'", bei angeforderter Verarbeitungszeit oder wenn das zu sendende MSB es erfordert, oder "0". Falls die Auswertung am Ende des *BiSS*-Frames nicht möglich ist, kann das CDM-Bit abhängig vom CDS-Bit (z.B. invertiert) gesendet werden.

SLAVE-REGISTER

Die Registerbelegung
Der Adressraum eines Slaves ist durch den Register-Frame auf 128 Bytes begrenzt. Üblicherweise verfügt ein Slave aber zusätzlich zu seinem RAM über einen nur lesbaren (ROM) oder wiederbeschreibbaren (E²PROM) Speicher, der neben der Einschalt-Konfiguration der Slave-Register auch das elektronische Datenblatt (EDS) und gegebenenfalls weitere Daten enthalten kann. Um den Zugriff auf den Speicher zu ermöglichen, werden jeweils 64 Byte zu einer Bank zusammengefasst. Der Zugriff auf eine Bank erfolgt über eine Bankumschaltung und die Adressen 0 bis 63. Die Register ab Adresse 64 sind ohne Bankumschaltung direkt zugänglich. Die Registerbelegung zeigt Tabelle 1.

Adresse (Dezimal)	Adresse (Hexadezimal)	Name	Größe	Bemerkungen
0 .. 63	0x00 .. 0x3F	Register Bank	64 Bytes	
64	0x40	Bankauswahl	0..8 Bits (1 Byte)	1) 4)
65	0x41	EDS-Bank	0..8 Bits (1 Byte)	1) 2)
66 .. 67	0x42 .. 0x43	Profilkennung	16 Bits (2 Bytes)	2) 3)
68 .. 71	0x44 .. 0x47	Seriennummer	32 Bits (4 Byte)	2) 3)
72 .. 119	0x48 .. 0x77	Slave-Register	48 Bytes	
120 .. 125	0x78 .. 0x7D	Geräte-Kennung	48 Bits (6 Bytes)	2) 3)
126 .. 127	0x7E .. 0x7F	Hersteller-Kennung	16 Bits (2 Bytes)	2) 3)

1) Wird keine Bankumschaltung verwendet, lehnt der Slave den Registerzugriff ab.
2) Register ist gegen unbeabsichtigtes Beschreiben geschützt.
3) Der Wert wird als Big Endian, d.h. mit dem höchstwertigen Byte an der niederwertigsten Adresse abgelegt.
4) Ungenutzte Registerinhalte sind mit “0“ zu füllen.

Tab. 1: Tabelle Registerbelegung

Register Bank
Jede der maximal 256 Banken kann je nach Slave weitere 64 Register oder 64 Byte Speicher beinhalten; somit ist maximal ein 16 KByte (256 * 64 = 16384 Bytes) großer Speicherbereich adressierbar.

Bankauswahl
Das Bankauswahl-Register an Adresse 64 wählt eine Register-Bank aus bis zu 256 Banken aus und blendet diese an den Registeradressen 0 bis 63 ein.

EDS-Bank
Da die Größe des zur Einschaltkonfiguration benötigten Speichers slavespezifisch ist, stellt der Slave in seinem Register EDS-Bank an Adresse 65 die Bank-Nummer des Beginns des frei verfügbaren Speichers zur Verfügung, welcher für das elektronische Datenblatt (EDS) verwendet wird. Das EDS ist beschrieben in:
BiSS-Interface - Electronic Data Sheet Definition.

Profilkennung
Um die Austauschbarkeit und Kompatibilität von *BiSS* C-Mode Slaves zu vereinfachen, werden für häufig benötigte Gerätetypen Profile definiert. Zu einer Profil-Definition gehören alle Datenkanalparameter eines Slaves. Ebenso enthalten sind Zusatzinformationen wie Namen oder Maßeinheiten von Datenkanälen und Registern. Im elektronischen Datenblatt gibt es einen Abschnitt, in dem im Profile definierte Parameter abgelegt werden.
Jedes Profil besitzt eine eindeutige, 16 Bit breite Kennung, anhand derer die Steuerung die Parameter zuordnen kann. Verwendet ein Slave kein vordefiniertes Profil, so sind die Register 66 und 67 nicht implementiert oder beinhalten den Wert Null (“0“). Profil-Kennungen werden ausschließlich von iC-Haus vergeben, neue Profile können unter support@biss-interface.com beantragt werden.

Slave-Register
Der Inhalt dieser 48 Bytes sind slavespezifisch. Sie sollten für Register verwendet werden, welche ohne Bankumschaltung direkt gelesen oder geschrieben werden müssen (z.B. Statusregister).

Geräte- und Hersteller-Kennung
Jeder Bus-Teilnehmer (Slave) stellt dem System eine eindeutige Kennung zur Verfügung, die aus Hersteller- und Geräte-Kennung besteht. Sie kann unter den Registeradressen 120 bis 127 (6 Byte für die Baugruppe, 2 Byte für den Hersteller) ausgelesen werden und erlaubt die Zuordnung eines in der Steuerung gespeicherten EDS zu dem Slave und die Überprüfung des Slave-Typs durch die Steuerung.

Die Hersteller-Kennung (Manufacturer-ID) ist eine spezielle für *BiSS* C-Mode erforderliche Kennung des Herstellers des Slave, sie wird von iC-Haus (support@biss-interface.com) vergeben.

Die Geräte-Kennung (Device-ID) muss vom Hersteller des Slaves festgelegt werden. Sie muss für jeden Slave in Verbindung mit der Hersteller-Kennung eindeutig sein und dient zur Referenzierung des elektronischen Datenblatts als XML-Datei.

Wird ein Slave in verschiedenen Varianten (z. B. unterschiedliche Datenlängen, verschieden Datenkanal-Konfigurationen) ausgeliefert, muss jede Variante eine unterschiedliche Device-ID besitzen.

Die Umschaltung der Variante kann auch dynamisch im Betrieb (durch Registerzugriffe oder Kommandos) erfolgen, erfordert aber die unmittelbare Umschaltung der Geräte-Kennung.
Das *BiSS*-Timeout kann bei Slaves ohne automatische Anpassung an T_{MA} durch Setzen der Bits 0 bis 2 an Adresse 124 verkleinert werden. Slaves mit automatischer Anpassung des *BiSS*-Timeouts haben an dieser Stelle ein ROM.

Seriennummer
Zudem ist im Registerbereich eine vom Hersteller des Slaves vergebene Seriennummer mit bis zu 32 Bit vorgesehen; sie muss zusammen mit Hersteller- und Geräte-Kennung einen global eindeutigen Wert besitzen. Dies ist vor allem bei mehreren Slaves gleichen Typs wichtig, um die Vertauschung oder den Ausfall eindeutig erkennen und zuordnen zu können.

Für die Hersteller-Kennung, Geräte-Kennung und Seriennummer sind die Werte Null (0) und der maximale Wert (alle Bits “1“) reserviert. Besitzt ein Slave keine Seriennummer, ist diese entweder auf Null (0 = “nicht vorhanden“) zu setzen oder die Register nicht zu implementieren.

INITIALISIERUNG

BiSS C-Mode erlaubt eine Vielzahl von Einstellungen in den Slave-, Master- und Buskonfigurationen. Diese Konfigurationen werden bei der Inbetriebnahme des *BiSS*-Systems ermittelt und gesetzt.

Der *BiSS* C-Mode Busaufbau
Der Bus-Aufbau wird durch Software auf der Masterseite durchgeführt. Von der Steuerung wird die Taktfrequenz $1/T_{MA}$ entsprechend der verwendeten Hardware und der Leitungstopologie vorgegeben (siehe BiSS-Interface - Physical Layer). Während des Busaufbaus wird die minimale Zykluszeit und der zulässige Bereich der Taktfrequenz ermittelt.

Zunächst werden alle Datenkanäle deaktiviert. Dann werden alle vorhandenen Slaves nacheinander detektiert, ihre Datenkanalparameter ermittelt und der Master entsprechend konfiguriert.

Im Fehlerfall kann der Busaufbau wiederholt werden, wobei die Datenkanäle fehlerhafter Slaves nicht mehr aktiviert werden. Störungen wie Kurzschlüsse oder Unterbrechungen können nur mit Buskopplern isoliert werden.

Des weiteren wird während des Busaufbaus

- der minimale *BiSS*-Timeout der Slaves ermittelt und ggf. konfiguriert,
- der maximal zulässige MA-Takt (in Abhängigkeit vom Master und den Slaves) ermittelt,
- die CRC-Startwerte im Master und in den Slaves konfiguriert,
- die Verarbeitungszeit per Parameter im Master konfiguriert sowie
- die minimale Zyklusdauer (aus der Anzahl der Bits im Datenbereich, der Verarbeitungszeit, der MA-Taktfrequenz und dem konfigurierten *BiSS*-Timeout) ermittelt.

Als zusätzliche Prüfung kann die Steuerung die erwarteten Slave-Kennungen und Seriennummern jedes Slaves speichern und bei jedem Bus-Aufbau mit den aktuell ermittelten Positionen vergleichen.

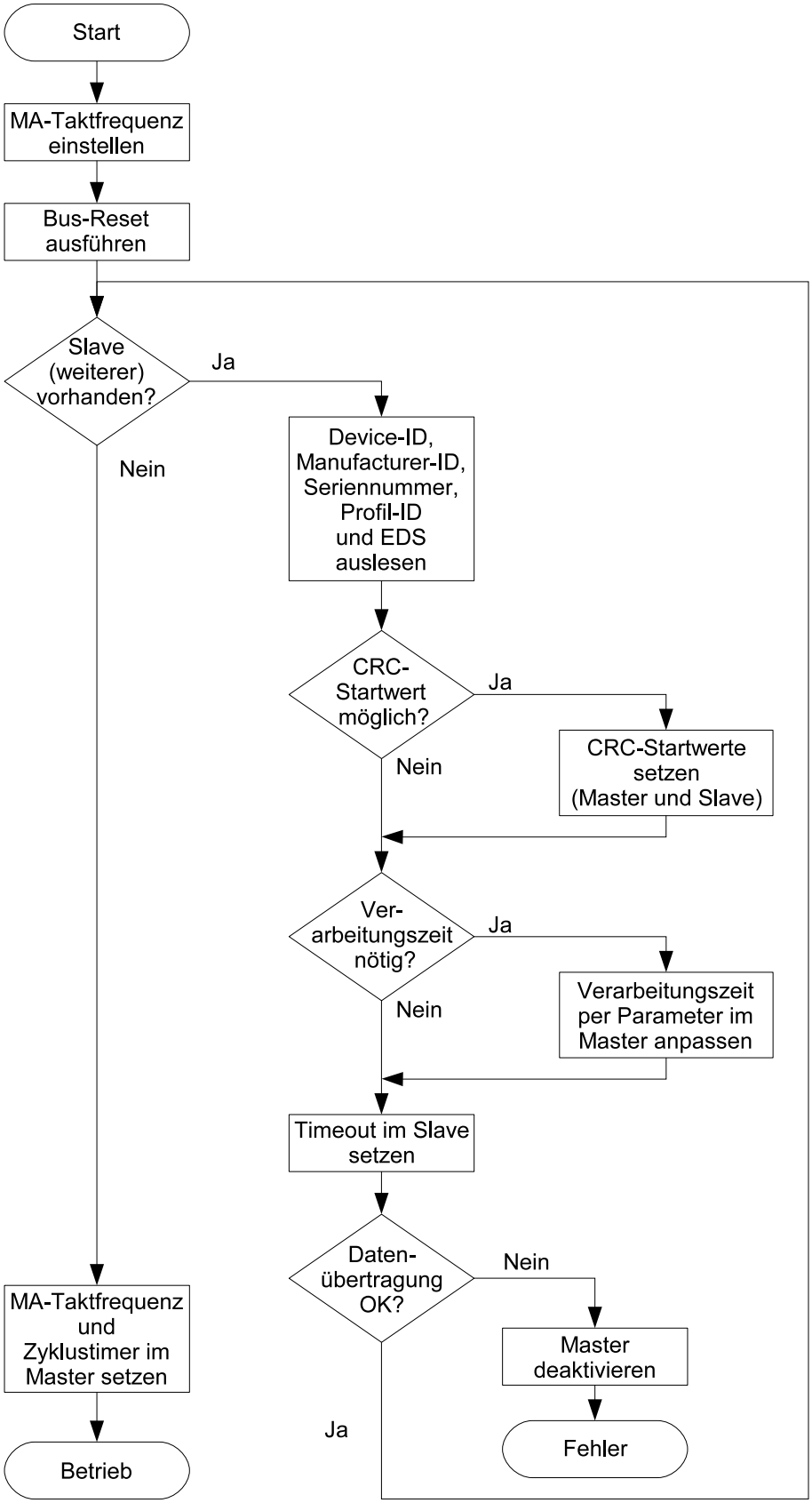


Bild 20: Ablauf eines Busaufbaus

SYSTEMENTWURF

Beim Entwurf eines *BiSS* C-Mode Systems ist zu beachten, dass die Dauer des *BiSS*-Frames variabel ist. Die Zykluszeit sollte so groß gewählt werden, dass der Beginn des *BiSS*-Frames auch bei maximalem *BiSS*-Timeout jitterfrei ist. Slaves mit Verarbeitungszeit per Anforderung erzeugen einen Jitter in dem Zeitpunkt der Übertragung.

Der *BiSS* C-Mode Master
Sind die Anforderungen eng definiert, können anwendungsspezifische Custom-*BiSS* C-Mode Master mit reduzierten Fähigkeiten verwendet werden. Dagegen müssen Master, an denen beliebige Slaves betrieben werden können, die folgenden Mindestanforderungen erfüllen. Sie werden als Full-*BiSS* C-Mode Master bezeichnet.

- Mindestens 16 unterschiedliche Datenkanäle programmierbar.
- Mindestens 16*8 = 128 Byte Speicher für Datenwerte (AD/SD)
- Konfigurierbare CRC-Prüfung mit 0 bis 8 Bit und variablem Polynom und Startwert je Datenkanal
- Senden des verkürzten *BiSS*-Frames
- Betrieb in der Punkt-zu-Punkt- und Bus-Konfiguration
- Programmierbare MA-Taktfrequenz ab 80 kHz
- Ein programmierbarer Zyklus-Timer, der den *BiSS*-Frame in gleichen Zeitabständen startet.
- Status je *BiSS*-Frame: Frame vollständig, Fehlercode
- Status je Datenkanal: Datenkanal vollständig empfangen und Datenwert gültig (CRC).
- Status für Registerkommunikation: Register nicht vorhanden, Registerwert gültig (CRC), alle 9 IDL-Bits
- Sequentieller Registerzugriff über mindestens 16 Register

Die *BiSS* C-Mode Slaves
Der *BiSS*-Timeout sollte nach dem Einschalten im Slave auf 12,5 bis 40 µs stehen, um eine automatische Erkennung durch die Steuerung sicherzustellen. Bei dem Betrieb mit geringeren Bitraten kann ein *BiSS*-Timeout von kleiner als 12,5 µs dazu führen, dass kein gültiger *BiSS*-Frame übertragen werden kann. Dieser Fall ist durch den Master von einer Leitungsunterbrechung nicht unterscheidbar. Eine automatische Bitraten-Erkennung kann implementiert werden.
Slaves, die beim Einschalten einen geringeren *BiSS*-Timeout verwenden, werden als Custom *BiSS* C-Mode Slaves bezeichnet.

Bei der Definition von Datenkanälen sollte bei rechtsbündiger Ausrichtung des Messwerts zusätzliche Signal- oder Fehlerbits links angeordnet werden, bei linksbündiger Ausrichtung rechts, um die Ausrichtung des Messwertes an den Bytegrenzen zu erhalten.
Falls ein Device mehrere Datenkanäle besitzt, kann jeder Datenkanal einem logischen Slave zugeordnet werden. Dies hat den Vorteil, dass für jeden Datankanal eine Profilkennung zur Verfügung steht und jeder Datenkanal individuell aktiviert werden kann.

KENNDATEN

Nr.	Symbol	Benennung	Bedingung	Min.	Max.	Einh.
01	$1/T_{MA}$	Taktfrequenz	Signal MA	80	[1]	kHz
02	$1/T_{MAmin}$	minimale Taktfrequenz	Signal MA, unterstützte Taktfrequenz aller Teilnehmer	80	10000	kHz
03	t_{MAl}	Taktsignal MA, Pulsbreite Low	MA = “0“	[1]	12.5	µs
04	t_{MAh}	Taktsignal MA, Pulsbreite High	MA = “1“	[1]	12.5	µs
05	$t_{BiSS\text{-}Timeout}$	*BiSS*-Timeout		12.5 [2]	40	µs
06	$t_{BiSS\text{-}Timeout_s}$	verkürztes *BiSS*-Timeout für Slaves ohne automatische Anpassung des *BiSS*-Timeouts an T_{MA}	Nach Setzen der Bits 0 bis 2 an Adresse 124	0.5	3	µs
07	$t_{BiSS\text{-}Timeout_a}$	adaptives *BiSS*-Timeout für Slaves mit automatischer Anpassung des *BiSS*-Timeouts an T_{MA} Abtastfrequenz: $1/T_{CLK}$	$T_{CLK} \leq 1.5 \cdot T_{MA}$ $T_{CLK} \geq 1.5 \cdot T_{MA}$	$1.5 \cdot T_{MA}$ $1.0 \cdot T_{CLK}$	$1.5 \cdot T_{MA} + 3.0 \cdot T_{CLK}$ $1.5 \cdot T_{MA} + 3.0 \cdot T_{CLK}$	
08	$t_{LineDelay}$	Verzögerung MA → SL	Messbar im *BiSS*-Frame von der zweiten steigenden MA-Flanke zur ersten fallenden SL-Flanke	0	40	µs
09	$t_{LineJitter}$	Verzögerungsjitter MA → SL	Innerhalb eines *BiSS*-Frames	-25	25	$\%T_{MA}$
10	t_{busy_s}	Verarbeitungszeit für Single-Cycle-Daten		0	40	µs
11	t_{busy_r}	Verarbeitungszeit für Registerzugriffe		0	20	ms

[1] Die maximale Taktfrequenz ist vom Übertragungsmedium (siehe BiSS-Interface - Physical Layer) und von den einzelnen Teilnehmern abhängig. Die maximale Taktfrequenz der Slaves ist u.a. im EDS gespeichert.
[2] Die Min.-Spalte gilt nicht nach Programmierung des *BiSS*-Timeouts, automatischer Anpassung an T_{MA} oder für Custom *BiSS* C-Mode Slaves. Der verringerte *BiSS*-Timeout kann die *BiSS* Kommunikation mit niedriger Taktfrequenz verhindern; der Fehler ist nur als allgemeiner Kommunikationsfehler erkennbar.

Tab. 2: Tabelle Kenndaten

DIAGRAMME UND FARBEN

Diagramme und Farben
Erfolgt die Darstellung einer Übertragung im Zeitbereich, verdeutlichen die abgeschrägten Kanten die endliche Steilheit und einen Bereich des Signalwechsels. Der *BiSS*-Frame wird im Zeitbereich übertragen und mit dem Signal MA getaktet.

Die Steuerungskommunikation wird jeweils mit einem Bit je Zyklus (=*BiSS*-Frame) übertragen. Die Control-Frames werden durch Zyklen getaktet, die geraden Kanten weisen auf die rein logische Natur der Bits hin.

Die folgenden Farben werden zur Verdeutlichung der Diagramme verwendet:

Farbe	Bedeutung
Orange	Beginn des *BiSS*-Frames, ermöglicht Laufzeit und Verarbeitungszeit.
Hellblau	Sensordaten (Single-Cycle-Sensor-Daten (SCDS))
Pfirsich	Aktordaten (Single-Cycle-Aktor-Daten (SCDA))
Gelb	Steuerungskommunikation (Kommandos und Registerzugriffe)

Tab. 3: Tabelle Farben

BEGRIFFE UND ABKÜRZUNGEN

Begriff	Bedeutung
Bus-Aufbau	Vorgang der Erkennung aller angeschlossen Slaves einschließlich der Parametrierung und Fehlererkennung.
Master	Schnittstellenbaustein zwischen dem *BiSS* C-Mode Bus und einer Steuerung.
Gerät (Device)	Physikalsche Einheit. Enthält einen oder mehrere Slaves.
Slave	Logische Einheit innerhalb eines Device. Belegt eine ID und enthält keinen, einen oder mehrere Datenkanäle.
BiSS-Frame	Wird zyklisch ausgeführt und überträgt Single-Cycle-Daten komplett und jeweils ein Bit Kommando- bzw. Registerdaten.
BiSS-Timeout	Von den Slaves vorgegebene Zeit. Trennt aufeinanderfolgende *BiSS*-Frames.
“Erster Slave“	Der Slave, dessen Daten zuerst mittels des *BiSS*-Frames übertragen werden.
“Letzter Slave“	Der Slave, dessen Daten zuletzt mittels des *BiSS*-Frames übertragen werden.
Datenbereich	Die Bits im *BiSS*-Frame, welche die Datenkanäle übertragen.
Datenkanal	Ein logischer Kanal, er enthält die Daten- und Sicherungsbits eines Datenwertes und wird als solcher parametriert.
Datenwert	Ein gesichert übertragener Wert, er kann weitere Bitgruppen (z.B. Messwert und Fehlerbits) enthalten, wird aber vom *BiSS*-Master als eine Zahl behandelt.
Registerbank	Ein umschaltbarer Bereich von 64 Registern, in den Register oder Speicher des Slaves eingeblendet werden können.
Steuerung	Die Steuerung ist die dem *BiSS*-Master übergeordnete Logik. Üblicherweise wird darunter der den *BiSS*-Master steuernde Prozessor und die darauf ablaufende Software verstanden.
Big Endian	Byte-Reihenfolge, bei der das höchstwertige Byte an der niederwertigsten Adresse gespeichert wird (In BiSS C-Mode verwendet).

Tab. 4: Tabelle Begriffe

Kürzel	Bedeutung	Beschreibung
SCD	Single Cycle Data	Ein in einem *BiSS*-Frame komplett übertragener Wert.
CDM	Control Data Master	Datenbit des Masters in der Steuerungskommunikation. Wird im *BiSS*-Timeout gesendet.
CDS	Control Data Slave	Datenbit des Slaves in der Steuerungskommunikation. Wird im *BiSS*-Frame nach dem Startbit gesendet.
DCH	Datenkanal	Logische Einheit zur gleichzeitigen gesicherten Übertragung mehrerer unabhängiger Daten.
SLI	Slave In	Dateneingang des *BiSS* C-Mode Slaves.
SLO	Slave Out	Datenausgang des *BiSS* C-Mode Slaves.
MO	Master Out	Datenausgang des *BiSS* C-Mode Masters.
MA	Master Clock	Taktausgang des *BiSS* C-Mode Masters.
SL	Slave return	Dateneingang des *BiSS* C-Mode Masters.
EDS	Electronic Data Sheet	Elektronisches Datenblatt.
XML	eXtensible Markup Language	Eine Sprache zur Beschreibung von Dokumenten.
CRC	Cyclic Redundancy Check	Verfahren zur Übertragungssicherung.
MSB	Most Significant Bit	Höchstwertiges Bit.
LSB	Least Significant Bit	Niederwertigstes Bit.

Tab. 5: Tabelle Abkürzungen