BiSS Interface

PROTOCOL DESCRIPTION (BiSS C)

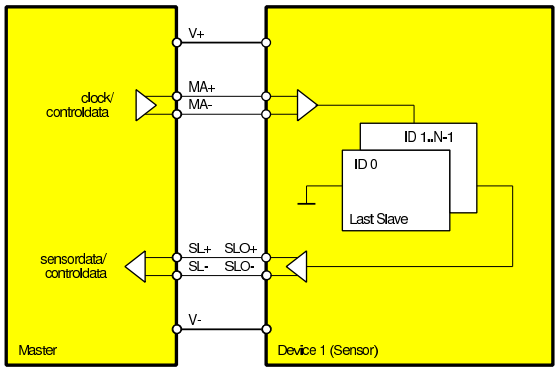
FEATURES

- Sensor/actuator interface
- Isochronous, real-time-capable data transmission
- Fast, serial, safe
- Permanently bidirectional
- Point-to-point or multi slave networks
- Compact and cost-effective
- Open standard
- Directly usable IP modules available

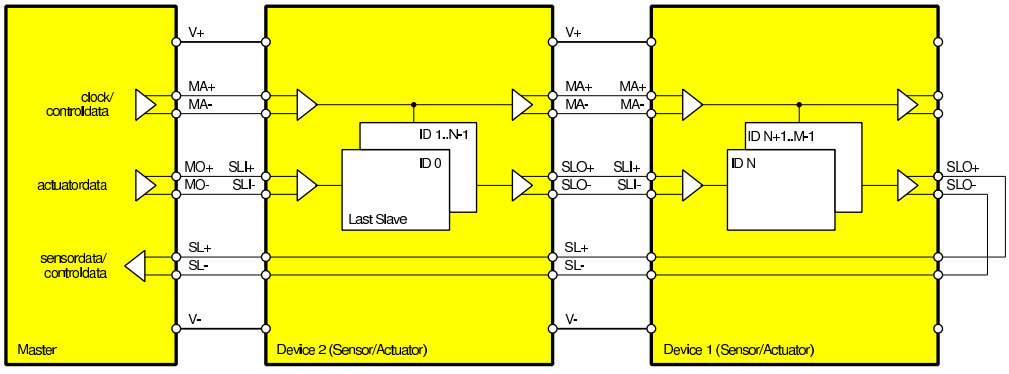
APPLICATIONS

- Drive controllers
- Rotary encoder
- Linear encoder
- Smart sensors
- Safe actuators
- Communication watchdogs

BLOCK DIAGRAM



Sensor bus resp. point-to-point network



Sensor/actuator bus

BRIEF DESCRIPTION

This specification describes a serial interface protocol for the isochronous, fast and safe exporting of sensor data, real-time writing of actuator data and simultaneous accessing of the register of the slaves.

In the point-to-point configuration, only one device with one or more sensors is operated on the master. The MO line is eliminated, and the SL line is routed back directly from the slave. The *BiSS* interface is hardware-compatible to the SSI interface in the point-to-point configuration and only requires two unidirectional lines.

In the bus configuration all device are connected in a chain. Each slave therefore has two connectors, i.e. *BiSS*-IN and *BiSS*-OUT, with assignments provided for differential signals. The MA line supplies the clock pulse of the master simultaneously to all slaves, and the SLI and SLO lines connect the master and all slaves to a chain (MO $\rightarrow$ SLI_N, SLO_N $\rightarrow$ SLI_{N-1}, SLO_1 $\rightarrow$ SL), by connecting the input of a slave (SLI) to the output (SLO) of the previous one.

Any desired number of slaves can be connected to a *BiSS* C interface. These can then function both as actuators and as sensors, and can transmit data simultaneously via several logical channels. Each slave has a memory which, in addition to its configuration, can also contain its identification (manufacturer and device ID) and, if necessary, a profile ID and its electronic data sheet (EDS).

For data transmission the *BiSS* C protocol uses two types of data with various throughputs:

- **Single Cycle Data (SCD)** are the primary data and are newly generated and completely transmitted in each cycle. A distinction is made between sensor data, which are transmitted from the slave to the master, and actuator data for the opposite direction. They are used for conveying very rapidly changing values, such as the position, acceleration or for the position control of drives.
- **Control Data (CD)** are transmitted with one bit per cycle (in the fixed start sequence and the final clock), however use only one bit per direction for all slaves. They permit the reading and writing of the slave register and the sending of commands to selected or all slaves.

The parameters of the individual data channels, including the number of bits and CRC format, are specified by the slave and are stored in its electronic data sheet. This is either a file referenced via the slave ID and stored in the controller (in the XML format) or it is stored directly in the memory of the slave. As an alternative, the transmission parameters can be obtained via application-specific profiles which are also referenced via an ID stored in the slave. The controller reads the profile ID or the identification and the EDS of the slaves by accessing the register and programs the master in accordance with the parameters of the slave.

In addition to using a "Full *BiSS* Master", which permits the connection of any desired slaves, "Custom *BiSS* Masters" can operate with a limited data channel parameter, and therefore only work together with one or a few slaves. This option enables implementations of *BiSS* masters with few resources in small FPGAs and with very little RAM.

The term "First Slave" refers to the slave with the data which are transmitted to the master first. Its output SLO is directly connected to the return line SL. The input SLI of the "Last Slave is connected to the line MO of the master or connected to "0" in the point-to-point configuration.

OPERATING DESCRIPTION

The *BiSS* C protocol enables the simultaneous transmission of sensor data (SD) from all slaves to the master, actuator data (AD) from the master to the slaves and control data (CD) to individual or several slaves.

The BiSS frame

The isochronous transmission of the BiSS frames is typically used for cyclical scanning systems. Here each cycle begins with the transmission of a *BiSS* frame, then the interface remains in the idle state up to the beginning of the next cycle (MA = MO = SLO_X = "1"). The cycle duration is therefore at least equal to the duration of a *BiSS* frame, and may be as long as desired.

The *BiSS* frame (transmission frame) is started by the master with the clock MA, clocked and ended. Here the first rising edge at MA is used for the synchronization of all slaves. It enables the isochronous scanning of sensor data and the isochronous output of actuator data. With the 2nd rising edge from MA, all slaves set their SLO line to "0" and generate their "Ack" (Acknowledge) signal with it; it remains active (SLO = "0") until the start bit arrives at the input SLI of the respective slave. The start bit is then passed on synchronously with the clock MA from each slave delayed by one clock pulse, while the CDS bit is either passed on by the slave or is set according to the rules of the control frame.

Beginning with the 2nd bit after the start bit and up to the stop bit of the *BiSS* frame, the data range follows, which transmits the sensor data from the slaves to the master and the actuator data from the master to the slaves.

The *BiSS* frame ends with the *BiSS* timeout. In this time no further clock pulses are sent to the MA by the master. The inverse state of the MA line during the *BiSS* timeout is the state of the CDM (Control Data Master) bit. At the end of the data transmission, the master sets its output MO to the idle state "1". The slaves then pass on this "1" received at SLI to their output SLO as soon as they have detected the expiration of the timeout themselves. This ensures that the BiSS timeout on the line SL is only signaled to the master when all connected slaves have detected the timeout. When the *BiSS* timeout expires, all slaves return to the idle state; all lines are set to the high signal level ("1") in the process.

In the point-to-point configuration (see figure 1), the start bit is generated by the last slave; it detects the point-to-point configuration from the fact that its input SLI is already "0" at the start of a frame. In the same way the slave signals the expiration of the *BiSS* timeout without waiting for a predecessor. The output of the only device (SLO) is directly connected to the SL input of the master in this case.

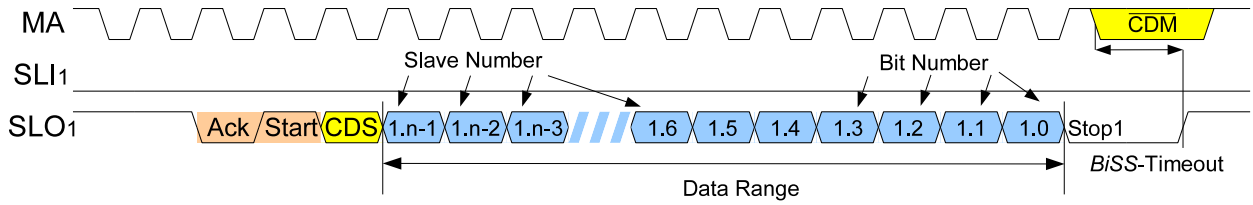


Figure 1: *BiSS* frame (point-to-point configuration)

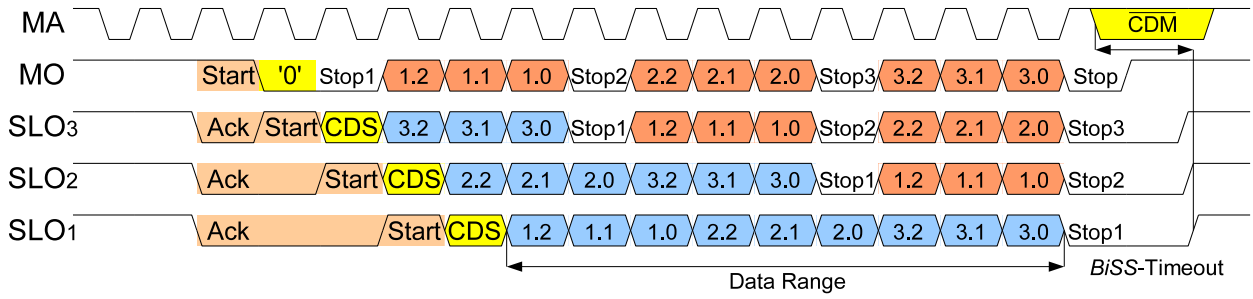


Figure 2: *BiSS* frame with 3 slaves (bus configuration)

In each *BiSS* frame, one bit of control data (CD) is transmitted per direction for the command or for register communication. The control data bit of the master

(CDM, Control Data Master) is sent to all slaves via the line MA as an inverse signal level of the *BiSS* timeout. The addressed slave responds with the CDS (Control Data Slave) bit, which is always transmitted in the first bit after the start bit. The master always sends the CDS bit with zero (MO: CDS = "0"). The control data bits of several consecutive BiSS frames are combined by the master and by the slaves to form a control frame (see Control communication). It permits the reading and writing of the slave register and the sending of commands to selected or all slaves.

Processing time per request
If a slave requires additional processing time before outputting its sensor data, e.g. for A/D conversion or for memory access, it can request this by delaying the start bit. The master detects the delayed start bit and generates the additionally required MA clock pulses.

If a device in the point-to-point configuration consists of several slaves, then all except the last slave must temporarily save the data of their predecessor received at SLI and then send this data to SLO following their own data. The slave with the longest processing time specifies the entire processing time; it is advisable to position it as the last slave.

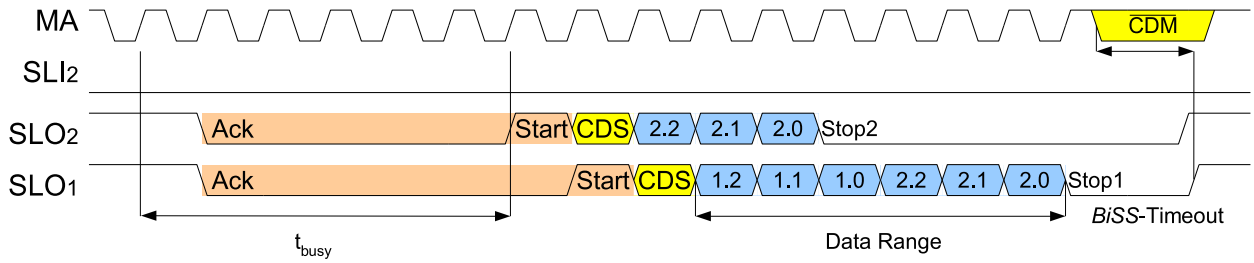


Figure 3: Requests for processing time (point-to-point configuration)

Processing time per parameter
In the bus configuration, the master delays the output of the start bit to MO. For this purpose, the master is configured to the maximum delay time of all connected slaves during the bus establishment. If the processing time required by the slave is variable, the maximum required processing time must be set. Specified times are converted by the controller into clock pulses with the currently set bus clock pulse, rounded up and configured in the master.

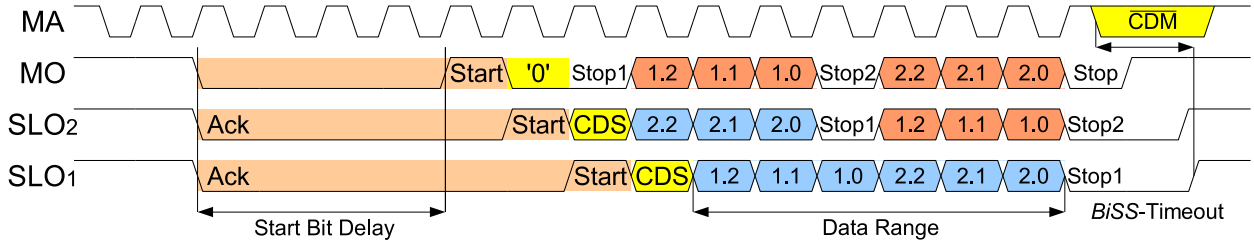


Figure 4: Parameterized processing time (bus configuration)

Line delay compensation

At high data rates, the line MA must have the same line topology and be provided with the same line drivers as the chain SLI-SLO. As a result, MA and SLI are assigned the same additional-time-dependent delay and remain synchronized. The total signal delay of the chain from the clock pulse MA to the output of the first slave (signal SL) can be measured out by the master and compensated with a corresponding shifting of the scanning of the slave signal. In addition, the data output to MO is delayed if the line delay time is greater than one period. To determine the line delay, the delay from the second rising MA edges to the falling edge of the Ack bit of the slave response (SL: "Ack") is used; it is ideally zero.

The line delay compensation enables accelerated communication with high data rates of typically 10 Mbits/s. It is carried out again with each BiSS frame, and therefore also takes aging and temperature dependent drift effects into account.

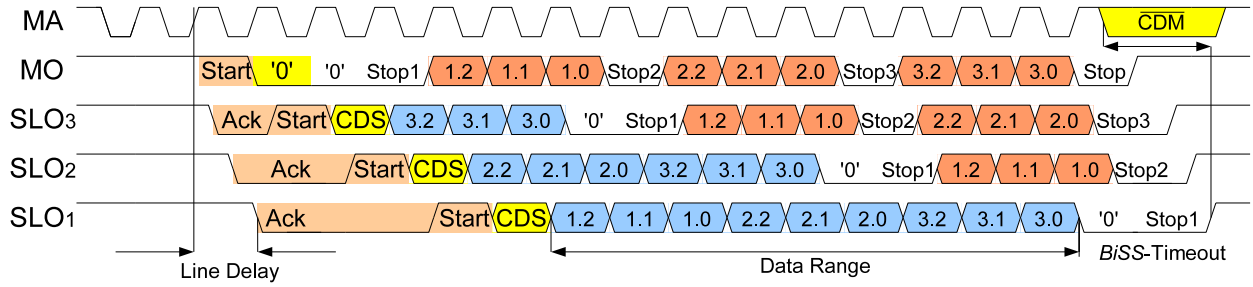


Figure 5: Line delay compensation by the master

Bus reset

After switch-on or an error, the master must maintain a break of 40 µs prior to the data transmission. This ensures that the *BiSS* timeout has expired and all slaves are ready for the data transmission. In the point-to-point configuration the last slave is not defined before the first MA pulse, which also causes the SL line to remain set to zero. The master must either generate a pulse to MA or start the first cycle without taking SL into account. Slaves that require a longer configuration phase reject all register accesses within this phase.

SENSOR AND ACTUATOR DATA COMMUNICATION

The data area
The data area is used to transmit the sensor data from the slaves to the master and the actuator data from the master to the slaves. The entire data area is divided into logical data channels. The position and length of the individual data channels is described for each slave in its parameters. A slave can have no, one or several data channels for sensor and/or actuator data.

The master must be programmed with the parameters of the individual data channels in their sequence and the sequence of the connected slaves to be able to correctly assign their bits. Valid data can only be transmitted with a correct configuration of the data channels in the master. The check bits contained for each data channel are used to detect transmission errors and bit offsets.

The data of the first slaves reach the master directly after the CDS bit. The further data channels follow without separation by start or stop bits. The length of the data area is the sum of the length of all data channels. The slave numbering is carried out in the sequence of the data transmission, and is therefore counted in the reverse sequence of the signal direction SLI → SLO.

The slaves can only correctly signal the *BiSS* timeout in the *BiSS* frame if all SLO lines have the signal level "0" at the start of the *BiSS* timeout. The master sends a leading zero before each packet (data channels of a slave), which is active as a stop bit at the slave output at the end of the cycle. If an error occurs, the *BiSS* frame can be canceled with a timeout of 40 µs at any time.

The received single cycle actuator data is sent back from the actuator to the master with the following *BiSS* frame. For sensor data channel the *BiSS* C master sends the signal level "0" to the MO output.

The sensor data are acquired with the first rising edge of the *BiSS* frame, while the actuator data are adopted in the slave with the expiration of the *BiSS* timeout from the shift register. The master control the output of the actuator data so that these are correctly contained in the shift registers of the slaves with the last MA clock pulse. For the isochronous operating mode, it is advisable to also output the actuator data with the first rising clock pulse edge of the following cycle at the outputs.

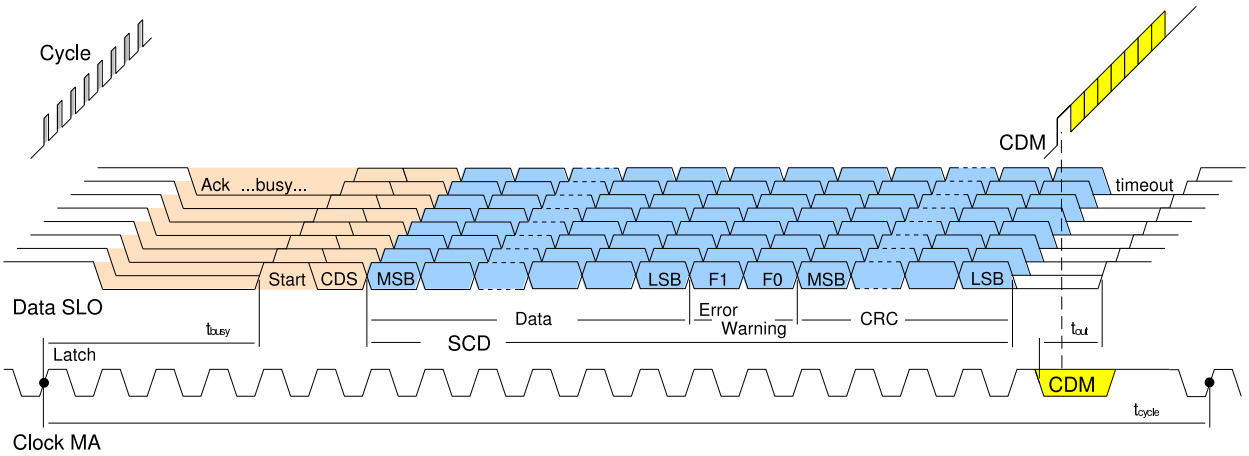


Figure 6: The data area and the data channels

The data channels
A data channel is a logical unit used for secured data transmission and describes the related parameters and data contents. Each channel is a sensor data channel (slave => master) or an actuator data channel (master => slave) and contains fast (SCD) data. A data channel occupies the parameterized length in the data area of the *BiSS* frame and in the memory of the master (number of data and check bits). In the electronic data sheet, each data channel has a section which contains the parameters necessary for secured data transmission and the description of the data contents. The controller configures a data channel in the master in accordance with these parameters.

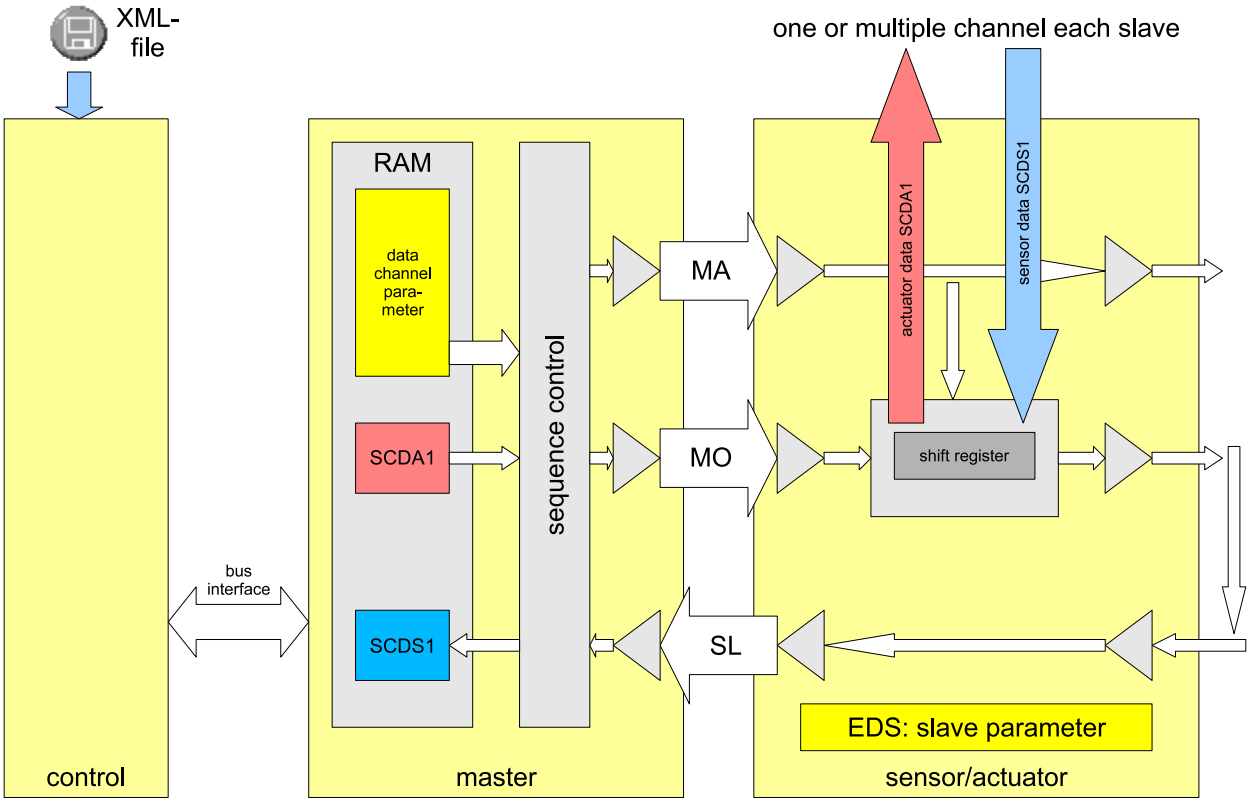


Figure 7: Configuration and transmission of sensor and actuator data

The data channel parameters
The following parameters are defined for data channels:

- Transmission direction and type
  1. SCDS (Single-Cycle Sensor Data)
  2. SCDA (Single-Cycle Actuator Data)
- Number of bits (0...64)
- Stopbit (before Actuator data) do not send/send
- Processing time (0 µs... t_{busy_s} or 0 µs... t_{busy_m})
- Data alignment (left or right-justified)
- CRC polynomial (for 0...16 CRC bits)
- CRC start value (for 0...16 CRC bits)

Single-Cycle Data (SCD)
A data channel with single-cycle data is used for fast and cyclical sensor or actuator data, which are transmitted completely in one cycle. SCDs require no addressing, have a parameterizable length of 1 to 64 data bits and a CRC checking of 0 to 16 bits.

Figure 8: An SCD data channel

Processing time for single-cycle data
The processing time for SCD begins with the first rising MA edge simultaneously in all slaves. The frame length is therefore only determined by the longest processing time of all slaves. Only this time is visible to the master. In the parameters the processing time is either specified as a time unit or in MA clock pulses; a dependency on the frame length is permissible. The maximum processing time for SCD is t_{busy_s} (see Characteristic).

The data value
All data values are transmitted with the most significant bit first ("MSB first"). A data value itself can consist of several bit groups, e.g. a measured value and several error flags. The composition and alignment of the data value are defined in the electronic data sheet of the slave. Figure 11 shows both versions with a data value of 13 bits, which is stored in a 16-bit wide word in the master.

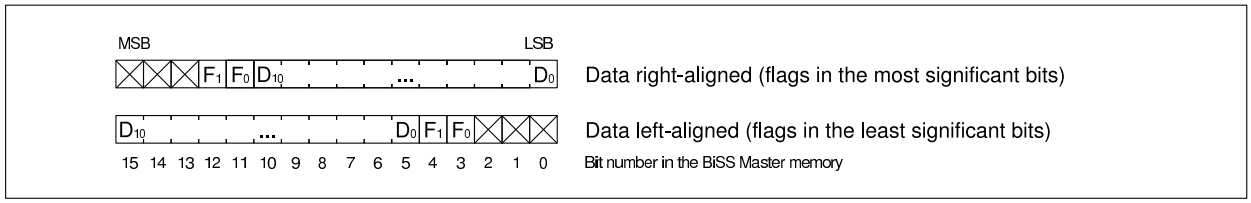


Figure 9: Alignment of the data value

Invalid values
It is recommended that the data value zero ("0") be reserved as invalid for the detection of an incorrectly configured transmission direction of a data channel for single-cycle data. The master also transmits the data value zero if no new data are available for an actuator data channel at the start of the *BiSS* frame. Usually, the value zero ("0") is avoided in the valid data by using at least one bit of the data value as a "0"-active error bit.

CRC checking
Each data channel can use a transmission checking with CRC in addition to its data value. The properties of the CRC checking are specified in the parameters of the data channel. The CRC polynomial also indicates the transmitted CRC bits; 0 to 16 bits are possible. The CRC check bits are always transmitted inversely first with the most significant bit.

The starting value for the CRC calculation is generally zero, however can be provided with a configurable starting value if the master and the slave support this function. The configuration of the CRC starting value permits the clear assignment of a data value to a slave, as the CRC check fails with a faulty configuration of the master or an exchanged sequence.

The controller assigns a CRC start value for each data channel and writes these to the slave by accessing the register; the corresponding register address is specified in the electronic data sheet.

The device manufacturer can also assigns a CRC start value for each data channel by configuring dedicated register in a protected configuration address range.

If other checking methods are used to protect a data channel, or if the maximum number of bits is not sufficient for the CRC check, then the CRC check or CRC generation via parameterization is deactivated (CRC polynomial = 0). The check bits are then transmitted within the (maximum of 64 bits per data channel) normal data bits and stored in the memory of the master. The check bits can now be checked with software.

CONTROL COMMUNICATION

The control frame

The control frame enables the protected and confirmed reading and writing of the register of a slave and the protected and confirmed sending of commands to selected or all slaves. The control frame results from a count of *BiSS* frames generating and transmitting sensor or actuator data.

The register access or the command is always carried out at the end of the cycle of the last CDM bit, i.e. with the expiration of the BiSS timeout in the slave. The control frame can be canceled at any time with the transmission of 14 "0" bits. The start bit of a control frame must be preceded by at least 14 cycles with CDM = "0".

Slave addressing with ID assignment

In contrast to actuator and sensor data communication, the control communication requires clear addressing. The address of the slave ("ID") is dynamically assigned according to the sequence in the chain. The ID assignment is carried out automatically during each control frame for the first 8 IDs by setting the ID lock bits (IDL).

The ID assignment begins in each control frame after the start bit of the masters (CDM = "1"). For this purpose, each slave waits for the first free CDS bit (SLI = "0") and sets it (SLO = "1").
Due to the sequence in the chain recurrent network, the CDS bits from the last slave are assigned first, which results in the IDs being assigned in the reverse sequence of the slave numbers.

The example in figure 10 shows the ID assignment of four devices, which contain one slave (device 3 + 4) or four slaves (Device 1+2) each. As the available eight IDs are not sufficient for ten slaves, the front slaves signal with ID8 = "'1"' that they have not received an ID.

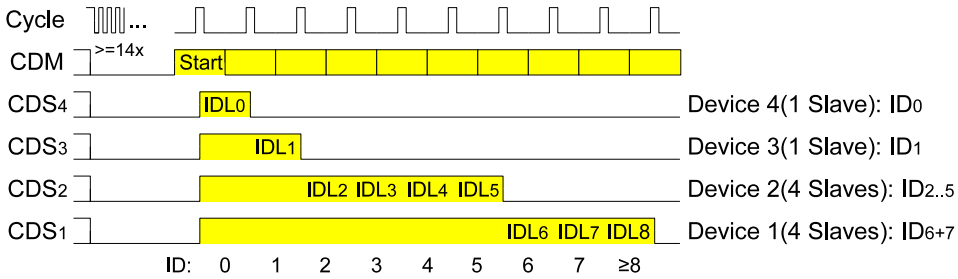


Figure 10: Example of an ID occupation

CRC checking

The control communication also uses a checksum for transmission checking. The CRC polynomial used is:
$X^4 + X^1 + X^0$
With it 4 CRC bits are available, which are transmitted inverted. Calculation is carried out with the starting value zero via the addressing sequence or the data bits beginning with the most significant bit and always excluding the start bit.

Note:
Before each control frame at least 14 bits with CDM = "0" must be transmitted.

The commands
Commands can be sent simultaneously to 8 slaves in any desired combination (addressed) or to all slaves on the bus together (broadcast).

A command frame begins with a start bit and the control select bit CTS, which is "0" (CTS = "0": command). The master selects the addressed slave(s) with the IDS bits that follow, whereby the bit-by-bit coding permits any desired combination of the first 8 IDs (e.g. $IDS_{0..7}$ = "1000 0100" is equal to ID = "0" und "5"). Then the master sends one of the four commands (CMD = "00" / "01" / "10" / "11") and completes the addressing sequence with a 4-Bit CRC. The ID assignment of the slaves is carried out with the ID lock bits at the same time as the addressing sequence. The addressed slaves confirm the correct receipt of the command on the following start bit (17th bit) by each addressed slave setting the corresponding IDA bit (ID acknowledge). The master or the controller can compare the IDA bits to the IDS bits bit by bit (IDS = IDA) and carry out the command depending on the result by sending an EX bit (Execute, CDM="1") or cancel the execution by sending 14 "0" bits.

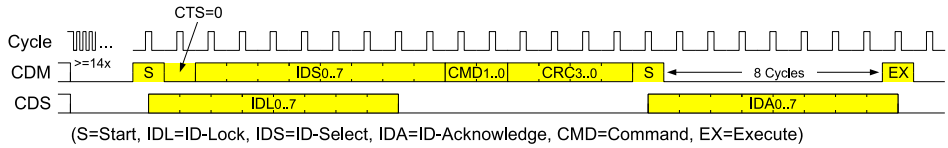


Figure 11: The command frame (addressed)

In addition, it is also possible to send commands to all slaves (broadcast) by not addressing any slave, i.e. none of the IDS bits is set. Broadcast commands are also carried out by the slaves even if they do not currently occupy any ID. As a special feature of the broadcast command, the *BiSS* frame can be shorted to two clock pulses and a timeout in the process.

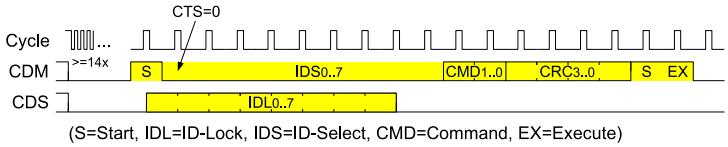


Figure 12: Command frame (broadcast)

The reduced BiSS frame
During initialization it is not necessary to transmit the complete *BiSS* frame, especially if its length is unknown due to the slave parameters not yet read. It is possible to cancel the *BiSS* frame after the acknowledge of the slaves and only to send the CDM bit in the *BiSS* C timeout. As no CDS bit is transmitted here, and therefore neither an ID occupation nor a response of the slaves possible, the shortened frame can only be used for broadcast commands.

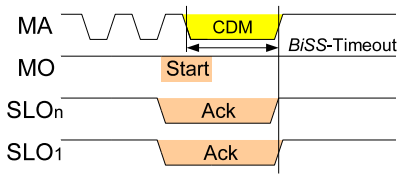


Figure 13: Reduced *BiSS* frame

The two command bits and the two addressing options therefore result in eight commands. Some of these commands are specified for *BiSS* C and must be interpreted by each *BiSS* C slave.

Reserved
(CMD = "00", broadcast):
The data channels of all slaves (single-cycle sensor or actuator data) are deactivated.

Reserved
(CMD = "01", broadcast):
The register and command communication of all slaves are activated. In the control frame that follows the IDs are assigned again in the physical sequence of the slaves.

Deactivate bus coupler or reserved
(CMD = "10", broadcast):
All bus couplers are switched to "bypass". This command is only relevant for bus couplers. All other slaves ignore it.

Reserved
(CMD = "11", broadcast):
The broadcast command "11" is reserved for future expansions.

Reserved
(CMD = "00", addressed):
The data channels of the addressed slaves (single-cycle sensor or actuator data) are activated.

Reserved
(CMD = "01", addressed):
The register and command communication of the addressed slaves are deactivated.

Activate bus coupler or free
(CMD = "10", addressed):
An addressed bus coupler switches over from "bypass" to "line operation". This command is freely definable for other slaves.

Free
(CMD = "11", addressed):
This command is freely definable for each slave.

The commands are required for bus establishment. With the broadcast commands "00" and "01" all data channels are deactivated and the register communication of all slaves is activated. These commands are transmitted with reduced BiSS frames. As the data area of the BiSS frame then has the length zero, the data channel parameters of the slave can now be read from the respective EDS with normal but short BiSS frames. If the command "00" is sent to individual slaves (with addressing, IDS > 0), then the data channels (sensor and actuator data) of the addressed slaves are inserted in the data area again. In addition, data channels and unknown or defective slaves can be logically removed from the bus in this way without negatively affecting other slaves.

All bus couplers are switched to "bypass" with the broadcast command "10". Any faults, such as short circuits or open circuits, can therefore be detected and isolated beginning with the first slave. The slaves of a bus are then switched over consecutively and individually from "bypass" to "line operation" and checked for a response of the following instance/connection.

The free commands can be used to carry out any desired actions (e.g. zeroing for rotary encoder) simultaneously for selected slaves. If more than two free commands are required, it is advisable to first select the command via a respective register access in the corresponding slaves and then to send the command to the slaves involved for simultaneous execution.

More than 8 IDs
If more than 8 possible IDs are required, the slaves set the 9th bit (ID Lock, IDL8). It signals the controller that there are additional slaves. The additional slaves can only be addressed after the control communication of one or several slaves, which occupy the first 8 IDs, have been deactivated with the command "01". The IDs are redistributed with the following control frame, whereby the slaves with deactivated register communication do not occupy any IDs and the corresponding number of slaves moves up into the first 8 IDs. Slaves with deactivated control communication can only be addressed via broadcast commands. The dynamic occupation of the IDs enables operation with any desired number of slaves.

Note: The master can only detect how many IDs are occupied using the ID lock bits (IDL0. . . 8), however not the assignment to the slaves.

The register communication
The read and write accesses to the registers of a slave are carried out with a control frame having a set CTS control select bit (CTS = "1": register access).

The register frame begins with the addressing sequence. Here the master sends the slave ID (3 bits), followed by the register address (7 bits) and a 4-bit CRC. Due to the binary coding, 8 slaves with 128 registers (= 128 bytes) can therefore be addressed. The assignment of the IDs by the slaves is carried out simultaneously with the addressing sequence (see ID assignment).

The two next CDM bits, i.e. the R and W bit, determine whether a read access (RW = "10") or a write access (RW = "01") is concerned. Both bits must be inverse to teach other and are returned to the master by the addressed slave for confirmation. They are not used when calculating the CRC.

The write access
With the write access, the two read/write bits have the value RW = "01". This is followed by a start bit, 8 data bits, a 4-bit CRC and a stop bit. The 8 data bits are protected during writing with a 4-bit CRC, and the transmitted register data are returned.

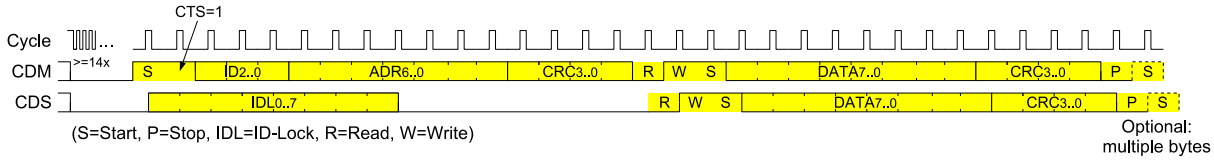


Figure 14: Register write access

The read access
During the read access the two read/write bits have the value RW = "10". This is followed by a start bit, 12 "0" bits and a stop bit. The register data are protected during reading with a 4-bit CRC.

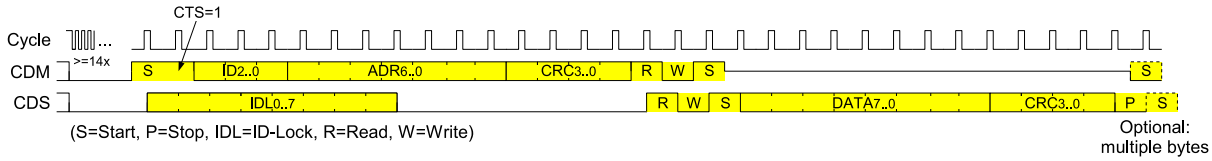


Figure 15: Register read access

Sequential register access
It is possible to read or write several consecutive registers in one access. For this purpose, the master sends another start bit (CDM = "1") directly after the stop bit of the first data value. During a write access the data byte, the 4 CRC check bit and the stop bit follow. During the read access only 13 "0" bits - including a stop bit - are sent. The slave internally increases the register address by 1 (auto increment) with each write or read access. A maximum of 64 registers can be read or written consecutive with an access. Sequential accesses beyond the register address 63 (0x3F) or 127 (0x7F) are not permitted. The sequential access ends when no further start bit follows at CDM.

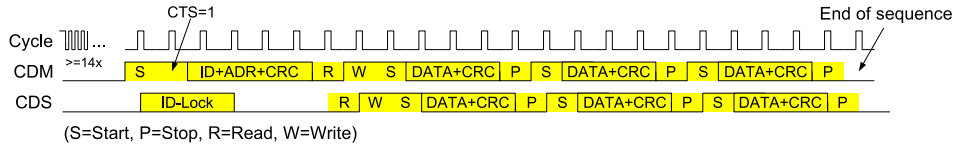


Figure 16: Writing several registers

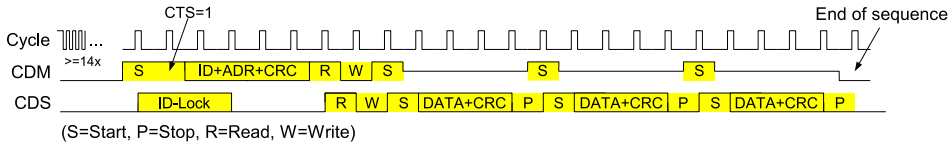


Figure 17: Reading several register

Non-implemented registers
The registers of a *BiSS* C slave can be "forbidden" or "not implemented". In this case, the slave rejects the access to the register by inverting the W-bit returned via CDS. During a write access, this leads to RW = "00", and during a read access to RW = "11".

If several registers are written or read consecutively and the register that follows is not implemented or cannot be addressed in the auto increment, the stop bit is inverted via CDS, i.e. is returned as "1" bit. The access to a non-implemented register ends the sequential access.

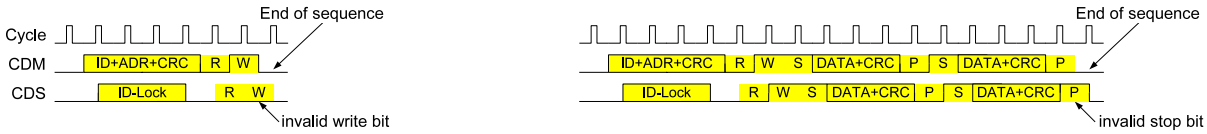


Figure 18: Accesses to non-implemented registers (here Write)
a) First register b) Additional register of a write sequence

Processing time for register access
If the slave requires additional processing time when reading or writing registers, it can request this for each byte individually by delaying its start bit. The master repeats the start bit during this time. If the start bit is not transmitted within the t_{busy_r}, the register access is canceled as invalid by the master canceling the repeating of the start bit. A register access usually requires processing time when an external memory is addressed. If processing time is required after the last transmitted register, e.g. for saving the value, this cannot be signaled via *BiSS*. However, the slave can request the processing time still required at the start of the following register access. The maximum processing time during register access is t_{busy_r} (see Characteristic).

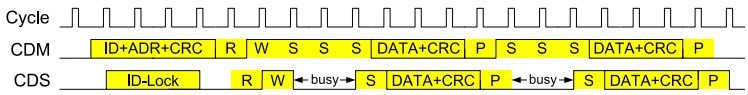


Figure 19: Write access with delay time

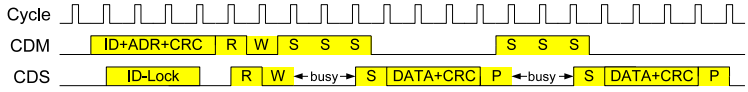


Figure 20: Read access with delay time

Note:
The CDS bit is transmitted in the *BiSS* frame before the CDM bit. During the register access the master must immediately evaluate the start bit received via CDS and reply in the same *BiSS* frame with the CDM bit. This is either "1" when processing time is requested or if required by the MSB to be sent, or "0". If evaluation is not possible at the end of the frame, the CDM bit can be sent independent on the CDS bit.

Register Protection
A register access protection can be defined in different levels. A register protection level RPL can be implemented to protect dedicated content. Different register protection level RPL can be implemented e.g.:

- No Access (e.g. OEM content)
- Read only (e.g. EDS, BP, serial number, BiSS ID, . . . content)
- Full access (e.g. USER DATA)

A register access violating the register protection level RPL can be based on read or write access. A register access violating the register protection level RPL will be refused like a non existing register: both R and W bits are identical refusing the access.

SLAVE REGISTER

The register assignment
The address area of a slave is limited by the register frame to 128 bytes. However, in addition to its RAM, the slave usually also only has a readable (ROM) or a rewritable (E^2PROM) memory. In addition to the activation configuration of the slave register, this can also contain the electronic data sheet (EDS) and other data if necessary. To enable access to the memory, 64 bytes are combined to a bank in each case. A bank is accessed via a bank switchover and the addresses 0 to 63. The registers from address 64 are directly accessible without a bank switchover. The register assignment is shown in Table 1.

Address (Decimal)	Address (Hexadecimal)	Name	Size	Memo
0 .. 63	0x00 .. 0x3F	Register bank	64 Bytes	
64	0x40	Bank selection	0..8 Bits (1 Byte)	1)
65	0x41	EDS-Bank	0..8 Bits (1 Byte)	1) 2) 4)
66 .. 67	0x42 .. 0x43	Profile ID	16 Bits (2 Bytes)	2) 3) 4)
68 .. 71	0x44 .. 0x47	Serial number	32 Bits (4 Byte)	2) 3) 4)
72 .. 119	0x48 .. 0x77	Slave register	48 Bytes	
120 .. 125	0x78 .. 0x7D	Device ID	48 Bits (6 Bytes)	2) 3) 4)
126 .. 127	0x7E .. 0x7F	Manufacturer ID	16 Bits (2 Bytes)	2) 3) 4)

1) If no bank switchover is used, the register should not be implemented.
2) Register is protected against accidental writing.
3) The value is saved as a Big Endian, i.e. with the highest-value byte at the lowest-value address.
4) Unused register contents must therefore be filled with "0".

Table 1: Table of register assignment

Register bank
Depending on the slave, each of the maximum of 256 banks can contain an additional 64 registers or 64 byte of memory. As a result, a memory area with a maximum of 16 Kbyte (256 * 64 = 16,384 byte) can be addressed.

Bank selection
The bank selection register at address 64 selects a register bank consisting of up to 256 banks and displays these at the register addresses 0 to 63.

EDS bank
As the size of the memory required for the activation configuration is slave-specific, the slave provides the bank number of the start of the freely available memory, which will be used for the electronic data sheet (EDS), in its register EDS bank at address 65. The EDS is described in: BiSS Interface - Electronic Data Sheet Definition.

Profile ID
To simplify the interchangeability and compatibility of *BiSS* C slaves, profiles are defined for frequently required device types. A profile definition includes all data channel parameters of a slave. It also contains additional information like the name or units of measure of data channels and registers. In the electronic data sheet there is a section in which the parameters defined in the profile are stored. Each profile has a unique, 16-bit ID based on which the controller can assign the parameters. If a slave does not use a predefined profile, then the registers 66 and 67 are not implemented or contain the value zero ("0"). Profile IDs are only assigned by iC-Haus, and new profiles can be applied for at support@biss-interface.com.

Slave register
The contents of these 48 bytes are slave-specific. They should be used for registers which must be read or written directly without bank switchover (e.g. status registers).

Device and manufacturer ID
Each bus device (slave) provides the system with a unique ID which consists of the manufacturer and the device ID. It can be read out under the register addresses 120 to 127 (6 bytes for the component, 2 bytes for the manufacturer) and permits the assignment of an EDS stored in the controller to the slave and the checking of the slave type by the controller.

The manufacturer ID is an ID of the slave manufacturer required especially for *BiSS* C. It is assigned by iC-Haus support@biss-interface.com.

The device ID must be specified by the slave manufacturer. It must be unique for each slave in conjunction with the manufacturer ID and is used to reference the electronic data sheet as an XML file.

If a slave is delivered in different versions (e.g. different data lengths, different data channel configurations), each version must have a different device ID.

The switchover of the version can also be carried out dynamically during operation (with register accesses or commands), however requires the immediate switchover of the device ID.

For slaves without an automatic *BiSS*-Timeout adaption, T_{MA} can be configured with bits 0...2 in address 124. Slaves with an automatic *BiSS*-Timeout adaption operate with a ROM on this memory address.

Serial number
In addition, a serial number with up to 32 bits assigned by the slave manufacturer is provided in the register area; it must have a globally unique value together with the manufacturer and device ID. This is primarily important with several slaves of the same type in order to clearly detect and assign interchanging or failure.

The values zero (0) and the maximum value (all bits "1") are reserved for the manufacturer ID, device ID and serial number. If a slave does not have a serial number, this must either be set to zero (0 = "does not exist") or the registers may not be implemented.

INITIALIZATION

BiSS C permits a large number of settings in the slave, master and bus configurations. These configurations are determined and set during initialization of the *BiSS* system.

The BiSS C bus establishment
The bus is established with software on the master side. The clock-pulse rate $1/T_{MA}$ is specified by the controller in accordance with the hardware used. During the bus establishment the minimum cycle time and the permissible range of the clock-pulse rate are determined.

During the bus establishment, first all data channels could be deactivated. Then all existing slaves are detected consecutively, their data channel parameters determined and the master configured accordingly. If an error occurs, the bus establishment can be repeated, whereby the defective slave is no longer taken into account.

In addition, during the bus establishment

- the minimum *BiSS* timeout of the slaves is determined and configured if necessary,
- the minimum and maximum permissible MA clock pulse is determined (in dependence on the master, the slaves and the line topology),
- the CRC start values are configured in the master and if possible in the slaves,
- the processing time per parameter is configured in the master and
- the minimum cycle duration is determined (from the number of bits in the data area, the processing time, the MA clock-pulse rate and the configured *BiSS* timeout), checked and configured in the master.

As an additional check, the controller can save the expected slave IDs and serial numbers of each slave and compare them to the currently determined positions during each bus establishment.

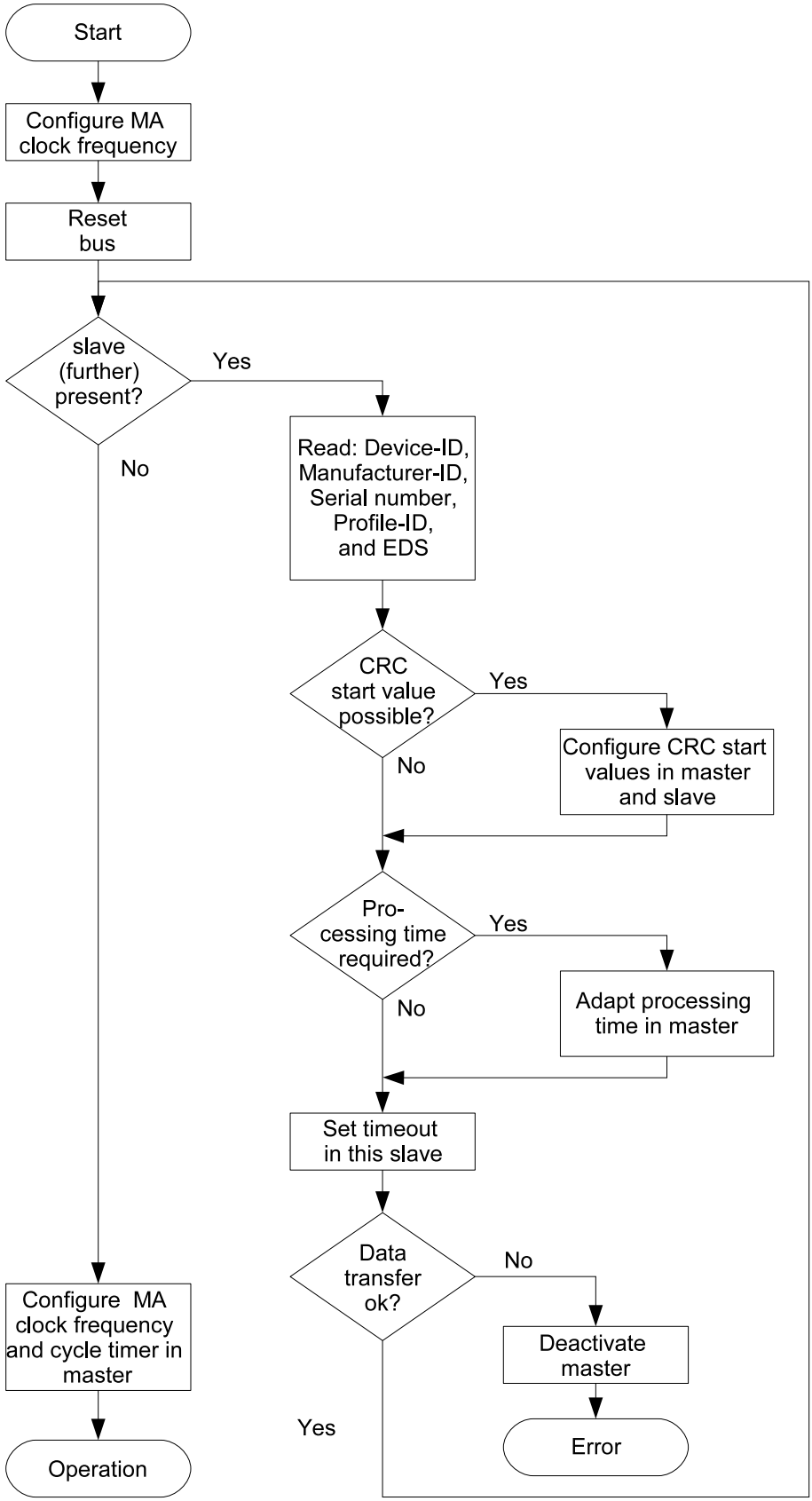


Figure 21: Procedure for a bus establishment

SYSTEM DESIGN

When designing a *BiSS* C system, it must be noted that the duration of the *BiSS* frame is variable. The cycle time should be selected long enough that the start of the *BiSS* frame is jitter-free even with the maximum timeout *BiSS*. Slaves with a processing time via request generate a jitter at the point in time of the transmission.

The BiSS C master
If the requirements are narrowly defined, user-specific custom *BiSS* C masters with reduced capabilities can be used. On the other hand, masters on which any desired number of slaves can be operated must meet the following minimum requirements. They are referred to as full *BiSS* C masters.

- At least 8 different data channels are programmable.
- At least 8*8 = 64 bytes of memory for data values (AD/SD)
- Configurable CRC check with 0 to 16 bits and variable polynomial and start value per data channel
- Sending of the shortened *BiSS* frame
- Operation in the point-to-point and bus configuration
- Programmable MA clock frequency from 80 kHz
- A programmable cycle time which starts the *BiSS* frame at the same time intervals.
- Status per *BiSS* frame: frame complete, error code
- Status per data channel: data channel completely received and data value valid (CRC).
- Status for register communication: register does not exist, register value valid (CRC), all 9 IDA bits
- Sequential register access via at least 16 registers

The BiSS C slaves
The *BiSS* C timeout should be set to 12.5 to 40 µs in the slave after switch-on to ensure automatic detection by the controller. During operation with low bit rates, a timeout of less than 12.5 µs can make it impossible to transmit a valid BiSS frame. This case cannot be distinguished by the master from an open circuit. An automatic bit rate detection can be implemented. Slaves which use a fixed short timeout during switch-on are referred to as custom *BiSS* C slaves.

When defining data channels, additional signal or error bits should be positioned on the left for the right-justified alignment of the data values, and with left-justified alignment on the right in order to maintain the alignment of the value at the byte limits.
When a device uses multiple data channels each data channel should be assigned to a logical slave. The advantage is that any data channel has its own profile-id and can be activated individual.

CHARACTERISTICS

Nr.	Symbol	Parameter	Condition	Min.	Max.	Unit
01	$1/T_{MA}$	Clock frequency	signal MA	80	[1]	kHz
02	$1/T_{MAmin}$	Minimum clock frequency	signal MA, all slave supported clock frequency	80	10000	kHz
03	t_{MAl}	Clock signal low level duration	MA = "0"	[1]	12.5	µs
04	t_{MAh}	Clock signal high level duration	MA = "1"	[1]	12.5	µs
05	$t_{BiSS\text{-}Timeout}$	*BiSS* timeout		12.5 [2]	40	µs
06	$t_{BiSS\text{-}Timeout\_s}$	reduced *BiSS*-Timeout for slaves without automatic *BiSS*-Timeout adaption on T_{MA}	after configuring bits 0. . . 2 in address 124	0.5	3	µs
07	$t_{BiSS\text{-}Timeout\_a}$	Adaptive *BiSS*-Timeout for slaves with automatic *BiSS*-Timeouts adaption on T_{MA} sampling frequency: $1/T_{CLK}$	$T_{CLK} \leq 1.5 \cdot T_{MA}$ $T_{CLK} \geq 1.5 \cdot T_{MA}$	$1.5 \cdot T_{MA}$ $1.0 \cdot T_{CLK}$	$1.5 \cdot T_{MA} + 3.0 \cdot T_{CLK}$ $1.5 \cdot T_{MA} + 3.0 \cdot T_{CLK}$	
08	$t_{LineDelay}$	Delay MA → SL	Measurable within a *BiSS*-Frame from the second rising MA edge to the first falling SL edge	0	40	µs
09	$t_{LineJitter}$	Delay jitter MA → SL	within a *BiSS* frames	-25	25	$\%T_{MA}$
10	t_{busy_s}	Processing time for single cycle data		0	40	µs
11	t_{busy_r}	Processing time for register access		0	20	ms

[1] The maximum clock-pulse rate is dependent on the transmission medium (see BiSS Interface - Physical Layer) and on the individual devices. The maximum clock-pulse rate is, among other things, stored in the EDS.
[2] The Min column is obsolete after programming via register access, automatic bit rate detection or for custom *BiSS* C slaves. The reduced timeout can prevent the *BiSS* communication at a lower clock-pulse rate; the error can only be detected as a general communication error.

Table 2: Table of characteristic

GRAPHS AND COLORS

Graphs and colors
If a transmission is displayed in the time area, the angled edges illustrate the finite steepness and an area of the unsure signal state. The *BiSS* frame is transmitted in the time area and clocked with the signal MA.

The control communication is each transmitted with one bit per cycle (=*BiSS* frame). The control frames are clocked with cycles, and the straight edges indicate the purely logical nature of the bit.

The following colors are used to make the graphs clearer:

Color	Meaning
Orange	Start of the *BiSS* frame; enables running time and processing time.
Light blue	Sensor data (Single-Cycle Sensor Data (SCDS))
Peach	Actuator data (Single-Cycle Actuator Data (SCDA))
Yellow	Control communication (commands and register accesses)

Table 3: Table of colors

TERMS AND ABBREVIATIONS

Term	Meaning
Bus establishment	Detecting all slaves including parameterizing and error handling.
Master	Interface device between *BiSS* C bus and control.
Device	Physical unit. Contains one or multiple slaves.
Slave	Logic unit within a device. Occupies one single ID and uses none, one or multiple data channel.
BiSS-Frame	Cyclically executed and carries single cycle data completely and one bit at a time of register data.
BiSS-Timeout	Timeout give by the connected slaves. Separates following *BiSS* frames.
"First slave"	The slave with the data which are transmitted first with the *BiSS* frame.
"Last slave"	The slave with the data which are transmitted last with the *BiSS* frame.
Data area	The bits in the *BiSS* frame which the data channels transmit.
Data channel	A logical channel which contains the data and check bits of a data value and is parameterized as such.
Data value	A value which is transmitted protected. It can contain additional bits and bit groups, however is treated by the *BiSS* master as a number.
Register bank	A switchable area of 64 registers in which registers or memory of the slave can be displayed.
Controller	The controller is the higher-ranking logic of the *BiSS* master. This is usually understood to be the processor which controls the *BiSS* master and the software running on it.
Big Endian	Byte sequence with which the highest-value byte is saved at the lowest-value address.

Table 4: Table of terms

Abbreviation	Meaning	Description
SCD	Single-Cycle Data	A value transmitted completely in one *BiSS* frame.
CDM	Control Data Master	Control Data Master for control communication. Is sent in the *BiSS* timeout.
CDS	Control Data Slave	Control Data Slave for control communication. Is sent within the *BiSS*-Frame after the start bit.
DCH	Data channel	Logic unit for a concurrent secured transmission of multiple independent data.
SLI	Slave In	Data input of the *BiSS* C slave.
SLO	Slave Out	Data output of the *BiSS* C slave.
MO	Master Out	Data output of the *BiSS* C master.
MA	Master Clock	Clock pulse output of the *BiSS* C master.
SL	Slave return	Data input of the *BiSS* C master.
EDS	Electronic Data Sheet	Electronic Data Sheet.
XML	Extensible Markup Language	A language for describing documents.
CRC	Cyclic Redundancy Check	Procedure for transmission checking.
MSB	Most Significant Bit	Highest-value bit
LSB	Least Significant Bit	Lowest-value bit
AD	Actuator data	actuator data to control the actuator
SD	Sensor data	sensor data of the sensor

Table 5: Table of abbreviations

REVISION HISTORY

Rel.	Rel. Date*	Chapter	Modification	Page
A1	2007-07-09		Initial release	all

Rel.	Rel. Date*	Chapter	Modification	Page
C6	2016-02-26		Drive controllers, Rotary encoder, Linear encoder and Communication watchdogs added to applications	1
		OPERATING DESCRIPTION	Figure 3 updated: T_{busy} starts at first rising edge of MA(latch), not first falling edge of SLO	4
		CONTROL COMMUNICATION	Figure 10 reference in text updated	9
		CONTROL COMMUNICATION	Details on the reduced BiSS frame added	10
		CONTROL COMMUNICATION	BiSS Commands 0b00 and 0b01 added	11
		CONTROL COMMUNICATION	Details on applications with more than 8 IDs added	11
		CONTROL COMMUNICATION	Details on Register Protection added	13
		TERMS AND ABBREVIATIONS	Abbreviation table updated	20
		REVISION HISTORY	Chapter added	21
			Minor text corrections	5, 8, 11, 13, 14, 15, 17

* Release Date format: *YYYY-MM-DD*