BiSS Interface

PROTOCOL DESCRIPTION

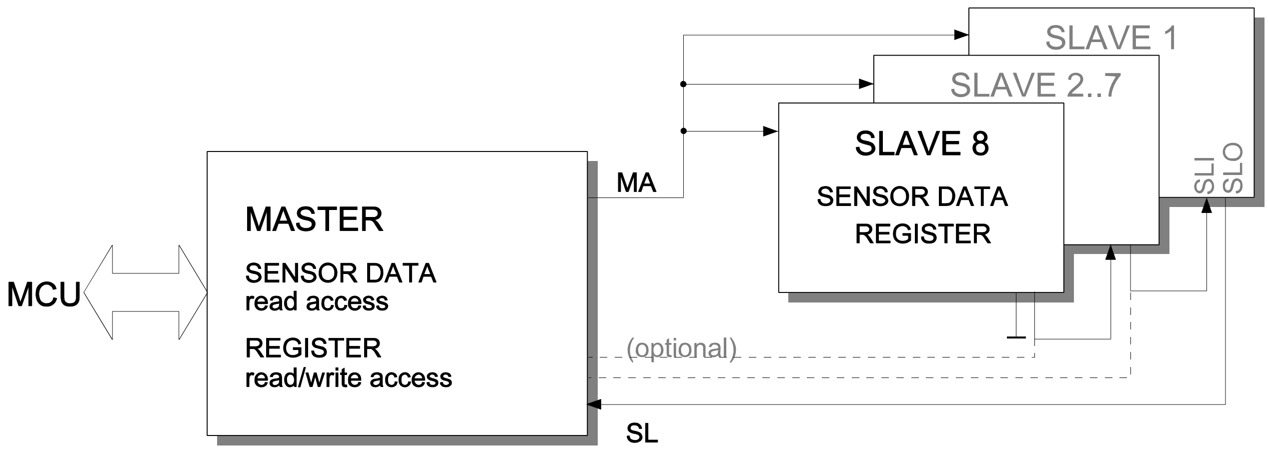
FEATURES

- Data transmission via two unidirectional lines
- Hardware-compatible with SSI
- Up to 8 subscribers possible (single master, multi slave)
- Serial or parallel connection of subscribers to the master
- Cyclic readout of sensor data of any length (as stipulated by the hardware)
- Automatic run-time compensation by the master enables high readout data rates
- Acquisition of sensor process time possible (delayed transfer)
- Synchronous data storage in all subscribers triggered by the master
- Bidirectional access to subscriber registers permits sensor parameterization
- Fail-safe register communication through data checking (CRC); also available for sensor data, depending on the hardware implementation

APPLICATIONS

- Industrial sensor bus, e.g. for position sensing at several axles
- Motor feedback
- Chip communication in multi-sensor systems

BLOCK DIAGRAM



INTRODUCTION OF CONFIGURATION

This specification describes a serial sensor interface protocol for synchronous, fast and safe readout of sensor data, as well as for bidirectional access to the sensor registers. The interface is hardware-compatible with the SSI interface and has two or more unidirectional lines.

A system consists of a controller (master) and one to eight sensors (slaves), which can be in a serial or semiparallel connection. For either type of connection the 'MA' line, originating from the master, is connected to all slaves.
With the serial connection of the subscribers an input (SLI) at the sensor (connected to the output of the predecessor) enables a read of data from this sensor and, as a result, also the bus operation with several subscribers (Figure 1). The master successively receives the data of the slaves.

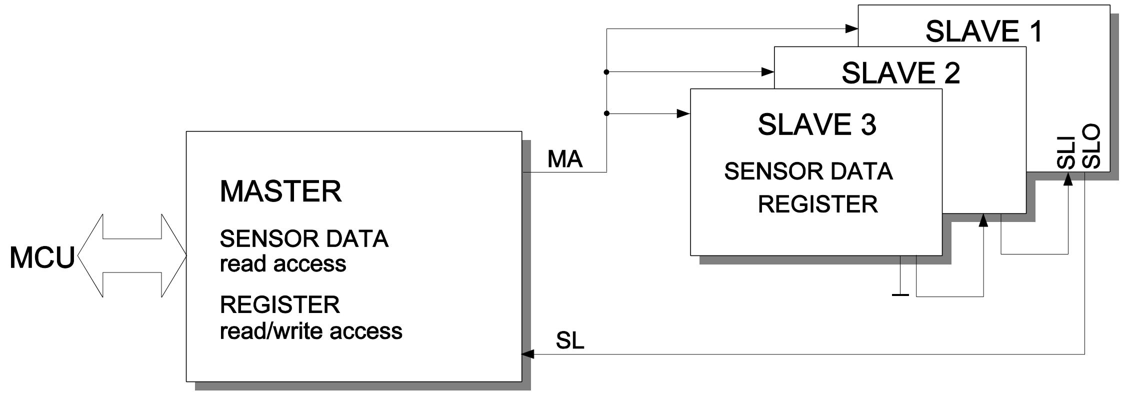


Fig. 1: Serial subscriber connection

In principle, there is also the possibility of directing the outputs of the subscribers in a parallel connection back to the master (lines SL0..SL2 in Figure 2) allowing to simultaneously read in the responses of all slaves. This method causes more effort for the master, however, it provides the advantage of shorter transmission times.

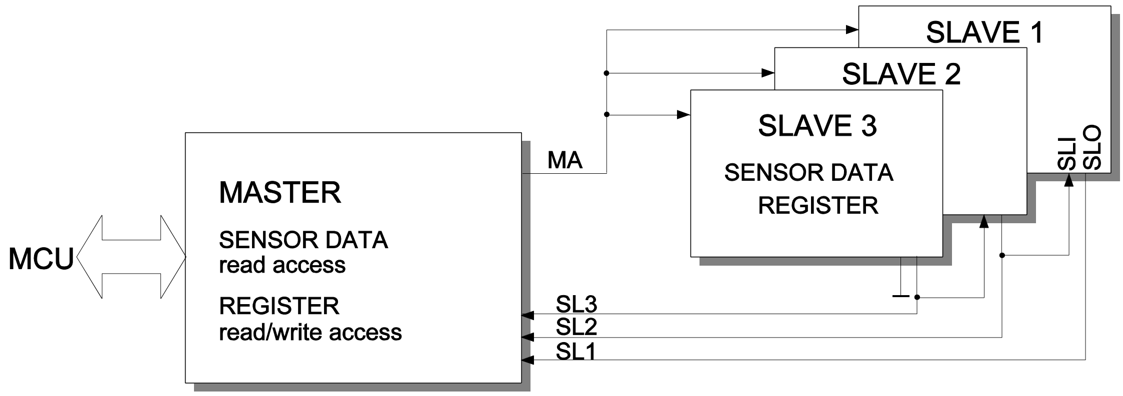


Fig. 2: Semiparallel subscriber connection

INTRODUCTION OF COMMUNICATION

The bus protocol enables two types of communication: 'Sensor Mode' and 'Register Mode'. In case of a fault it guarantees that the slave will always return into a defined state.

In the sensor mode a cyclic fast readout of sensor data is available without requiring addressing. With initializing the communication, line delays are evaluated and automatically compensated by the master to permit high data rates. The master then simply continues the clock signal necessary to output the slave sensor data (Figure 3). To ensure the absence of faults of the transmitted data a corresponding hardware based check is advisable.

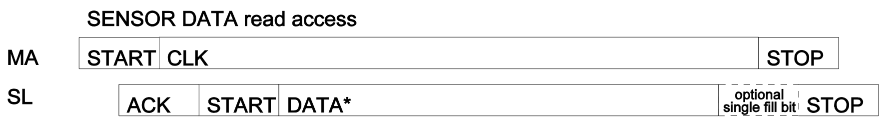


Fig. 3: Signals in sensor mode

In the register mode the registers in the individually addressable sensors can be written or read. For this purpose, *BiSS* provides a special addressing sequence (Figure 4) where 3 bits are used for the subscriber addressing and 7 bits are used for the register addressing. To increase the transmission reliability the data of the addressing sequence is protected by a 4-bit CRC.

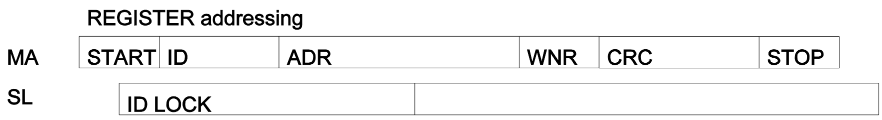


Fig. 4: Signals in register mode - address sequence

The 7 bits for the register addressing allow to address 128 registers of 8 bits each per slave. If this address range is not sufficient, it is possible to allocate additional blocks with 128 registers each to a slave, with each of the additional blocks reducing the number of slaves available within the system by one.

For reading out data of a register the master has only to provide the corresponding number of clocks after the addressing sequence. During write access to the registers the master transmits the data to be registered as PWM-coded data.

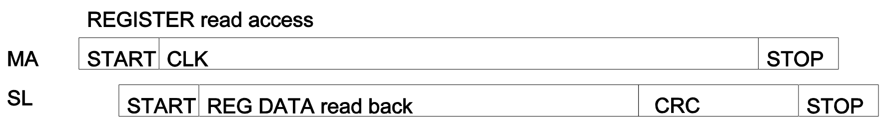


Fig. 5: Signals in register mode - read access

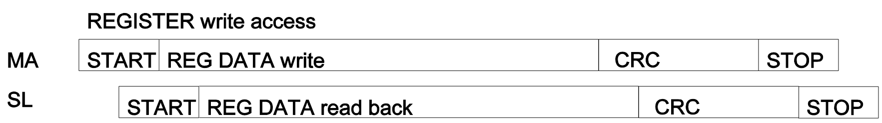


Fig. 6: Signals in register mode - write access

PROTOCOL DESCRIPTION

Communication Modes

The *BiSS* sensor interface allows to either perform a very fast readout of data or to perform write and read transfers to subscriber registers. The switching between these two communication modes is performed through a time condition at the start of each communication cycle. If the first low level of the master signal is longer than the (configurable) time 'timeoutSENS', the sensors will be set to register mode. If it is shorter than 'timeoutSENS', they will be set to sensor mode.

'SENSOR MODE' Communication

SINGLE-SENSOR OPERATION or SEMIPARALLEL SUBSCRIBER CONNECTION
After sending the first falling edge the controller (master) must generate a rising edge within the time interval 'timeoutSENS' to initialize a readout of the sensor data. Following this edge the sensor (slave) stores the current measurement data or starts the conversion of captured measurement values. The second rising edge prompts the sensor to also generate a low level at its output (Figure 7, Part S1).

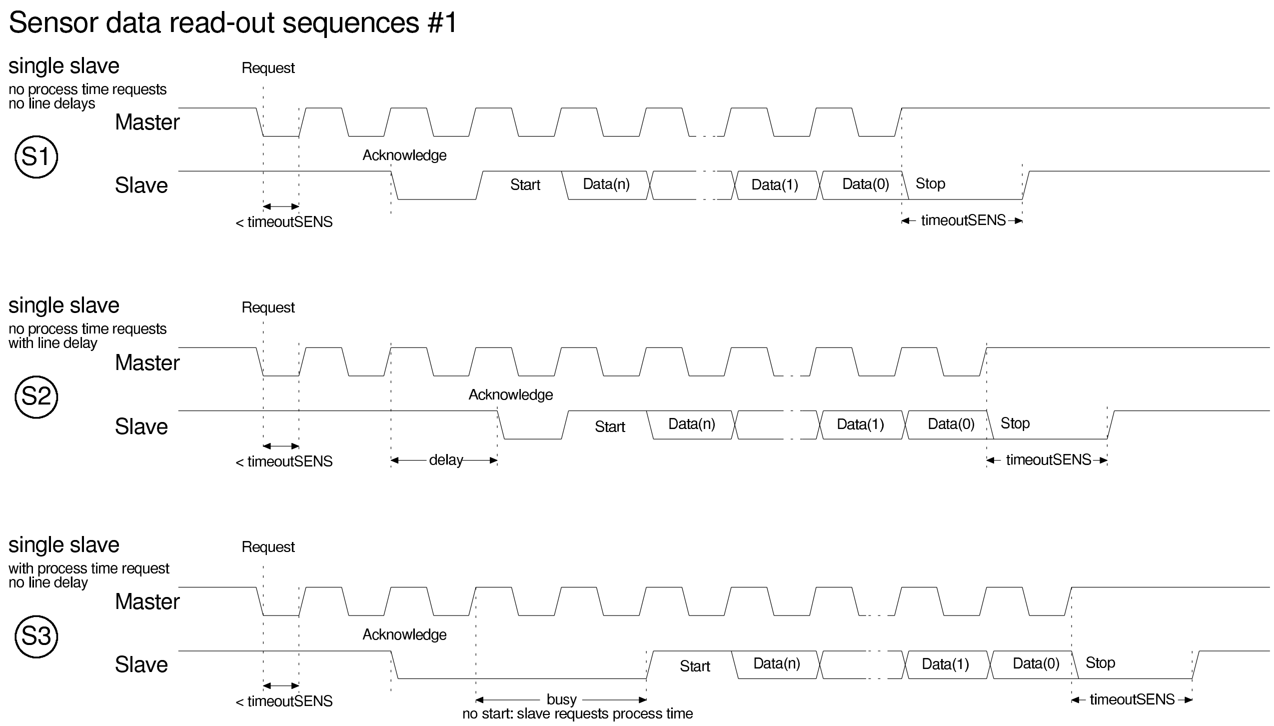


Fig. 7: Reading sensor data from a single slave:
S1: without delay, without processing time
S2: with delay, without processing time
S3: without delay, with processing time

Based on the time difference between the second rising master edge and the slave response the master can determine the line delay and compensate it by correspondingly shifting the sampling of the slave signal. This enables an accelerated communication (Figure 7, Part S2).

After this initialization of communication the output signal of the master is used to clock out the sensor data of the slave with the rising edges. In this process the slave response starts with a start bit ('1') and ends with a stop bit ('0'). This allows the slave to correspondingly request processing time by delaying the start bit (Figure 7, Part S3).

SERIAL SUBSCRIBER CONNECTION

With the serial connection of several slaves the data of the 'back' slaves are passed through the 'front' slaves (i.e. located nearer to the input of the master). Here, the data bits of the individual slaves directly succeed without a separation through start or stop bits. Figure 8 illustrates the corresponding signal traces for the serial connection of two slaves with Slave1 being the 'front' one and Slave2 being the 'back' one.

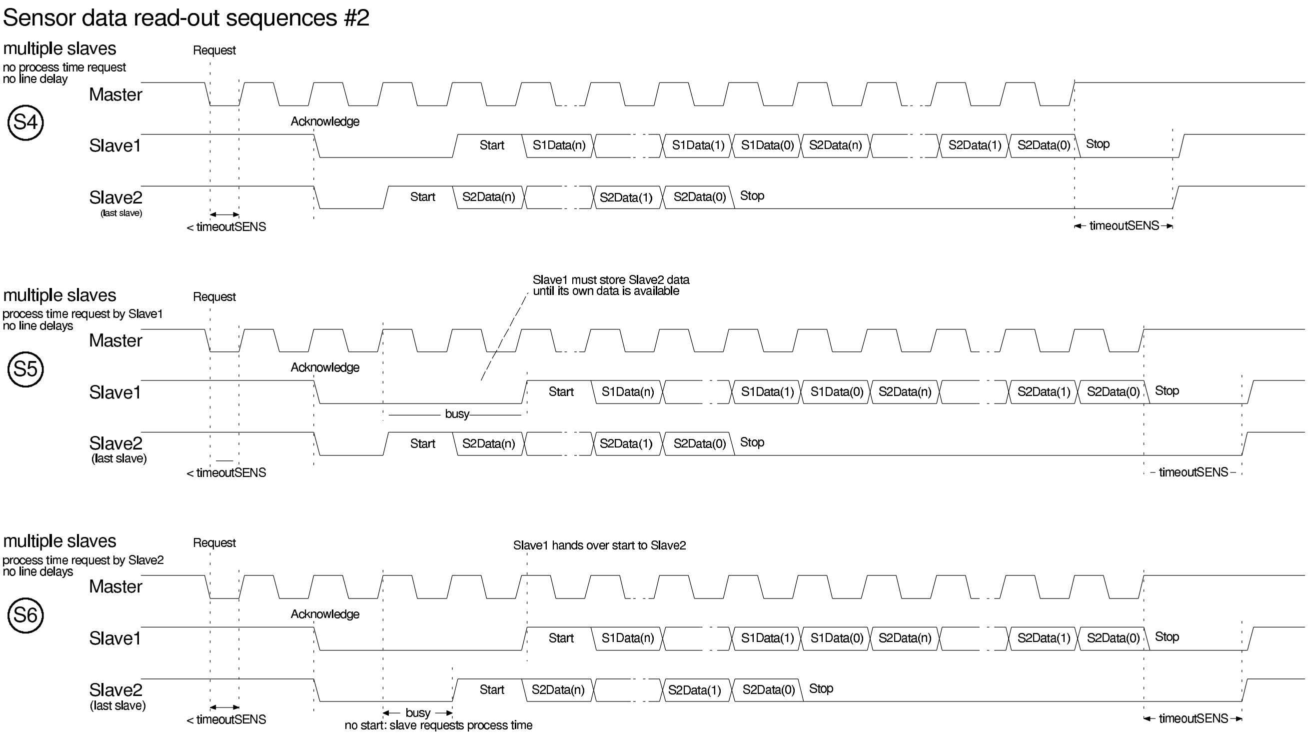


Fig. 8: Reading sensor data from two slaves connected in series (displayed without line delay)
S4: without processing time
S5: with processing time of Slave1
S6: with processing time of Slave2

If a 'front' located slave requires processing time it must buffer the incoming signals of the back slaves during this period (Figure 8, Part S5). If a 'back' located slave delays its start signal because it requires processing time, the corresponding delay will be passed on to the master by the slaves located in front (Figure 8, Part S6).

'REGISTER MODE' Communication

The register mode is entered when the master holds the first low level of a communication cycle for longer than the 'timeoutSENS' period. After the 'timeoutSENS' period has expired the slave also responds to the falling edge of the master with a falling edge at its output, which signalizes the master that the actual data transmission can be started. This data transmission is PWM-coded, which allows an easy transmission of clock and data through the master line. However, the line delays must be small compared to the clock rate used.

First, the device and register addressing is carried out, then the clock out of the register data of the slave or the write-in of new data follows.

The addressing is performed by sending one start bit ('1'), followed by a 3-bit slave ID, a 7-bit register address, 1 bit ('WNR') for read/write switching, a 4-bit CRC (CRC polynomial '10011b', inverted output) and one stop bit ('0'). During the addressing sequence each slave, starting from the 'backmost' slave, will occupy as many slave IDs as required to address all of its registers. This is accomplished by each slave responding with a corresponding delay of its rising signal edge (see Figure 9 and Application Notes).

Slave addressing (register mode)

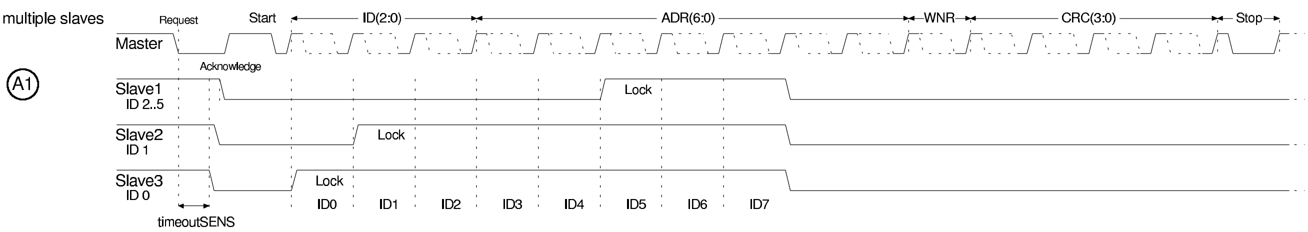


Fig. 9: Slave addressing in register mode

READ ACCESS TO REGISTERS

If the transmitted WNR bit is a '0' one or more registers will be read out after the completion of the addressing sequence. At that point in time the slave output is on a low level. If the transmitted slave ID is identical with the own slave ID the slave responds with a start bit ('1') (Figure 10, Part R1). If the slave requires additional processing time it also will delay the start bit output correspondingly (Figure 10, Part R2).

The protocol allows that one addressing can be used to read (or write) several consecutive registers. For this, the master continues clocking and the slave internally counts up the address (auto-increment) and responds with a new data word, which consists of a start bit ('1'), the actual data bits followed by four CRC bits (CRC polynomial '10011b', inverted output) and a stop bit ('0') (Figure 10, Part R3).

At this point, it could happen that forbidden or non-existing registers are tried to be read out. In this case the slave responds by providing a continuous '0' at its output (Figure 10, Part R4).
If a slave requires additional processing time during the readout of a register, which has been addressed by the automatic incrementing, it can request the processing time by delaying the start bit of the corresponding data word (Figure 10, Part R5).

Figure 10, Part R6 illustrates the behaviour of several slaves during the readout of registers. A sensor (in the example Slave3) that 'succeeds' the addressed slave provides a continuous '0', the addressed device (Slave2) outputs its register data and the sensor (Slave1) that 'precedes' the addressed slave transparently passes on the data received from the predecessor but also generates an own stop bit.

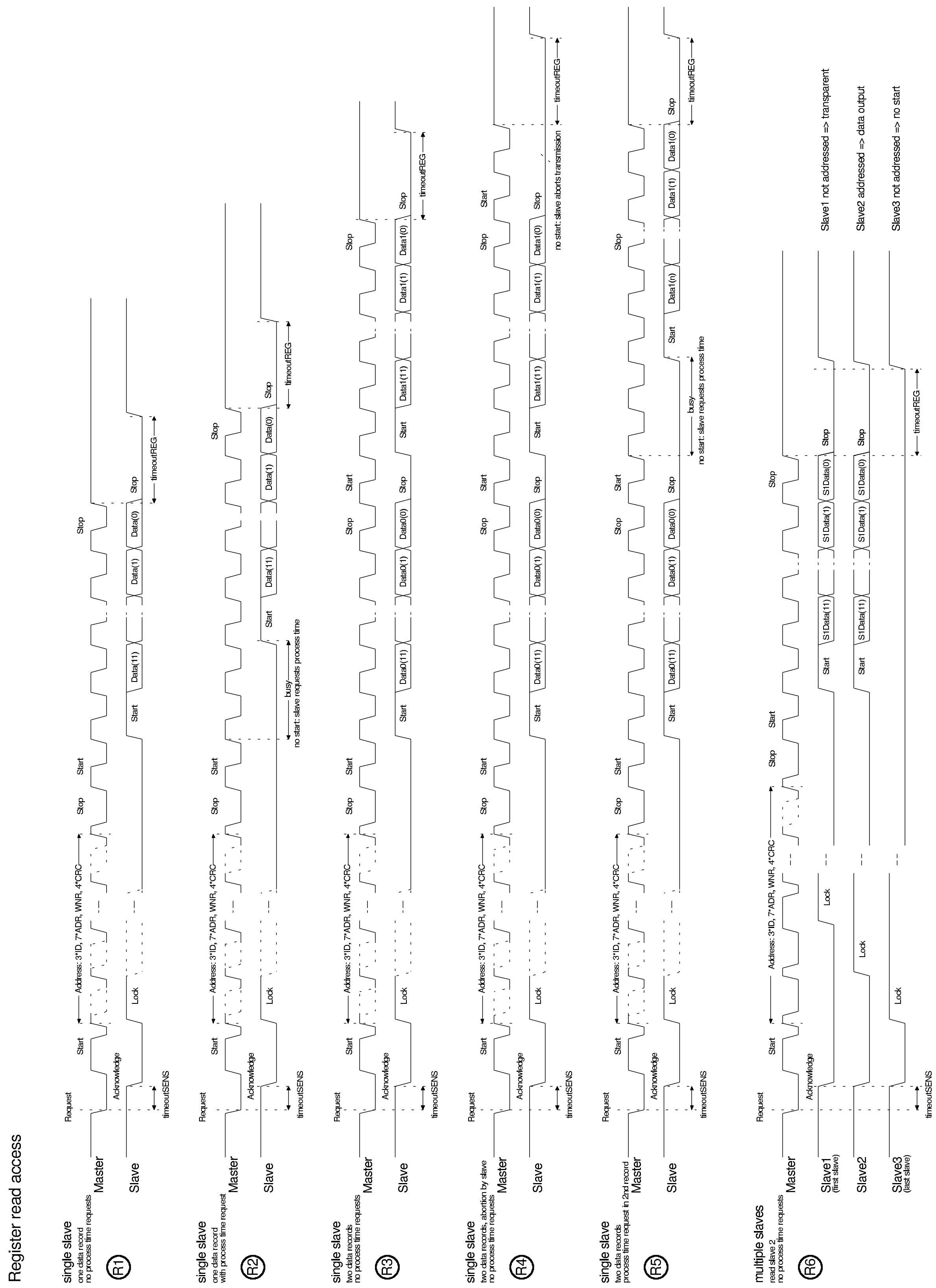


Fig. 10: Read access to registers - R1..R2: Reading a single data word
R3..R5: Reading several data words
R6: Reading a data word with several slaves in a serial connection
Data(11) to Data(0): 8 bit register data plus 4 bit CRC

WRITE ACCESS TO REGISTERS

If the WNR bit transmitted in the addressing sequence is a '1', the master, after this sequence, will transmit the 8 data and 4 CRC bits - framed by a start bit ('1') and a stop bit ('0') - PWM coded to the addressed slave (Figure 11, Part W1). The slave returns the written data uncoded back to the master, which, as a result, can monitor the communication bit by bit and respond in case of an error.

As before, it can also happen during the write-in of register data that the slave requires processing time. In this case the slave delays its start bit by the required amount of time. During that time the master cannot transmit any data to the slave register and, therefore, must repeat the transmitted bit until the slave responds (Figure 11, Part W2, Part W5 for several data sets).

As with the readout, the write access to a slave also allows that several consecutive registers can be written in one write-in cycle, by successively transmitting consecutive data words - framed each by a start bit ('1') and a stop bit ('0') - (Figure 11, Part W3). If forbidden or non-existing registers are addressed the slave will interrupt the transmission by providing a continuous '0' instead of returning the received start bit (Figure 11, Part W4).

Figure 11, Part W6 illustrates the signal forms during write-in access to a slave in a serial connection of sensors. Sensors (Slave3) succeeding the addressed slave provide a continuous '0' at the output; the addressed slave (Slave2) returns an uncoded output of the PWM data, which have been read in; preceding sensors (Slave1) transparently pass on the data of the predecessors but also generate an own stop bit.

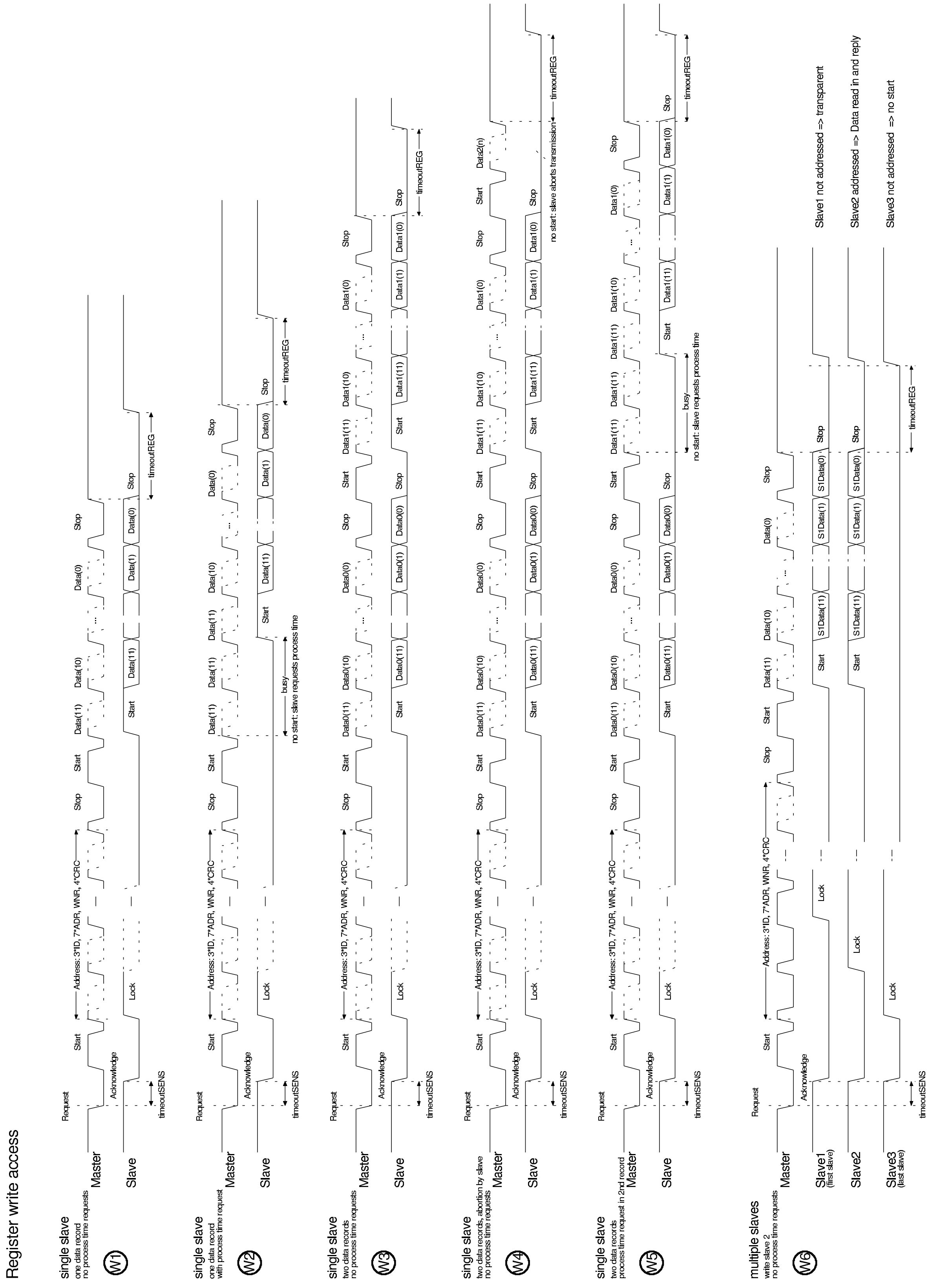


Fig. 11: Write access to registers - R1..R2: Writing a single data word
R3..R5: Writing several data words
R6: Writing with several slaves in a serial connection
Data(11) to Data(0): 8 bit register data plus 4 bit CRC

APPLICATION HINTS

Declaration of a Slave Type Identification

Any bus subscriber, integrated devices or other circuits provide the system with an unambiguous type identification. The *BiSS* master expects this identification at the register addresses '78h' and '7Fh' (6 bytes for the manufacturer's product code, 2 bytes for the name of manufacturer) and then configures itself for the connected subscribers. iC-Haus provides support regarding the allocation of type identifications.

Extension of the Register Address Range

The protocol of the sensor interface provides 3 bits for the slave identification and 7 bits for the register selection as standard range. If a sensor contains more than the 128 registers, which can be addressed with 7 bits, it can occupy two or more slave IDs during the addressing sequence by delaying the output of the lock bit by the corresponding number of clocks (see Figure 9, Slave1). As a result, it will then respond to several slave IDs instead of only one. The number of slaves available within the system will be reduced corresponding to the occupation of several slave IDs by a single subscriber.

It is advisable to place subscribers with two or four required slave IDs in the overall circuitry so that the occupied slave IDs are only different in the LSB(s). In this case these bits of the slave ID can be directly used as most significant register addressing bits without requiring an address mapping (see Figure 12).

Register mode address sequence
(up to 8 slaves, for 3-bit slave ID and 7-bit register address)

<table>
<tr><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th></tr>
<tr><td>START</td><td colspan="3">ID(2:0)</td><td colspan="7">ADR (6:0)</td><td>WNR</td><td colspan="4">CRC</td><td>STOP</td></tr>
</table>

Register mode address sequence
(for 2-bit slave ID, 8-bit register address)

<table>
<tr><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th></tr>
<tr><td>START</td><td colspan="2">ID(2:1)</td><td colspan="8">ADR (7:0)</td><td>WNR</td><td colspan="4">CRC</td><td>STOP</td></tr>
</table>

When using two slave IDs (eg. 100 and 101) the LSB of the slave ID can also be used as the #8 bit of the register address.

Fig. 12: Bit schematics with 7- and 8-bit register address

Example: A subscriber with more than 128 but less than 256 registers is placed so it occupies the addresses '100b' and '101b'. The LSB of the slave ID then corresponds to the MSB of the register address. As a result, the subscriber is allocated the 2-bit slave ID '10b'.

Error Checking

The *BiSS* protocol already includes a 4-bit CRC error checking for the addressing sequence and the register data. A similar checking is advisable for the transmission of the sensor data, respectively. For example, a 6-bit CRC should be configured and transmitted with the data to perform an error checking of a 20-bit measurement data.