iC-MCW

BiSS WATCHDOG FOR RS422 WITH SPI

preliminary

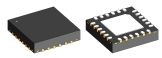
FEATURES

- Acquisition of BiSS Single Cycle Data
- Acquisition of BiSS Control Data
- Monitoring of BiSS Communication Status
- Error Diagnostics for Debugging
- Pin for Simple Error Indication
- Auto Compensation of Line Delay and Conversion Times
- Internal 14-bit BiSS Frame Counter
- Built-In RS422 Receiver for Direct Interfacing with BiSS
- Support of BiSS Data Transfer Rates of up to 10 MBit/s
- Serial Controller Communication via SPI™ Slave Interface
- Support of Interrupt-Driven I/O
- Built-In System and High Speed Sample Clock Generation
- Built-In Power on Reset
- 3.3 V to 5 V Supply (+/-10%)
- Industrial Temperature Range

APPLICATIONS

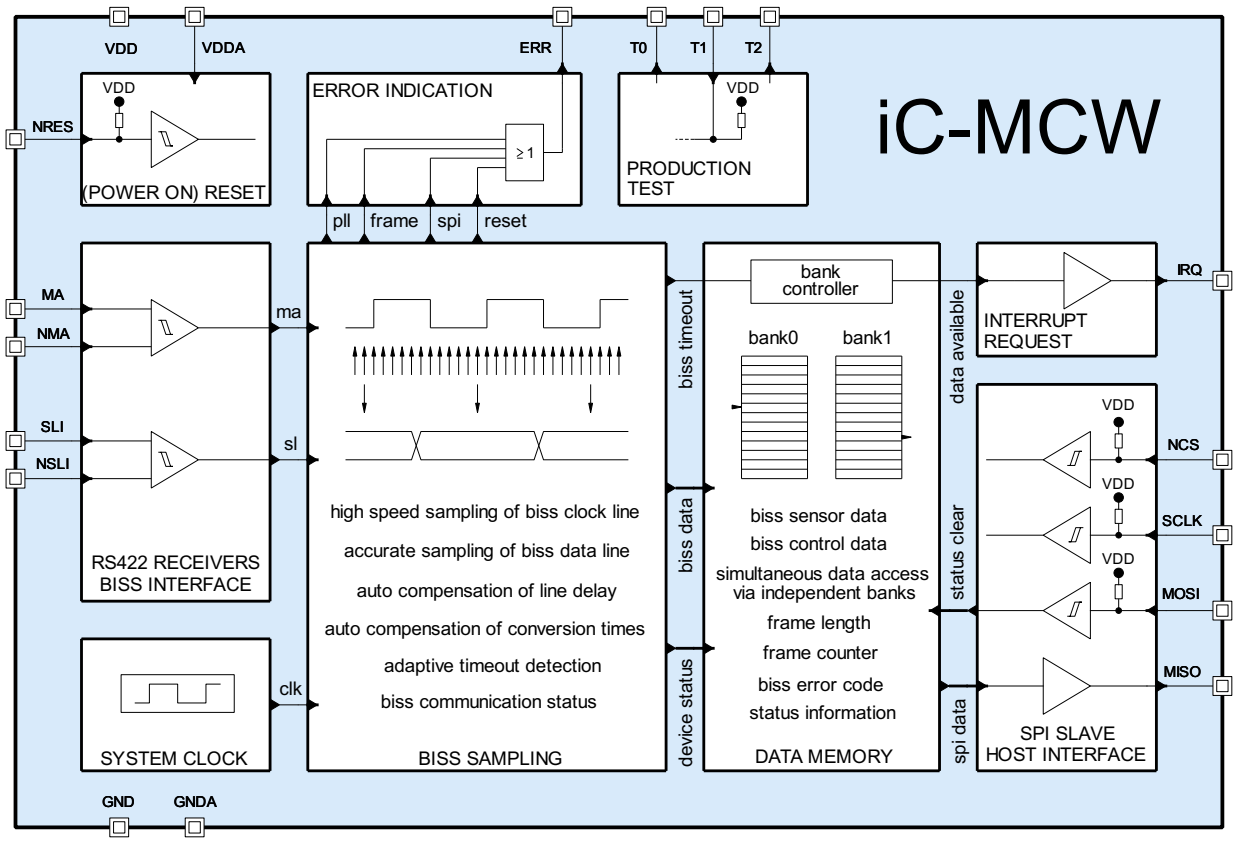
- BiSS Communication Monitoring
- BiSS Safety Extension of BiSS Standard Drives

PACKAGES



QFN24-4x4

BLOCK DIAGRAM



DESCRIPTION

The iC-MCW is a BiSS C communication monitoring device to be connected to a microcontroller via standard SPI. The integrated circuit observes the BiSS C protocol sequence and samples sensor and control data at maximum transmission rates of 10 MBit/s. The iC-MCW includes RS422 receivers to be connected to the differential BiSS transmission lines directly. Due to built-in clock generation, the chip does not need any external oscillators.

An adaptive BiSS timeout detects the end of a received BiSS frame and triggers an interrupt request to notify the microcontroller about new BiSS communication status and frame data. Alternatively, it is possible to constantly poll the chip status, which indicates the availability of new frame data as well.

The frame data is stored in internal random access memory, which is organized into two banks. Hence, current frame data can be read through the SPI interface while a new BiSS frame is sampled. If the frame data of one bank has not been fetched in time, it will be overwritten with new frame data. iC-MCW informs the microcontroller about lost frames indirectly by counting the BiSS timeout of each incoming BiSS frame. The internal frame counter is available through SPI as well.

Detailed status and error information are available by register access. Critical errors are mapped to the error pin and must manually be reset via SPI.

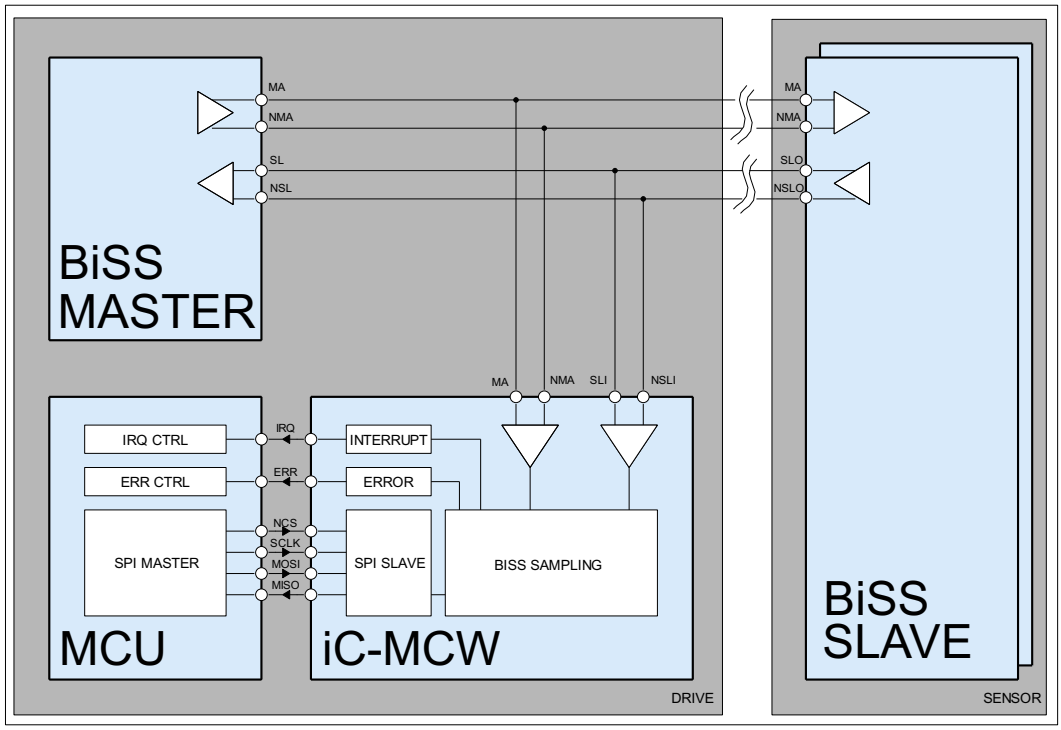


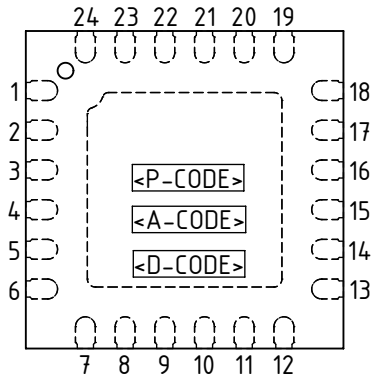
Figure 1: Integration of the iC-MCW in a BiSS Communication System

Figure 1 shows the extension of a BiSS communication system with the iC-MCW. The standard drive establishes a common differential BiSS communication line to one or more BiSS slaves by BiSS master. The BiSS clock and BiSS data lines are connected to the RS422 receiver within the iC-MCW. The microcontroller is connected to the SPI interface of the iC-MCW and can import the BiSS frame data without disrupting the transmission between BiSS master and BiSS slave. Thus, it is possible to implement any additional features to extend the standard drive to match safety requirements, for example.

The device offered here is a multifunctional iC that contains integrated BiSS C interface components. The BiSS C process is protected by patent DE 10310622 B4 owned by iC-Haus GmbH. Users benefit from the open BiSS C protocol with a free license which is necessary when using the BiSS C protocol in conjunction with this iC.
Download the license at
www.biss-interface.com/bua

PACKAGING INFORMATION TO JEDEC

PIN CONFIGURATION QFN24 4mm x 4mm



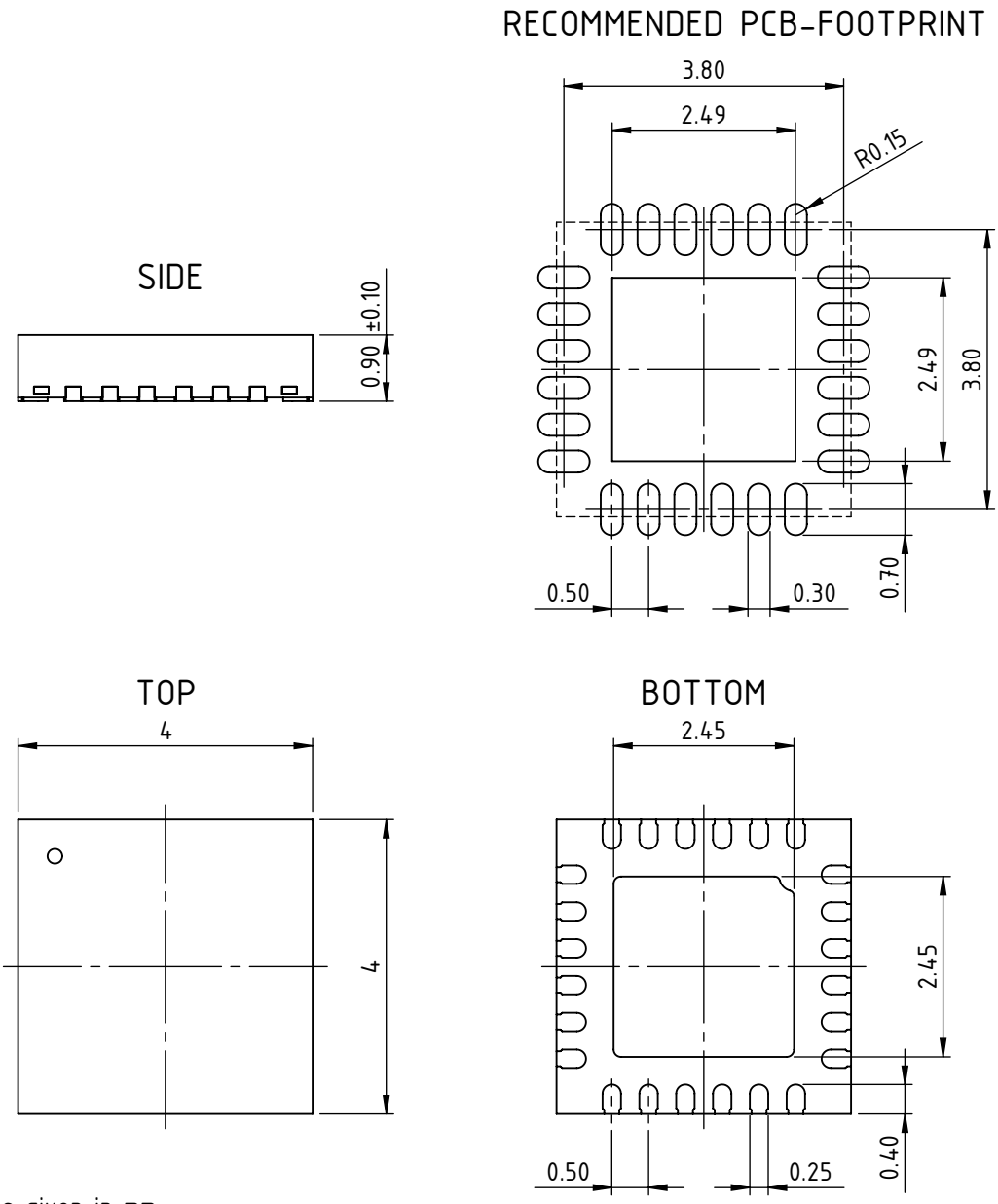
PIN FUNCTIONS

No.	Name	Function
1	T0	Test Pin [1]
2	n.c.	[2]
3	VDDA	+3.3 V . . . +5 V Analog Supply Voltage
4	n.c.	[2]
5	GNDA	Analog Ground
6	n.c.	[2]
7	n.c.	[2]
8	n.c.	[2]
9	SLI	BiSS Data Line Input
10	NSLI	BiSS Data Line Input (inverted)
11	MA	BiSS Clock Line Input
12	NMA	BiSS Clock Line Input (inverted)
13	n.c.	[2]
14	IRQ	Interrupt Request Output
15	GND	Digital Ground
16	T1	Test Pin [3]
17	NRES	Reset Signal Input (low active)
18	ERR	Error Output
19	MOSI	SPI Serial Data Input
20	MISO	SPI Serial Data Output
21	NCS	SPI Chip Select Input
22	SCLK	SPI Clock Input
23	VDD	+3.3 V . . . +5 V Digital Supply Voltage
24	T2	Test Pin [1]
	BP	Backside Paddle [4]

IC top marking: <P-CODE> = product code, <A-CODE> = assembly code (subject to changes).
1) Test Pins T0 and T2 shall not be connected.
2) Pin numbers marked n.c. are not connected.
3) Test Pin T1 is low active, requires external high for normal operation.
4) Connecting the backside paddle to GNDA is recommended. A current flow across the paddle is not permissible.

PACKAGE DIMENSIONS QFN24 4mm x 4mm

All dimensions given in mm.



All dimensions given in mm.
Tolerances of form and position according to JEDEC MO-220.

drb_qfn24-4x4-1_pack_1, 10:1

ABSOLUTE MAXIMUM RATINGS

Beyond these values damage may occur; device operation is not guaranteed.

Item No.	Symbol	Parameter	Conditions	Min.	Max.	Unit
G001	VDD	Supply Voltage VDD		-0.3	6	V
G002	I(VDD)	Current in VDD		-100	150	mA
G003	VDDA	Analog Supply Voltage VDDA		-0.3	6	V
G004	I(VDDA)	Current in VDDA		-100	150	mA
G005	V()	Voltage at all pins, excluding VDD, VDDA, GND and GNDA		-0.3	6	V
G006	I()	Current in all pins excluding VDD, VDDA, GND and GNDA		-100	150	mA
G007	V_{esd}()	ESD Susceptibility at all pins	HBM, 100 pF discharged through 1.5 kΩ		2	kV
G008	Tj	Operating Junction Temperature Range		-40	125	°C
G009	Ts	Storage Temperature Range		-40	85	°C

THERMAL DATA

Operating Conditions: VDD = 3.0 V ... 5.5 V

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
T01	Ta	Standard Operating Ambient Temperature Range		-40		125	°C
T02	$R_{thjaQFN}$	Thermal resistance chip/ambient	QFN24 package mounted on PCB, *thermal pad* at approx. 6 cm² cooling area		80		K/W

All voltages are referenced to ground unless otherwise stated.
All currents flowing into the device pins are positive; all currents flowing out of the device pins are negative.

ELECTRICAL CHARACTERISTICS

Operating Conditions: VDD = VDDA = 3.0 V ... 5.5 V, Tj = -40 ... 125 °C, unless otherwise stated

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
General							
001	VDD, VDDA	Permissible Supply Voltage		3		5.5	V
002	I(VDD)	Supply Current in VDD	VDD=3 V, outputs not loaded VDD=5.5 V, outputs not loaded	2 4	5 10	8 16	mA mA
003	I(VDDA)	Supply Current in VDDA	VDDA=3 V, outputs not loaded VDDA=5.5 V, outputs not loaded	130 160	330 400	530 640	µA µA
004	Vc()hi	Clamp Voltage hi at all pins excluding VDD, VDDA, GND, GNDA	Vc()hi = V() - VDD, I() = 1 mA	0.3		1.5	V
005	Vc()lo	Clamp Voltage lo at all pins excluding VDD, VDDA, GND, GNDA	Vc()lo = V(), I() = -1 mA	-1.5		-0.3	V
SPI Host Interface: NCS, SCLK, MISO, MOSI							
A01	Vs()hi	Saturation Voltage hi at MISO	Vs()hi = VDD - V() I() = -1.6 mA			400	mV
A02	Vs()lo	Saturation Voltage lo at MISO	Vs()lo = V() I() = 1.6 mA			400	mV
A05	Vt()hi	Threshold Voltage hi at NRES, NCS, SCLK and MOSI				70	%VDD
A06	Vt()lo	Threshold Voltage lo at NRES, NCS, SCLK and MOSI		30			%VDD
A07	Vt()hys	Threshold Voltage Hysteresis at NRES, NCS, SCLK and MOSI		200			mV
A08	Ipu()	Pull-Up Current at NRES, NCS, SCLK and MOSI	V() = 0 V..VDD - 1 V VDD = 3 V VDD = 5.5 V	 -16 -61		 -3.3 -7	 µA µA
A09	fclk()	Permissible Clock Frequency at SCLK		10			MHz
A10	T_{int}	Interrupt Time		4.5	5.6	8	µs
RS422 BiSS Interface: MA, NMA, SLI, NSLI							
B01	Vin()	Permissible Input Voltage		-10		10	V
B02	Vcm()	Input Common Mode Voltage		-7		7	V
B03	Vdiff()	Differential Input Voltage	Vdiff() = V(MA) - V(NMA)	-12		12	V
B04	Rin()	Input Resistance	MA/SLI vs GND, NMA/NSLI vs GND	4			kΩ
B05	Vt()hi	Differential Input Threshold hi	Vdiff(MA/SLI) = V(MA/SLI) - V(NMA/NSLI)			200	mV
B06	Vt()lo	Differential Input Threshold lo	Vdiff(MA/SLI) = V(MA/SLI) - V(NMA/NSLI)	0			mV
B07	Vt()hys	Differential Input Hysteresis	Vdiff(MA/SLI) = V(MA/SLI) - V(NMA/NSLI)	5	55		mV
B08	fclk()	Permissible Frequency at MA		10			MHz
B09	tout()	Adaptive BiSS Timeout			1.5 x 1/fclk		
B10	T_{Smp}	Internal Sample Period		4.8		9	ns
Ports: IRQ, ERR, NRES							
C01	Vs()hi	Saturation Voltage hi at IRQ and ERR	Vs()hi = VDD - V() I() = -1.6 mA			400	mV
C02	Vs()lo	Saturation Voltage lo at IRQ and ERR	Vs()lo = V() I() = 1.6 mA			400	mV
C05	Vt()hi	Threshold Voltage hi at NRES				70	%VDD
C06	Vt()lo	Threshold Voltage lo at NRES		30			%VDD
C07	Vt()hys	Threshold Voltage Hysteresis at NRES		200			mV
C08	Ipu()	Pull-Up Current at NRES	V() = 0 V..VDD - 1 V VDD = 3 V VDD = 5.5 V	 -16 -61		 -3.3 -7	 µA µA

ELECTRICAL CHARACTERISTICS

Operating Conditions: VDD = VDDA = 3.0 V ... 5.5 V, Tj = -40 ... 125 °C, unless otherwise stated

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
Internal Oscillator:							
D01	fosc	Internal Oscillator Frequency		16	23	30	MHz
Power-On Reset:							
E01	VDDoff	VDDA Turn-Off Threshold (Undervoltage Reset)	VDDA decreasing	2.55		2.85	V
E02	VDDon	VDDA Turn-On Threshold	VDDA increasing	2.65		2.98	V
E03	VDDhys	Undervoltage Hysteresis	VDDhys = VDDon - VDDoff VDD = 3 V, VDD = 5.5 V,	 75 15	 130 155		 mV mV

OPERATING REQUIREMENTS: BiSS Interface - BiSS C Frame

Operating conditions: VDD = VDDA = 3.0 ... 5.5 V, Tj = -40 ... 125 °C

Item No.	Symbol	Parameter	Conditions	Min.	Max.	Unit
Sensor Data Cycle						
I001	T_{MAS}	Clock Period		80	12500	kHz
I002	t_{pLine}	Permissible Line Delay			unlimited	
I003	Δt_{pL}	Permissible Propagation Delay Jitter	Δ tpL = max(|tpLine - tpLx|); x= 1 ... n		12.5	% T_{MAS}
I004	T_{tos1}	Adaptive BiSS Timeout		1.5xT_{MAS} - 2xT_{Smp}	1.5xT_{MAS} + 2xT_{Smp}	
I005	T_{tos2}	Static BiSS Timeout		12.5	40	µs

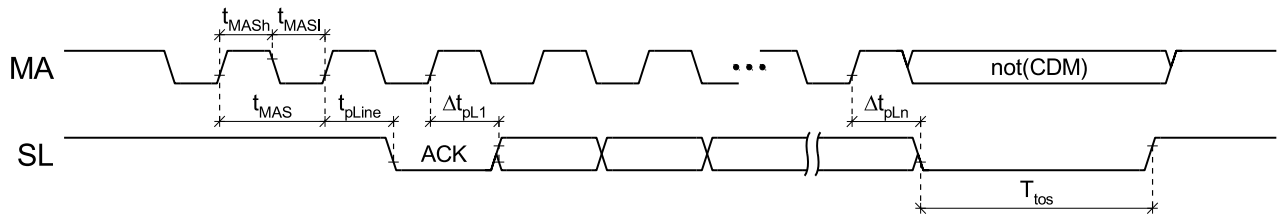


Figure 2: Timing Diagram BiSS Frame

OPERATING REQUIREMENTS: Host Interface, SPI mode

Operating Conditions: VDD = VDDA = 3.0 ... 5.5 V, Tj = -40 ... 125 °C;
Lo Input Level = 0 V, Hi Input Level = VDD, Lo Output Level = 0 V, Hi Output Level = VDD
Capacitive Load at MISO = 20pF

Item No.	Symbol	Parameter	Conditions	Min.	Max.	Unit
I101	t_{C1}	Permissible Cycle Time		100		ns
I102	t_{W1}	Wait Time: NCS lo → hi to NCS hi → lo		100		ns
I103	t_{S1}	Setup Time: NCS lo before SCLK lo → hi		50		ns
I104	t_{P1}	Propagation Delay: MISO stable after NCS hi → lo			50	ns
I105	t_{P2}	Propagation Delay: MISO hiZ after NCS lo→ hi			50	ns
I106	t_{H1}	Hold Time: NCS lo after SCLK lo → hi		50		ns
I107	t_{S2}	Setup Time: MOSI stable before SCLK lo → hi		10		ns
I108	t_{H2}	Hold Time: MOSI stable after SCLK lo → hi		10		ns
I109	t_{P4}	Propagation Delay: MISO stable after SCLK hi → lo			47	ns

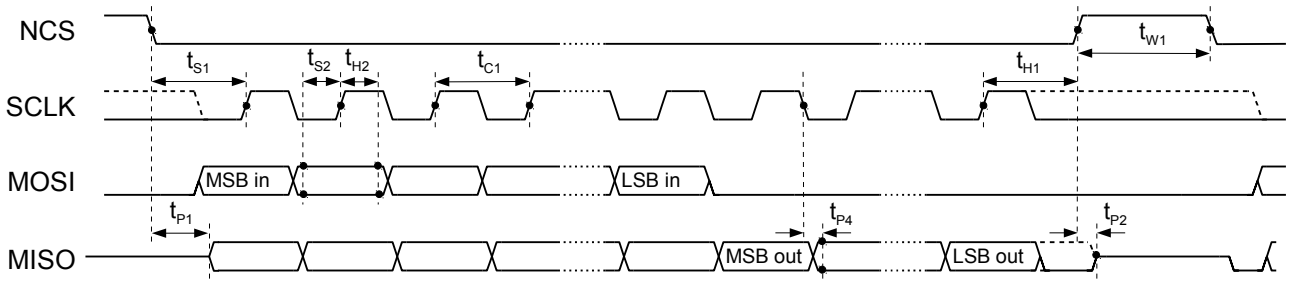


Figure 3: SPI Access

ELECTRICAL CONNECTIONS

Power Supply

To put the iC-MCW into operation, all that is needed is a power supply of 3.3 V up to 5 V. Two 100 nF external buffer capacitors between VDD/VDDA and GND/GNDA and one 1 nF external buffer capacitor between VDD and GND stabilize the input voltages of the iC-MCW. Since the analog and digital power supplies are separate, VDD and VDDA as well as GND and GNDA must be at the same voltage level. Figure 4 illustrates the connection of the iC-MCW for a typical use case.

System Clock

Due to its internal Phase Locked Loop (PLL)-driven oscillator, the iC-MCW does not need any external clock source. The high precision sample clock is generated independently. In case of problems with the sample clock generation, an error is reported to the chip status.

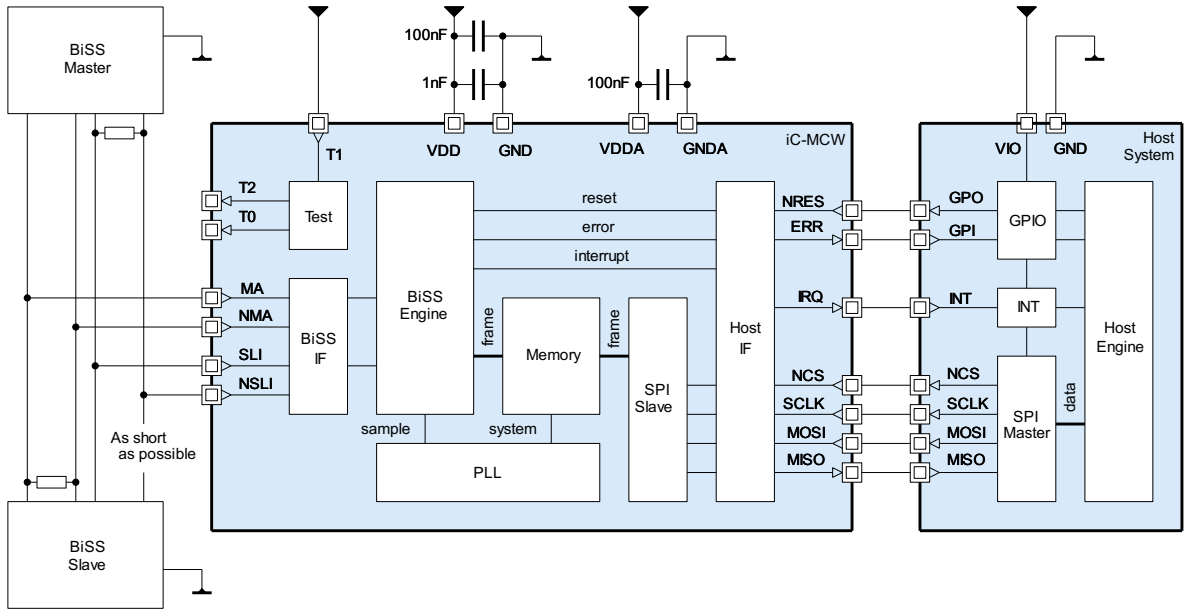


Figure 4: Connections of the iC-MCW

Digital Input

All digital input pins have internal pull-up circuits to provide a default chip state when floating. It is recommended to support unused digital input pins by directly connecting the external power supply voltage. Table 6 shows the the essential digital input pins of the iC-MCW:

Name	Pin	Function
T1	16	Test Enable
NRES	17	Chip Reset
NCS	21	SPI Chip Select

Table 6: Essential Input Pins

BiSS Interface

Since RS422 receivers are included in the BiSS interface of the iC-MCW, the differential BiSS clock line and data line can be connected to the corresponding pins MA/NMA and SLI/NSLI directly.
The iC-MCW is a pure BiSS listener and can only be used to extend a complete BiSS communication system. The BiSS communication system consists of one BiSS master and one or more BiSS slaves connected via RS422 standard. As shown in Figure 4, the RS422 standard suggests suitable termination resistors to allow high frequency data transmission. The additional connection to the iC-MCW does not need any external resistors for signal termination. However, the connection between the BiSS transmission line and the iC-MCW must be designed to be as short as possible to avoid signal reflections.
The BiSS interface follows the electrical specification of the RS422 standard.

Host Interface

A common 4-wire Serial Peripheral Interface (SPI) offers communications between the iC-MCW and a host system, e.g. a microcontroller. The iC-MCW supports SPI transmissions with data rates up to 10 MHz, if common high frequency connection concepts are being considered.
For use in interrupt-driven I/O systems, an interrupt request pin is available. Whenever new BiSS frame data

is available, the IRQ pin produces an interrupt impulse, that can be observed by the host system.
The error pin can be connected to a general input line of the host system to quickly evaluate the error status of the iC-MCW without triggering an additional SPI transfer.
A simple general purpose output connection to the external reset pin NRES allows the host system to reset the iC-MCW to its defined start state whenever necessary.
The digital I/O voltage levels of the host interface are defined by VDD and GND. Except for the BiSS interface, the whole system must be connected to the same power supply voltage.

START-UP

Power On Reset

After connecting power supply to the iC-MCW, the internal Power On Reset (POR) puts the chip into its defined start state. The reset signal is removed as soon as the required power supply voltage is reached and the PLL delivers a stable system clock.
After system restart or an external reset signal, an error bit of the chip status is set. Since the error bits must be reset manually via SPI, an unintentional chip reset will always be detected. This feature can also be used to monitor the power supply.
More information about the chip status can be found in section "MODE OF OPERATION" on page 11.

BiSS Input

The sampling registers of the BiSS lines are reset to digital high. If there is no MA line connected to the chip, the RS422 receiver generates a digital low to the BiSS sample unit. Since a falling edge on MA signals a new incoming BiSS frame, an invalid connection is recognized as a communication error at start-up. In that case, an error bit in the chip status and the external error pin ERR are set. Hence, to test the connection of the BiSS interface, a power cycle or an external reset is necessary.

BiSS Frame Counter

The BiSS frame counter is an important safety feature of the iC-MCW. The sign-of-life counter within the BiSS protocol ensures data consistency at the host system. If the process cycle time of the host system does not match the frame cycle time of the BiSS system, the internal frame counter can be used to count lost BiSS frames.
At start-up or after an external reset signal, the frame counter is reset to zero.

Chip Configuration

The iC-MCW is a robust device with a single data path. For safety reasons, there is no internal chip configuration to alter the behavior of the iC-MCW. Any setup concerning the BiSS protocol is exclusively done within the host system.

MODE OF OPERATION

Once activated, the iC-MCW constantly observes the BiSS communication lines. The transitions of the internal logical states of the iC-MCW are triggered by specific events on the lines MA or SLI.
There are eight logical states the iC-MCW cycles through during BiSS frame processing. Each state handles the signaling of a specific stage of the BiSS communication. Table 7 lists all implemented states and describes the task of the hardware for each state.
The logical states can be mapped to the different frame sections of a BiSS C frame. Figure 5 illustrates the mapping of the logical states of the iC-MCW to the BiSS C frame. The logical states are shown in red on top of the MA clock signal.

State	Task
IDLE	Wait for Frame Start
ABTO	Measurement of the Adaptive BiSS Timeout
DELAY	Measurement of the Line Delay
READY	Wait for Clock Cycle
START	Waiting for Start Bit
CDS	Sampling of CDS Bit
DATA	Sampling of Sensor Data
ERROR	Wait for BiSS Timeout

Table 7: System States of the iC-MCW

BiSS Communication Start
As long as the connected BiSS clock line MA is digital high, the iC-MCW stays in state *IDLE* waiting for a new BiSS communication to start. A timer in the host system can be used to check idle times between two BiSS frames.
A new BiSS communication starts with a falling edge on BiSS clock line MA. As shown in Figure 5 SLI is supposed to be digital high at this point. If SLI is digital low at the start of the communication, the BiSS frame will be reported as erroneous and an error bit of the chip status is set. However, the BiSS frame is still processed until the BiSS timeout is detected.
The error bit is routed to the external error pin ERR.

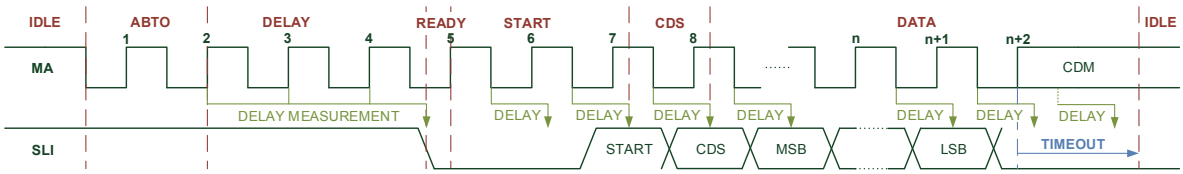


Figure 5: BiSS Frame

Static BiSS Timeout
Before the adaptive BiSS timeout (ABTO) is measured, the static BiSS timeout (BTO) is used to determine the end of a BiSS frame. The iC-MCW observes the BiSS clock line MA and starts the BTO counter as soon as a new BiSS communication begins. The timeout counter is reset for every change of the voltage level at MA. If a BTO is detected, the iC-MCW sets an error bit of the chip status, generates an interrupt request and goes back to the state *IDLE*.

Adaptive BiSS Timeout
The ABTO is measured during the first 1.5 MA clock periods. If the second rising edge at MA has been detected, the iC-MCW recognizes a BiSS timeout whenever there are no further changes within that measured period of time. It can be important to extend the first 1.5 MA clock periods, if a SPI master interface is used to emulate the BiSS C protocol for example. The maximum time for the ABTO is the BTO.
Depending on the logical state the iC-MCW is operating in, a BiSS timeout might trigger an error. According to the BiSS C protocol, a BiSS timeout occurring in the logical states *ABTO*, *START* and *CDS* is not allowed and will be reported to the chip status.

BiSS Line Delay
The iC-MCW is capable of compensating line delays between the rising edge of MA triggered by the BiSS master and the corresponding response of the BiSS slave on the data line SLI. The line delay measurement starts at the second rising edge of MA. If SLI is already digital low at the beginning of the line delay measurement, the iC-MCW continues to the logical state *ERROR* and sets an error bit to report an invalid BiSS protocol.
Since the delay is measured on a clock cycle by clock cycle basis, there is no limit to the line delay measurement. **If there is no response by the BiSS slave, only the BiSS master is responsible to stop generating MA clock cycles.**
Figure 5 shows the line delay measurement in between MA and SLI colored in green. Once measured, the line delay is used for determining the optimum sample point of data bits received on the line SLI.

BiSS Conversion Time
After measuring the line delay, the iC-MCW waits for the next clock cycle of the BiSS frame to begin. At this point, the optimum sample point of the BiSS data line can be used to sample the frame data. The logical state is *READY*.
Before any data is transmitted, the BiSS slave sends the start bit to inform the BiSS master that the requested data is available. The start bit can be delayed by the slave if a conversion time is needed to provide the data. The iC-MCW supports the conversion time by resting in the state *START* until the start bit is recognized. **The conversion time of the iC-MCW is unlimited as well and has to be managed by the BiSS master.**

BiSS Control Data
BiSS Control Data bits are sampled by the iC-MCW in both directions. The CDS bit is sampled after the start bit has been detected successfully. The NCDM bit on the other hand is read at the end of a BiSS frame when the timeout occurs. The BiSS Control Data is not concatenated by the iC-MCW and must be loaded for each BiSS frame separately. The CDS and NCDM bits of the last BiSS frame can be read via SPI.
More information about the BiSS Control Data can be found in section "HOST INTERFACE" on page 15 and in section "REGISTER LAYOUT" on page 19.

BiSS Frame Length
Since the iC-MCW does not need to be configured to any specific BiSS frame data length, the chip will accept BiSS frames of any data lengths until the memory

limit is reached. That means BiSS frames with a data length of 0. . . 250 bits will be accepted without any error triggered.
In general the iC-MCW supports the three classes of BiSS frames that are shown in Table 8.
The iC-MCW samples all bits of the BiSS frame until an ABTO is recognized. It is possible, that the iC-MCW samples more bits than are generated by the BiSS slave. That is because the BiSS master toggles the MA line until the number of configured data bits is processed. The additionally sampled bits do not carry any information and must be omitted by the host system. The number of sampled bits is stored internally and are accessible via SPI.

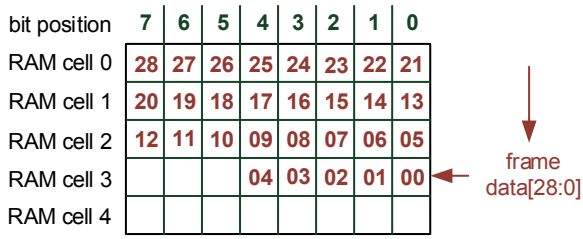


Figure 6: Example Frame in RAM

The frame data is written to the internal random access memory (RAM). Since each RAM cell contains one byte of data, the BiSS frame is divided into 8 bit chunks. The BiSS frame, which is sent MSB-first, is shifted into the LSB of each memory cell. Figure 6 illustrates an example frame of 29 bits written into memory.

BiSS Frame	Description
Reduced Frame	The reduced frame can be used to send broadcast commands to unconfigured BiSS slaves. The frame is completed when the adaptive BiSS timeout is detected in the logical state DELAY or READY.
Short Frame	A short frame is a BiSS frame with a data length of zero. Only CDS and NCDM bits are exchanged between master and slave. It is used for fast register communication during configuration.
Long Frame	A long frame is a BiSS frame that is configured to transmit sensor data of a specific length within each frame cycle. Simultaneously, the CDS and NCDM bits are transmitted to provide control data communications. This is the standard BiSS operation.

Table 8: Classes of BiSS Frames

The last byte of each frame may be incompletely filled. The host system must be configured to the expected frame length to correctly mask the last byte. More information about masking the last byte can be found in section "HOST INTERFACE" on page 15.

Host System
There are two ways to inform the host system about new BiSS frame data. First, the iC-MCW generates an interrupt request every time a BiSS timeout occurs and the memory bank can be switched to make the frame data accessible. The interrupt pin IRQ is held digital high for a specific time or until a SPI transfer has been started by the host system. The duration of the IRQ pulse restricts the minimum BiSS frame period of the communication system.

Second, it is possible to poll the chip status and observe the bit *DETECT* that is set every time a BiSS timeout occurs. Since the Frame Counter (*FCNT*) is also incremented every time a BiSS timeout occurs, the bit *DETECT* is digital high if *FCNT* is not zero. However, the bit *DETECT* only reports a complete BiSS frame detected by the iC-MCW. Due to memory bank access collisions, the frame data is only available if the bit *AVAIL* is set as well. Thus, the bit *AVAIL* matches the activation of an interrupt request.
Figure 7 shows the status bits above during execution of a chip communication sequence that is detailed described in subsection "Memory Banks". The *FCNT* and the bits *AVAIL* as well as *DETECT* are reset every time the frame data of the iC-MCW is accessed via SPI.

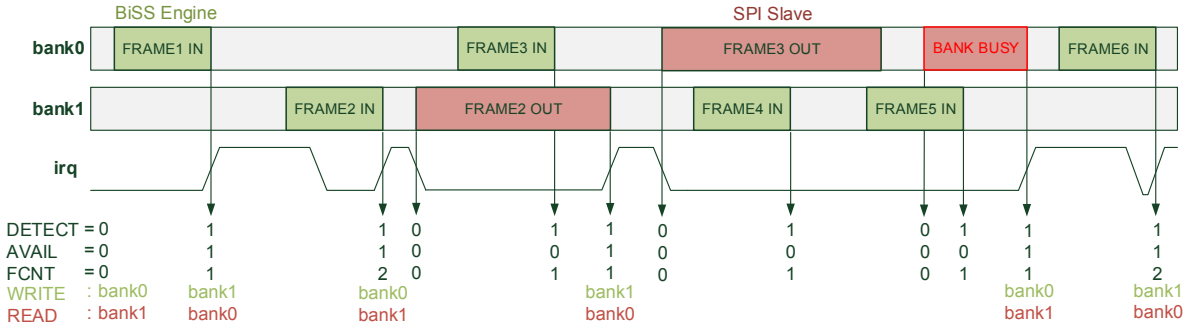


Figure 7: Memory Bank Access

Memory Banks

The frame data memory of the iC-MCW is organized into two independent banks. Each frame bank can hold 250 frame bits which correspond to three fully configured BiSS frames with 16-bit CRC protection for each channel.

Figure 7 shows the bank selection, the interrupt request generation and the corresponding status bits during an example frame write and read sequence. The incoming BiSS frames, that are written into the memory of the iC-MCW are colored in green, the read of the host via SPI is colored in red.

If there is no SPI transmission in progress, new frame data is written to the memory banks alternately. The bank selection for read access is accordingly set to the memory bank that holds the new frame data. The *FCNT* is incremented whenever a new frame transmission is completed by the BiSS timeout. The bits *DETECT* and *AVAIL* report new frame data that can be read via SPI. The interrupt request is directly set and can be used to implement interrupt-driven I/O applications.

During a frame data transmission via SPI, incoming BiSS frame data is written into the second memory bank simultaneously. If the SPI command is not completed at the end of the incoming BiSS frame, the bit *DETECT* is set and the *FCNT* is incremented but the bit *AVAIL* and the interrupt request is delayed until the end of the SPI command.

Since the switch of memory banks is also delayed, another incoming BiSS frame overwrites the data of the previous BiSS frame as long as the iC-MCW is busy accessing frame data via SPI. Thus, the iC-MCW allows the access of one BiSS frame until another incoming BiSS frame begins.

However, lost BiSS frames are captured by the internal frame counter that is attached as a frame number to the SPI access supporting the safety related sign-of-life counter even for slower host systems.

If the read and write access overlap, there is no interrupt request generated and the SPI command reports BANK BUSY to mark the received data as invalid.

More information about the SPI commands can be found in section "HOST INTERFACE" on page 15.

BiSS Frame Data

The iC-MCW is a BiSS to SPI bridge only and does not analyze any BiSS frame data. The extraction of the sensor data and data consistency checks, like CRC calculation, is intended to be done by the host system. Hence, the host system must exactly be configured just as the connected BiSS slave to interpret the BiSS frame data of the iC-MCW correctly.

HOST INTERFACE

The host interface of the iC-MCW consists of the NRES pin, the IRQ pin, the ERR pin and the 4 SPI pins. It is suitable for applications that use interrupt-driven I/O or chip status polling.

NRES Pin

The pin NRES is intended to set the iC-MCW back to its defined start state. It provides the functionality to test if the BiSS lines are properly connected to the iC-MCW. However, because of the internal POR the external NRES pin can be left unconnected.

IRQ Pin

An interrupt request is generated every time a new BiSS timeout is detected and the memory bank is ready to switch the address selection for read access. It does not matter what kind of BiSS frame is transmitted or if there are any errors during communications, the IRQ pin indicates new data is available.
After a BiSS timeout, the interrupt request is digital high for a specific time only (see ELECTRICAL CHARACTERISTICS A10). Every SPI transmission acknowledges the current interrupt request by resetting the interrupt pin even before the specific time runs out.
The pin IRQ is intended to be used in systems that support interrupt-driven I/O.

ERR Pin

The pin ERR is generated by a logical OR gate connected to all error bits of the register DeviceError. For interrupt-driven I/O systems, the pin ERR can be used to observe the register DeviceError without sending any SPI command. Only if the pin ERR is set, an SPI command is needed to get detailed information about the error occurred.
The error bits and the error pin can only be reset by sending the SPI command WriteRegister to the register DeviceError. Detailed information about the register DeviceError can be found in section "REGISTER LAYOUT" on page 19.

SPI Mode

The iC-MCW provides a SPI slave to communicate with the host system. The SPI supports two modes of the Motorola standard.
The iC-MCW samples MOSI on rising edges of SCLK and generates data on falling edges of SCLK. The corresponding Motorola modes that can be used by the SPI master of the host system are:

CPOL = 0, CPHA = 0
CPOL = 1, CPHA = 1

The data lengths of opcodes, addresses and registers are 8-bit. All data is sent MSB-first. Continuous register access and BiSS frame data are transmitted serially in successive byte chunks.
The chip select line NCS is low active. It starts a new SPI transmission on the falling edge and cancels any SPI transmission on the rising edge.
The idle state of SCLK is not determined.

SPI Opcodes

Each SPI transmission starts with an opcode sent from the host system to the SPI slave via MOSI. Each opcode triggers a command executed by the iC-MCW.
Table 9 lists the four available opcodes and the corresponding commands:

Opcode	Command
0x9C	ReadStatus
0xA6	ReadFrameData
0x8A	ReadRegister
0xD2	WriteRegister

Table 9: SPI Opcodes

Invalid SPI opcodes trigger an error that is reported to the error pin. The SPI error bit must be reset manually with the SPI command WriteRegister.

ReadStatus

The SPI command *ReadStatus* triggers the iC-MCW to latch the StatusByte that can be clocked out of MOSI. The first byte of MOSI is the opcode of the command and the second byte of MISO shows the latched status information. An example of the *ReadStatus* transmission can be found in Figure 8.

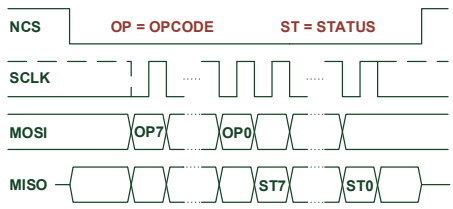


Figure 8: SPI Transmission ReadStatus

The command *ReadStatus* is intended for applications that do not use interrupt- driven I/O to manage the communication. The StatusByte is composed of several

bits from the registers DeviceStatus and DeviceError. More information about the registers of the iC-MCW can be found in section "REGISTER LAYOUT" on page 19.
Table 10 lists the bits of the StatusByte.

Bit	Name	value
7	AVAIL	0 : No Frame Data Available 1 : New Frame Data Available
6	DETECT	0 : No BiSS Frame Detected 1 : New BiSS Frame(s) Detected
5-4	WAIT	00 : Wait for Begin of Frame 01 : Wait for Delay Acknowledge 10 : Wait for Start Bit 11 : Wait for BiSS Timeout
3	PLL	0 : PLL Lock OK 1 : PLL Lock Warning
2	SPI	0 : SPI Communication OK 1 : SPI Communication Error
1	FRAME	0 : Frame Data OK 1 : Frame Error Detected
0	RESET	0 : No Chip Reset 1 : Chip Reset Detected

Table 10: ReadStatus Bits

AVAIL and DETECT:
These bits are used to inform the host about new frame data for communication systems that use status polling instead of interrupt requests. Especially the bit *AVAIL* should be used to control the request of frame data from the iC-MCW.

WAIT:
The bits *WAIT* show the wait state of the iC-MCW. According to Table 10 the bits can be used to monitor what signal the iC-MCW is waiting for. This information can be used to debug the communication if the chip waits for the start bit an exceptional period of time for example.
WAIT can be mapped to the internal logical states as shown in Table 11. More information about the internal logical states can be found in section "MODE OF OPERATION" on page 11.

PLL, SPI, FRAME and RESET:
PLL, *FRAME* and *RESET* correspond to the error bits of the register DeviceError directly.
The bit *SPI* combines the error bits BYTE, OPCODE, READ and WRITE from the register DeviceError. The bit is set if one of the error bits is set by the SPI engine. Detailed information about the register DeviceError can be found in section "REGISTER LAYOUT" on page 19.

Logical State	Wait for
IDLE	Begin of Frame
ABTO	BiSS Timeout
DELAY	Delay Acknowledge
READY	BiSS Timeout
START	Start Bit
CDS	BiSS Timeout
DATA	BiSS Timeout
ERROR	BiSS Timeout

Table 11: Wait States of the iC-MCW

ReadFrameData
In general the command *ReadFrameData* provides a continuous stream of bytes of the last BiSS frame. The command should only be used if an interrupt request is detected by the host system or the bit *AVAIL* of the StatusByte is set.
After receiving the opcode of the command in the first byte on pin MOSI, the iC-MCW returns the byte *INFO* that reports the status of the current BiSS frame in the second byte on pin MISO. There are several codes of *INFO* that indicate what data is provided in the following SPI bytes. Table 12 lists the possible output of *INFO* and what it means.

INFO	Code	Description
NO_FRAME	0xFF	No BiSS Frame in Memory,
BANK_BUSY	0xFE	Memory Bank Busy, Data Invalid
FRAME_ERROR	0xFD	BiSS Frame Error, See Error Code
NO_CDS	0xFC	Reduced BiSS Frame Detected, NCDM Only
LENGTH_ERROR	0xFB	Incomplete BiSS Frame Data, 250 Bits Available Only
FRAME_LENGTH	0xFA - 0x01	Valid BiSS Frame, Data Ready to Transmit
NO_DATA	0x00	Valid BiSS Frame, CDS and NCDM Only

Table 12: INFO Byte of ReadFrameData

NO_FRAME:
If *INFO* equals *NO_FRAME*, there is no new frame detected since the last *ReadFrameData* SPI Command. Hence, there is no frame data to be read out of the memory of the iC-MCW.

BANK_BUSY:
If the current read bank has already been read out and a new incoming frame is still in progress, the iC-MCW did not have the chance to switch the memory bank. Thus, the bank is considered to be busy and *INFO* equals *BANK_BUSY* to report an invalid data access. Hence, there is no frame data to be read out of the memory of the iC-MCW.
It is recommended to only send a *ReadFrameData* command if an interrupt request signal is detected or the bit *AVAIL* is set.

FRAME_ERROR:
FRAME_ERROR indicates a BiSS error during frame processing. The register FrameError is sent on MISO in the third byte and can be used to debug the communication. FrameError consists of the bits *CODE[1:0]* and WARN. The bits *CODE[1:0]* represents the type of the BiSS error and the bit WARN is set if the BiSS data line SLI was digital low at the beginning of the current BiSS frame. Table 13 lists the implemented error types and the corresponding *CODE[2:0]*.

Error Type	CODE[2:0]	Description
ABTO	00_b	Timeout During Measurement of Adaptive BiSS Timeout
START	01_b	Timeout During Start Bit Execution
CDS	10_b	Timeout During CDS Execution
LEVEL	11_b	SLI-Level Error During Line Delay Measurement

Table 13: BiSS Frame Error Code

More information about the register FrameError can be found in section "REGISTER LAYOUT" on page 19.

NO_CDS:
The reduced BiSS frame only sends the NCDM bit from the BiSS master to the BiSS slave. Since there is no sensor data transmitted, there are no further frame data bytes to be clocked out of pin MISO.

LENGTH_ERROR:
The iC-MCW is capable of capturing BiSS frames of up to 250 bits including CRC protection. If the BiSS engine detects a valid communication of 251 bits and above, the byte *INFO* shows *LENGTH_ERROR* and the following 32 bytes can be clocked out to transmit 250 bits of the processed BiSS frame. However, the BiSS communication is not completely available and must be handled as a communication error.

FRAME_LENGTH:
FRAME_LENGTH shows the valid frame length of the current BiSS frame in memory. The value of *INFO* must be used to calculate the number of bytes that need to be clocked out of MISO to completely transmit the frame data available.
Since the frame data is shifted from the LSB to the MSB of each memory cell and the number of sampled bits does not naturally match the number of configured frame bits, the last byte must possibly be masked to accurately extract the data bits. The difference of the expected BiSS frame length and the sampled frame length reported by *FRAME_LENGTH* can be used to identify the actual BiSS frame bits of the last byte. Figure 9 shows the position of the frame bits within the last data byte to help calculating the valid frame data.

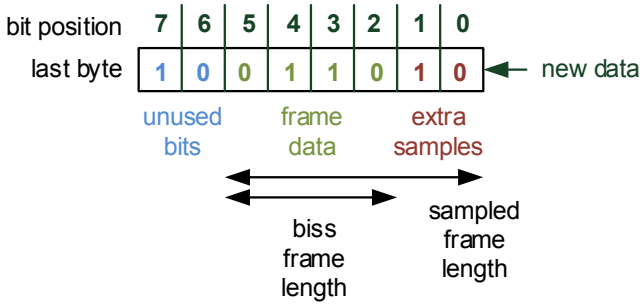


Figure 9: Last Data Byte

NO_DATA:
A short BiSS frame does not transmit any sensor data to the slave. If the iC-MCW detects a BiSS timeout directly after the CDS bit has been sampled, the byte *INFO* is set to *NO_DATA*. There are no further data bytes that need to be clocked out of pin MISO.

At the end of the SPI command *ReadFrameData*, there are two optional bytes that contain the control data bits CDS and NCDM as well as the number of the current frame the host read the data of. The CDS bit is sent at bit seven of the second to last byte followed by the NCDM bit at the sixth bit. The 14-bit frame number is sent MSB first and starts right after the NCDM bit. Figure 10 illustrates the sequence of a *ReadFrameData* command with *INFO = FRAME LENGTH* and transmission of the optional control data and frame number.

Since the iC-MCW counts erroneous BiSS frames as well, the frame number is also attached to frame with *INFO = FRAME_ERROR*. Especially *NO_CDS* and *NO_DATA* frames need to clock out the byte that contains the CDS and NCDM bit. Only *NO_FRAME* and *BANK_BUSY* does not require any frame number.

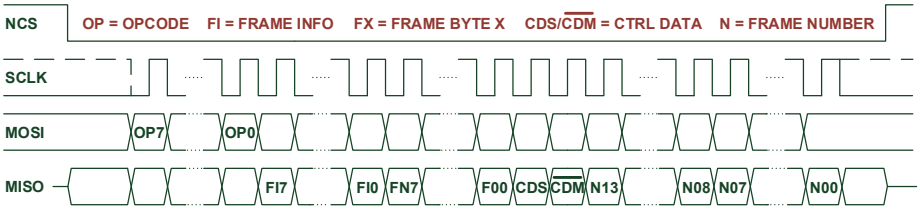


Figure 10: SPI Transmission ReadFrameData

ReadRegister

The command *ReadRegister* is used to read an address of the register bank shown in Table 15 of section "REGISTER LAYOUT" on page 19.

The first byte of MOSI is the opcode of the command and the second byte is the desired register address. While transmitting the register address, the pin MISO clocks out the StatusByte. The third byte contains the data of the register that was selected by the address byte.

Additional bytes activate the continuous read function. The iC-MCW increments the register address automatically and multiple registers can be read with one single SPI transmission.

Invalid register addresses trigger a SPI error that is reported to the error pin. The SPI error bit must be reset manually.

Figure 11 shows an example of the *ReadRegister* transmission.

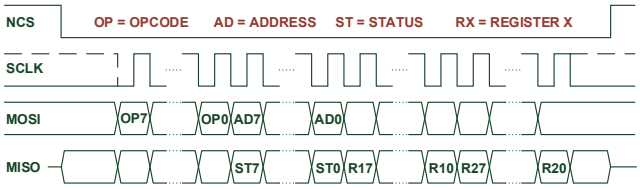


Figure 11: SPI Transmission ReadRegister

WriteRegister

The command *WriteRegister* is used to write an address of the register bank shown in Table 15 of section "REGISTER LAYOUT" on page 19.

The first byte of MOSI is the opcode of the command and the second byte is the desired register address. While transmitting the register address, the pin MISO clocks out the StatusByte. The third byte contains the data to be written to the register that was selected by the address byte.

In contrast to the command *ReadRegister*, there is no continous write function implemented.

Since the iC-MCW does not need any configuration, the command *WriteRegister* is exclusively used to reset the error bits in the register DeviceError. If the SPI engine recognizes a write access to the register DeviceError, the error bits are reset no matter what the write data of the SPI command *WriteRegister* is.

Invalid register addresses and register addresses that are read only trigger a SPI error that is reported to the error pin. The SPI error bit must be reset manually.

Figure 11 shows an example of the *WriteRegister* transmission.

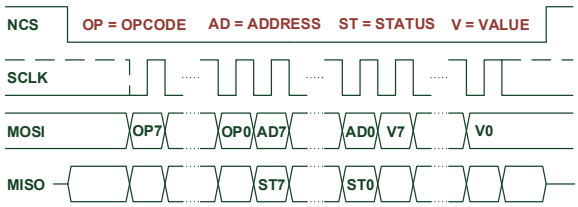


Figure 12: SPI Transmission WriteRegister

REGISTER LAYOUT

The register bank of the iC-MCW can be accessed via SPI command ReadRegister and WriteRegister. Table 15 gives an overview of the available chip addresses and data.
It is possible to read multiple register addresses successively by continuing to generate SPI clocks after the data of the first address has been clocked out.
Invalid read addresses trigger an error of the register *DeviceError*.

Revision	Value
MCW_0	0xA1
MCW_1	0xA2
MCW_Z	0xB1
MCW_Z1	0xB2

Table 14: ChipRevision Values

ChipRevision
The register *ChipRevision* can be used to test the status of the iC-MCW and the SPI communication system. Table 14 shows the values of the register *ChipRevision* for different chip versions.

OVERVIEW								
Addr	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ChipRevision (R)								
0x00	CHIP_REV[7:0]							
DeviceStatus (R)								
0x01	RESERVED			AVAIL	DETECT	STATE[2:0]		
DeviceError (R/W)								
0x02	RESERVED	PLL	BYTE	OPCODE	READ	WRITE	FRAME	RST
FrameInformation (R)								
0x03	INFO[7:0]							
FrameNumberCD (R)								
0x04	FNUM_LSB[7:0]							
0x05	CDS	NCDM	FNUM_MSB[13:8]					
FrameError (R)								
0x06	RESERVED					CODE[1:0]		WARN
FrameCounter (R)								
0x07	FCNT_LSB[7:0]							
0x08	RESERVED		FCNT_MSB[13:8]					

Table 15: Register layout

DeviceStatus
The bit *AVAIL* reports new frame data ready to be transmitted to the host. The bit *DETECT* is set as soon as a BiSS timeout is detected by the iC-MCW. *DETECT* is cleared together with the frame counter if the SPI command ReadFrameData is successfully executed. If the bit *DETECT* is set but *AVAIL* equals zero, the currently activated RAM bank is busy and cannot be addressed to access the received frame data.

The bits *STATE(2:0)* show the state the iC-MCW currently operates in. Table 16 shows the codes of the logical states used for frame processing. More information about the logical states of the iC-MCW can be found in section MODE OF OPERATION on page 11.

State	Code	Task
IDLE	000_b	Wait for Frame Start
ABTO	001_b	Measurement of the Adaptive BiSS Timeout
DELAY	010_b	Measurement of the Line Delay
READY	011_b	Wait for Clock Cycle
START	100_b	Waiting for Start Bit
CDS	101_b	Sampling of CDS Bit
DATA	110_b	Sampling of Sensor Data
ERROR	111_b	Wait for BiSS Timeout

Table 16: System States of the iC-MCW

DeviceError
The register *DeviceError* shows several error bits that are set if any erroneous signaling is detected by the iC-MCW. The error bits are persistent and must be reset manually by sending the SPI command WriteRegister to this register.
Table 17 describes the different error bits in detail.

Error Bit	Description
PLL	Internal PLL Lost Lock Signal
BYTE	ReadFrameData Byte Overflow
OPCODE	Invalid SPI Opcode
READ	Invalid SPI Read Address
WRITE	Invalid SPI Write Address
FRAME	BiSS Protocol Error Detected
RST	Reset of the iC-MCW Executed

Table 17: Implemented Error Bits

The error bits *PLL*, *OPCODE*, *READ*, *WRITE*, *FRAME* and *RST* are connected to the OR-gate of the error pin of iC-MCW to externally inform the host system about any unusual behaviour.

FrameInformation
The register *Frame Information* matches the section INFO of the SPI command ReadFrameData that holds the number of bytes of the current BiSS frame. If any frame error occurred, INFO shows the code of the error for debugging. Detailed information about the INFO coding can be found in section "HOST INTERFACE" on page 15.

FrameNumberCD
The register *FrameNumberCD* matches the frame information transmitted in the last two bytes of the SPI command ReadFrameData. It holds the current index of the frame that can be used for verifying the time-of-life-counter of the BiSS frame and the *CDS* and *NCDM* bits of the BiSS C Control Data communication. Detailed information about the frame number and Control Data can be found in section "HOST INTERFACE" on page 15.

FrameError
The register *FrameError* matches the error code that is transmitted by the iC-MCW if the INFO section of the SPI command ReadFrameData reports an error during frame execution. Detailed information about the error code can be found in section "HOST INTERFACE" on page 15.

FrameCounter
The register *FrameCounter* shows the number of frames detected by the iC-MCW since the last SPI command ReadFrameData. The frame counter value is incremented whenever a new BiSS frame timeout is detected. The 14-bit value is capable of counting up to 16382 BiSS frames. If the counter reaches 0x3FFF it will not overflow but stop incrementing.

REVISION HISTORY

Rel.	Rel. Date*	Chapter	Modification	Page
A1	2018-01-18		Initial Release	

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

ORDERING INFORMATION

Type	Package	Order Designation
iC-MCW	24-pin QFN24, 4 mm x 4 mm, thickness 0.9 mm, RoHS compliant	iC-MCW QFN24-4x4
Evaluation Board	80 mm x 100 mm eval board	iC-MCW EVAL MCW1D

Please send your purchase orders to our order handling team:

Fax: +49 (0) 61 35 - 92 92 - 692
E-Mail: dispo@ichaus.com

For technical support, information about prices and terms of delivery please contact:

iC-Haus GmbH
Am Kuemmerling 18
D-55294 Bodenheim
GERMANY

Tel.: +49 (0) 61 35 - 92 92 - 0
Fax: +49 (0) 61 35 - 92 92 - 192
Web: http://www.ichaus.com
E-Mail: sales@ichaus.com

Appointed local distributors: http://www.ichaus.com/sales_partners