SafeSPI - Serial Peripheral Interface for Automotive Safety

1 Introduction

INFO_001 The serial peripheral interface (SPI) is a synchronous serial communication interface used for short distance communication, usually between devices on a printed board assembly. The interface was developed by Motorola and is now a de-facto standard for several automotive applications.

Because there is no formal SPI standard, a wide variety of protocol options exist. This flexibility means every device defines its own protocol, increasing the development effort of new systems, devices and software.

In automotive safety applications, there is often an independent monitoring device (often termed “safing”) which listens to sensor data on the SPI bus. This monitoring device is usually implemented in hardware, and imposes constraints on the SPI protocol.

This specification describes a standard for a target SPI interface used in automotive applications. Its main focus is the transmission of sensor data between different devices.

1.1 Requirement specification types

DEF_002 Each requirement within this specification is marked with a unique identification. The identification consists of a classifier and a unique number. The number is unique over all versions of this specification. The classifiers are the following:

- INFO: The following content has informative character.
- DEF: The following content represents a definition. A definition itself cannot be fulfilled alone. However, other requirements refer to this definition and to fulfil these requirements, this definition must be followed
- REQ: The following content is a requirement to the slaves and the masters
- REQM: The following content applies only to SPI masters
- REQL: The following content applies only to SPI monitor (Listener)
- REQS: The following content applies only to SPI slaves

Headings do not present any kind of requirement.

REQ_003 A device may call itself SafeSPI compatible if it fulfils all requirements (REQ).

REQ_003a A device may call itself SafeSPI In-Frame compatible if it fulfils all In-Frame requirements (REQ).

REQ_003b A device may call itself SafeSPI Out-Of-Frame compatible if it fulfils all Out-Of-Frame requirements (REQ).

1.2 Scope

DEF_004 This SafeSPI standard targets automotive SPI devices. The main focus is sensors, interface integrated circuits (ICs), system application specific ICs (ASICs) and microcontrollers.

INFO_005 Other devices may call themselves “SafeSPI compatible” according to *REQ_003* if wished.

2 Overview about SPI communication

INFO_006 A standard SPI interface consists of 4 ports as shown in Figure 1.

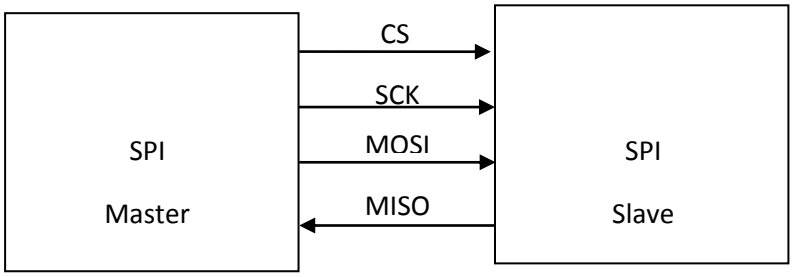

Figure 1 SPI-Interface

The Serial Clock (SCK) represents the master clock signal. This clock determines the speed of data transfer and all receiving and sending is done synchronously to this clock. The Chip Select (CS) activates the SPI interface at the SPI. As long as the CS signal is at high level, the SPI Slave will not accept the SCK signal or the Master-Out-Slave-In input (MOSI), and the Master-In-Slave-Out output (MISO) is in high impedance. When the CS signal is at low level, data can be transferred from the SPI Master to the SPI Slave and vice versa. Commands are transmitted through the MOSI to the SPI Slave and the SPI Slave returns its response through the MISO.

REQL_006a On the monitor device, all signals (CSx, SCK, MOSI, MISO) are inputs only.

INFO_007 SPI bus systems support several slave devices on one bus by using either multiple chip select lines, one for each slave, or by a logical addressing with only one common CS. For several airbag and safety systems a monitor device is connected as listener to the bus. This device is often an ASIC which needs a dedicated SPI format. An example configuration is depicted in Figure 2.

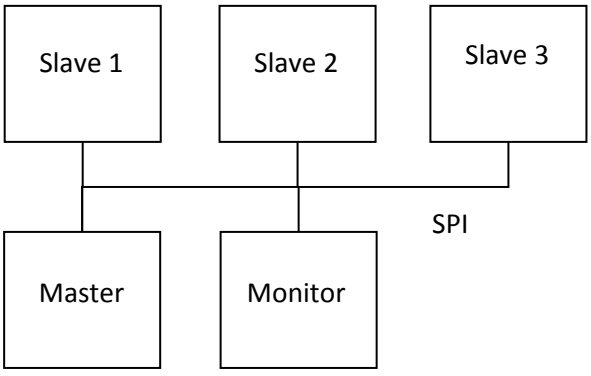

Figure 2 Typical SafeSPI system configuration

INFO_008 The power supplies for each device on the SafeSPI bus are not specified, and can be independent, as shown in the figure below.

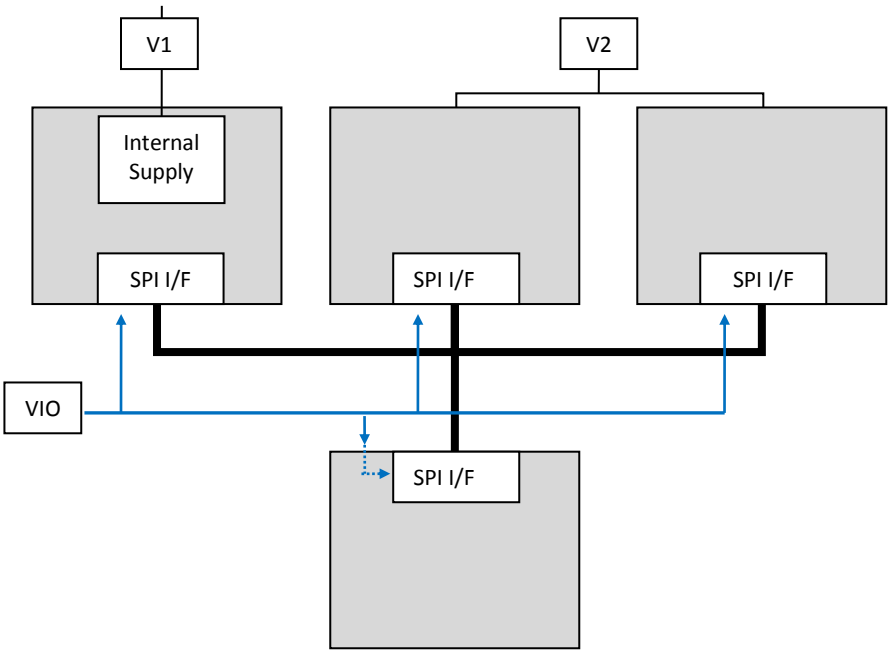

Figure 3 Example of different power domains V1, V2 & V3 with common VIO

3 Physical layer specification

INFO_010 The following chapter describes the physical layer of the SafeSPI specification. Besides voltage and current levels, capacitances of pins, the timings of the different communication lines are described.

3.1 Voltage levels and capacitances

INFO_011 Each of the devices can be powered from an independent supply. However, the SPI interface circuit of each device is supplied by a common VIO.

DEF_012 VIO defines the supply voltage of the SPI interface of the device in order to have common reference for voltage input / output levels.

A supply voltage of a SafeSPI component may have tighter specification.

Positive current flows into the device.

	Parameter	Symbol	Condition	Min	Max	Unit
REQ_013	Mode 3.3V: Supply voltage of SPI interface (to be provided and required respectively at the VIO pin)	V_{IO}		3.0	3.6	V

DEF_013a The following requirements apply to all four communication PINs, namely MISO, MOSI, CS and SCK if not noted otherwise.

ID	**Parameter**	**Symbol**	**Condition**	**min**	**Max**	**Unit**
REQ_014	Input / output capacitance	C_{IO}			6	pF
REQ_015	Total signal load capacitance	C_{LOAD}		6	100	pF
REQ_016	Input low voltage	V_{IL}			0.8	V
REQ_017	Input high voltage	V_{IH}		2.4	V_{IO}	V
REQ_018	Output low voltage	V_{OL}	$I_{LOA\ D}$= 1 mA		0.4	V
REQ_019	Output high voltage	V_{OH}	I_{LOAD} = -1 mA	V_{IO}-0.4	V_{IO}	V
REQ_020	Input voltage hysteresis	V_{HYST}		0.2		V
REQS_021	Output leakage current	I_{LEAK}	MISO only	-10	10	µA
REQS_022	Input pull-up current (1)	I_{PU}	CS only @ V_{CS} 0V .. 2.4V	-70	-20	µA
REQS_022a	Input pull-up resistance (1)	R_{PU}	CS only @ V_{CS} 0V .. 2.4V	60	140	KOhm
REQS_023	Input pull-down current (2)	I_{PD}	MOSI and SCK @ $V_{SI/SCK}$ 0,8 .. V_{IO}	20	70	µA
REQS_023a	Input pull-down resistance (2)	R_{PD}	MOSI and SCK @ $V_{SI/SCK}$ 0,8 .. V_{IO}	60	140	KOhm

Notes:

(1) The pull-up at pin CS can be an active current source or a passive resistor.

(2) The pull-down at pins SCK/MOSI can be an active current source or a passive resistor.

3.2 Timing specifications

INFO_024 The timings are specified to allow an operation of up to at least 10 MHz on the SPI. All timings are valid for the full range of specified voltage levels, input capacitances and current levels. The different parameters are defined in the following graphic.

DEF_025

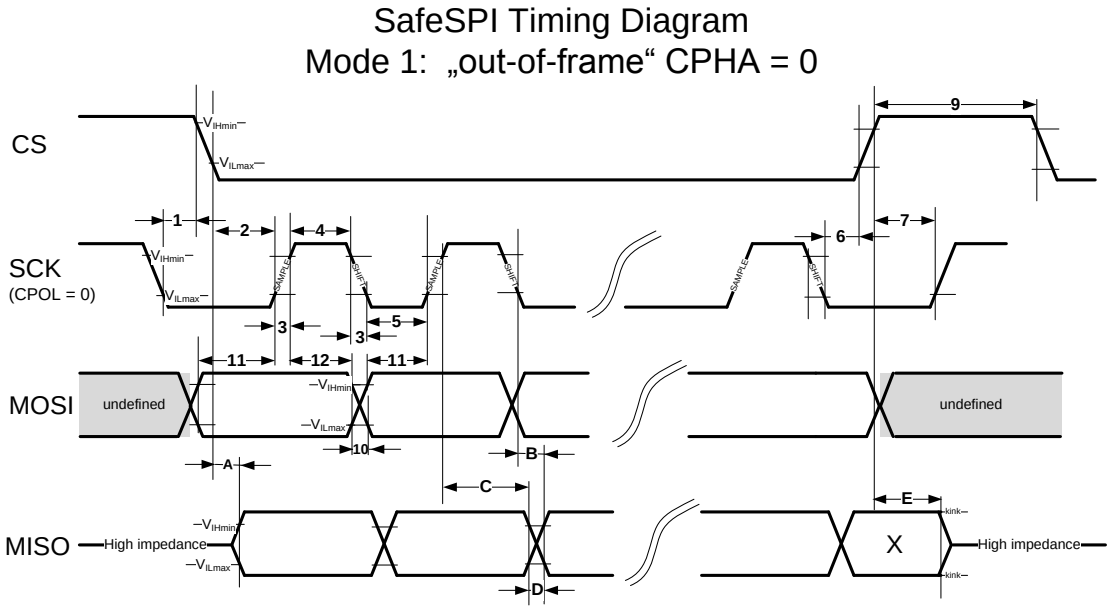

Figure 4 SPI timings for Mode 1: Out-of-frame / CPHA = 0

DEF_025a

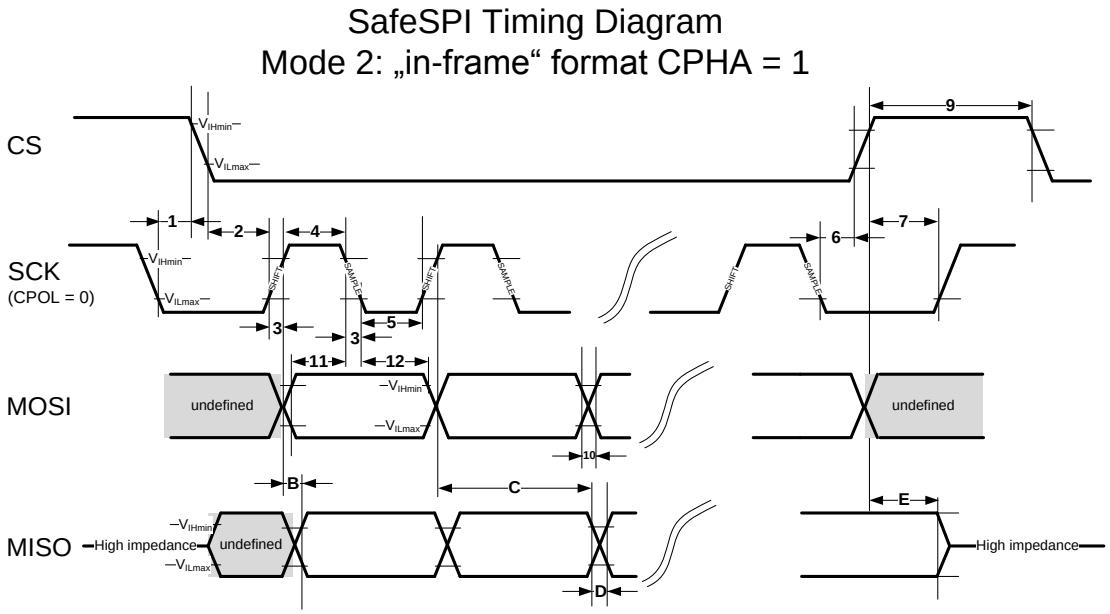

Figure 5 SPI timings for Mode 2: In-frame / CPHA = 1

DEF_026 All timings are specified from V_{IHmin} to V_{ILmax} or vice versa.

DEF_027 All timings are specified over full voltage range of V_{IO}, unless specified otherwise

DEF_028 All timings are specified over full range of bus load C_{LOAD}, unless specified otherwise

DEF_029 All timings are specified over full temperature range, unless specified otherwise

INFO_030 The following requirements are to the slave from master-point of view:

ID	Parameter	Symbol	Condition	Min	Max	Unit
REQS_031	MISO data valid time (CS)	A		*	40	ns
REQS_032	MISO data valid time (SCK) *Medium driver mode*	B	C_{LOAD} = 100pF	*	30	ns
REQS_032a	MISO data valid time (SCK) *Weak driver mode*	B	C_{LOAD} = 60pF	*	30	ns
REQS_032b	MISO data valid time (SCK) *Strong driver mode*	B	C_{LOAD} = 200pF	*	30	ns
REQS_033	MISO data hold time	C		X	*	ns
REQS_034	MISO rise/fall time *Medium driver mode*	D	C_{LOAD} = 30 to 100pF	4	16	ns
REQS_034a	MISO rise/fall time *Weak driver mode*	D	C_{LOAD} = 10 to 60pF	4	16	ns
REQS_034b	MISO rise/fall time *Strong driver mode*	D	C_{LOAD} = 60 to 200pF	4	16	ns
REQS_035	MISO data disable lag time	E		*	50	ns

DEF_036 X: MISO data is guaranteed to be stable until the next SCK shift edge

INFO_037 Parameter A, B do not include the rise/fall time of CS and SCK

INFO_038 To cover all three variants of the parameter B and D min/max time for the specified signal load capacitance ranges, a drive strength configuration of the MISO is proposed.

INFO_039I NFO_039 The following requirements are to the master from slave -point of view:

ID	Parameter	Symbol	Condition	Min	Max	Unit
REQM_040	SCK disable lead time	1		10		ns
REQM_041	SCK enable lead time	2		40		ns
REQM_042	SCK rise and fall time *Medium driver mode*	3	C_{LOAD} = 30 to 100pF	5	15	ns
REQM_042a	SCK rise and fall time *Weak driver mode*	3	C_{LOAD} = 10 to 60pF	5	15	ns
REQM_042b	SCK rise and fall time *Strong driver mode*	3	C_{LOAD} = 60 to 200pF	5	15	ns
REQM_043	SCK high time	4		40		ns

REQM_044	SCK low time	5		40		ns
REQM_045	SCK enable lag time	6		20		ns
REQM_046	SCK disable lag time	7		10		ns
REQM_047	Sequential transfer delay (Out-Of-Frame)	9		700		ns
REQM_048	Sequential transfer delay (In-Frame)	9		200		ns
REQM_049	MOSI rise and fall time *Medium driver mode*	10	C_{LOAD} = 30 to 100pF	5	15	ns
REQM_049a	MOSI rise and fall time *Weak driver mode*	10	C_{LOAD} = 10 to 60pF	5	15	ns
REQM_049b	MOSI rise and fall time *Strong driver mode*	10	C_{LOAD} = 60 to 200pF	5	15	ns
REQM_050	MOSI data setup time	11		10		ns
REQM_051	MOSI data hold time	12		20		ns

INFO_052 To cover all three variants of the parameter 3 and 10 min/max times for the specified signal load capacitance range, a drive strength configuration of the SCK and MOSI is proposed.

INFO_053 To achieve the parameter 9 it takes more time for an Out-Of-Frame protocol since it has to prepare the correct data in between two frames.

4 Logical layer

4.1 General layer description

DEF_054 There are two possible logical frame dependencies within SPI logical layers. One is called in-frame since the data of the slave response is within the same time slot as the masters' request. As out-of-frame the communication is called if the logical response of the slave is within the next frame of the master. Figure 6 depicts both options

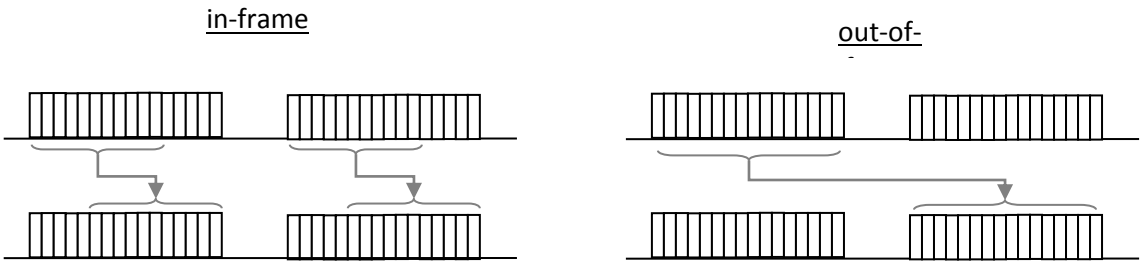

Figure 6 in-frame and out-of-frame communication

REQ_055 The number of bits is fixed to 32.

REQ_056 The MSB is transferred first within a frame.

INFO_057 The SafeSPI supports an out-of-frame as well as an in-frame protocol. Both have the same CRC polynomial and two separate response frames, one for sensor data (identified by D=1) and one for other data (D=0). Sensor data can have up to 16bits.

REQ_064 In case a slave receives a frame he could not "understand" / "decode", the slave should respond accordingly to table §4.2.4 (Fault management), to avoid collision in case of multiple slave with common CS.

4.2 Bit encoding

INFO_058 This section describes the logical encodings for different protocol options.

DEF_059 Bits within the master request (MOSI) or the slave response (MISO) which are marked as '*' can be freely defined and are not specified within this specification.

DEF_060 A bit which is marked as '$' within the slave response (MISO) represents a tri-state of the output pin (high impedance).

4.2.1 General Frame Format (out-of-frame)

REQ_061 The following table shows the command frame format for the out-of-frame protocol

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOSI	TA9:0										*																			C2:0		
value	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Only bits 9:7 of the TA are mandatory in the request.

REQ_062 The following table shows the response frame format for the out-of-frame protocol in case of sensor data is transferred.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MISO	D	SA9:0										S1	DATA15:0																S0	C2:0		
value	**1**	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

REQ_063 The following table shows the response frame format for the out-of-frame protocol for all other data.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MISO	D	SA9:0										*																		C2:0		
value	**0**	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Only bits 9:7 of the SA are mandatory in the response.

4.2.2 General Frame Format (in-frame)

REQ_065 The following table shows the command frame format for the in-frame protocol.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOSI	TA9:5					*																						CC2:0			*	
value	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Only bits 9:7 of the TA are mandatory in the request.

REQ_066 The following table shows the response with sensor data from a slave in the in-frame protocol.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MISO	$					*	D	SA9:5					DATA15:0																S0	CR2:0		
value						0/1	**1**	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

REQ_067 The following table shows the other responses from a slave in the in-frame protocol.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MISO	$					*	D	SA9:5					*																	CR2:0		
value						0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Only bits 9:7 of the SA are mandatory in the response

4.2.3 Description of frame bits (out-of-frame and in-frame)

ID	Symbol	Name
REQ_068	TA9:0	The target address (TA) is the command which defines the command to the sensor. TA9:7 are mandatory TA9:8 shall correspond to the programmable slave address, if the slave supports the use of common CS for up to four slaves TA6:0 are optional and can be used also for other purposes
REQ_069	SA9:0	The source address (SA) is the address uniquely identifying the content of the response data (data15:0). SA9:8 are mandatory and shall correspond to the device individual programmable slave address to allow unique addresses on one SPI bus with up to four slaves. SA7 is mandatory SA6:0 are optional and can be used also for other purposes Example for the use of SA7:0 for a PSI5 transceiver: SA7:5 PSI5 channel SA4:3 PSI5 time slot SA2:0 PSI5 frame (see PSI5 V2.1 substandard Chassis and Safety) Note: In-frame uses only SA9:5.
REQ_070	D	The sensor data bit identifying if response contains sensor data or not (i.e. identifying frame format for response. D=0: no sensor data D=1: sensor data format

REQ_071	S1:0	Out-of-frame format: The sensor status bits describe the status of the sensor data within a data frame response. 00b: valid sensor data 01b: Sensor in any error state or non sensor data available; Data15:0 may contain any kind of data 11b: sensor in init state; Data15:0 still contains sensor data 10b: free to use, sensor data has to be valid
REQ_071a	S0	In-frame format: The sensor status bit describes the status of the sensor data within a data frame response. 0b: valid sensor data 1b: Sensor in any error state or non sensor data available;
REQ_071b	DATA15:0	Up to 16bit of sensor data given in 2nd complement. If sensor data is less than 16bit, data should be aligned on MSB. Unused bits are free to use.
REQ_072	C2:0	CRC for out-of-frame command and response frames calculated over bit 31 to 3. See section 4.3 for details.
REQ_073	CC2:0	CRC for in-frame command frames calculated over bit 31 to 5. See section 4.3 for details.
REQ_074	CR2:0	CRC for in-frame response frames calculated over bit 26 to 3. See section 4.3 for details.

4.2.4 Example of Fault management (out-of-frame and in-frame)

Fault	*In-Frame*		*Out-of-frame*	
	(with common CS)	*(with dedicated CS)*	*(with common CS)*	*(with dedicated CS)*
MOSI CRC error	*last bit of CR2:0 to be inverted by slave (CRC intentionally wrong)*		*High impedance of MISO (in next frame)*	*MISO shall return an error indication**
SCK cycle not equal to 32 (including 0)	*No reaction of slave possible. CRC might be wrong depending on number of bit missing or added.*	*No reaction of slave possible. CRC might be wrong depending on number of bit missing or added.* *Error indication (or information should be given in the next frame if possible)*	*High impedance of MISO (in next frame)*	*MISO shall return an error indication **
TA9:8 (Slave Address)	*High impedance of MISO (in answer frame)*	*NA*	*High impedance of MISO (in next frame)*	*NA*
TA7:0 (for all suported bits)	*MISO shall return an error indication **	*NA*	*MISO shall return an error indication **	*NA*
TA9:0 (for all suported bits)	*NA*	*MISO shall return an error indication **	*NA*	*MISO shall return an error indication **
module internal error	*MISO shall return an error indication (preferable) if possible, if not possible high impedance(no answer)*			

NA: not applicable
*) either status bits (S1:0) or any other mean can be used for this

4.3 CRC protection

REQ_075 For the in-frame format a 3bit CRC with the polynomial 0x5 ($x^3 + x^1 + x^0$) is used with a start value of 111b and a target value of 000b.

REQ_076 For the out-of-frame format a 3bit CRC with the polynomial 0x5 ($x^3 + x^1 + x^0$) is used with a start value of 101b and a target value of 000b.

INFO_076a There are different conventions how to represent a polynomial in hexadecimal notation. The notation 0x5 in this documents is a hexadecimal representation (Koopman notation) of the polynomial x3 + x1 + 1, using x3 as the highest bit and an implicit +1 term.

INFO_077 Note that the bits over which the CRC is calculated is not equal for all frame formats. The following test cases can be used to verify the implementation.

ID	protocol	32bit frame	CRC result
INFO_078	out-of frame, command/response	0x 00 00 00 03	OK
INFO_079	out-of frame, command/response	0x FF FF FF F8	OK
INFO_080	out-of frame, command/response	0x 0F 0F 0F 0A	OK
INFO_081	out-of frame, command/response	0x 0F F2 C8 FE	OK

INFO_082	in-frame, command	0x 00 00 00 04	OK
INFO_083	in-frame, command	0x FF FF FF F7	OK
INFO_084	in-frame, command	0x 0F 0F 0F 13	OK
INFO_085	in-frame, command	0x 0F F2 C8 E7	OK
INFO_086	*in-frame, response*	0x 00 00 00 06	*OK*
INFO_087	*in-frame, response*	0x FF FF FF FC	*OK*
INFO_088	*in-frame, response*	0x 0F 0F 0F 0A	*OK*
INFO_089	*in-frame, response*	0x 0F F2 C8 FE	*OK*
INFO_090	*all responses & commands*	*0x 00 00 00 00*	*FAIL*
INFO_091	*all responses & commands*	*0x FF FF FF FF*	*FAIL*
INFO_092	*all responses & commands*	*0x 0F 0F 0F 0F*	*FAIL*
INFO_093	*all responses & commands*	*0x 0F F2 C8 FA*	*FAIL*

4.4 Example for the use of the target and source address

DEF_094 The following example shows four slaves on a SPI bus, where two support a common CS.

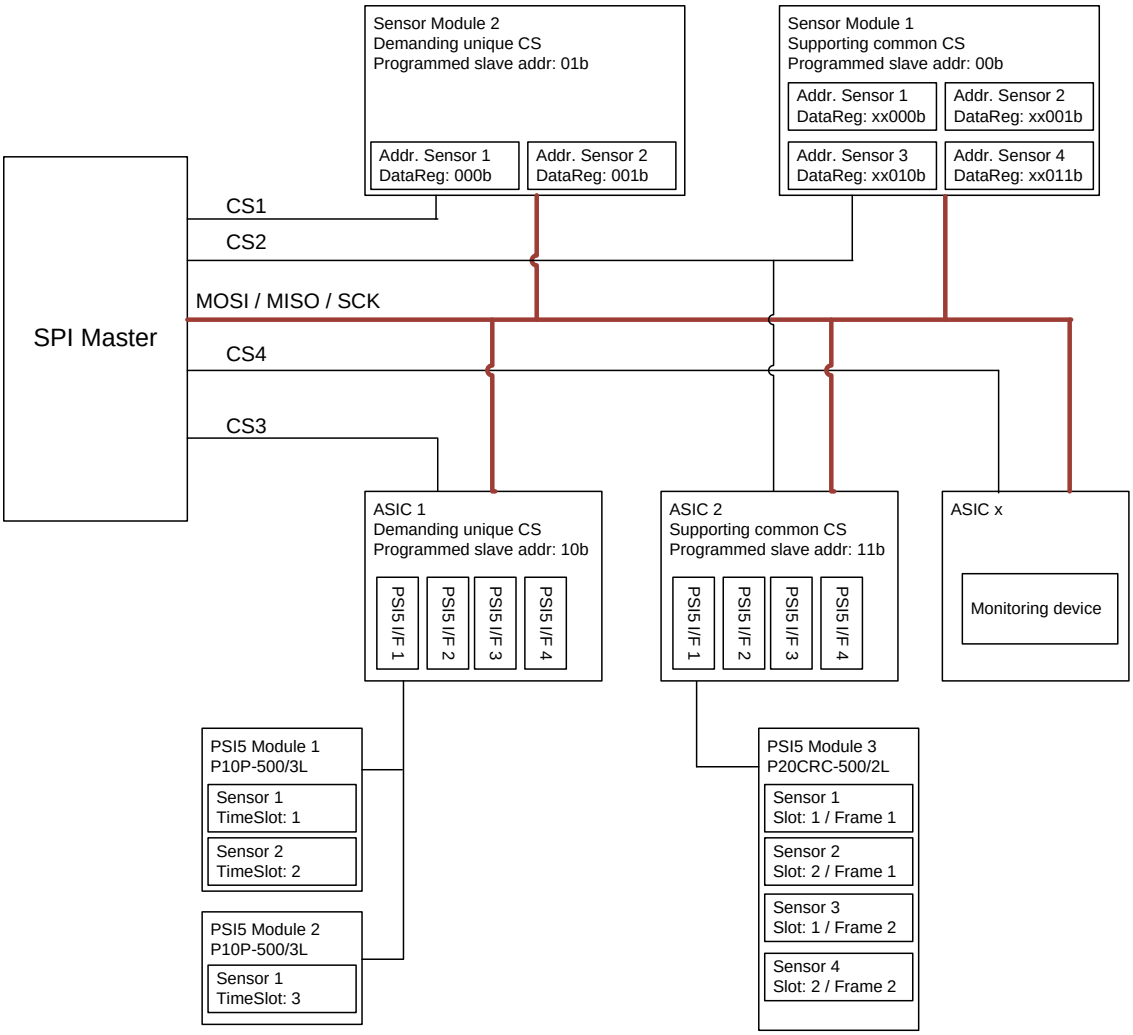

Figure 7 Example of SPI network

The following table shows the SPI transfers, which are required to read the sensor data.

Request			Response	
	TA	CS	D + SA	
Sensor Module 2: Sensor 1	000b	CS1	1 01 000x xxxxb	Slave 2; channel 0
Sensor Module 1: Sensor 1	00 000b	CS2	1 00 000x xxxxb	Slave 1; channel 0
Sensor Module 1: Status Register	00 10110b	CS2	0 00 1011 0xxxb	Slave 1; Non sensor data
ASIC 1: Sensor Data PSI5 I/F 1	000x xxxx xxb	CS3	1 10 000 00 xxxb	Slave 3; PSI5 I/F 1; TimeSlot 1; Frame n/a
ASIC 1: Sensor Data PSI5 I/F 1	000x xxxx xxb	CS3	1 10 000 01 xxxb	Slave 3; PSI5 I/F 1; TimeSlot 2; Frame n/a
ASIC 1: Sensor Data PSI5 I/F 1	000x xxxx xxb	CS3	1 10 000 10 xxxb	Slave 3; PSI5 I/F 1; TimeSlot 3; Frame n/a
ASIC 2: Sensor Data PSI5 I/F 1	11 000x xxxxb	CS2	1 11 000 00 000b	Slave 4; PSI5 I/F 1; TimeSlot 1; Frame 1
ASIC 2: Sensor Data PSI5 I/F 1	11 000x xxxxb	CS2	1 11 000 01 000b	Slave 4; PSI5 I/F 1; TimeSlot 2; Frame 1
ASIC 2: Sensor Data PSI5 I/F 1	11 000x xxxxb	CS2	1 11 000 00 001b	Slave 4; PSI5 I/F 1; TimeSlot 1; Frame 2
ASIC 2: Sensor Data PSI5 I/F 1	11 000x xxxxb	CS2	1 11 000 01 001b	Slave 4; PSI5 I/F 1; TimeSlot 2; Frame 2

5 Change History

Rev.N°	Chapter	Description / Changes	Date
0.15	All	Initial version of this specification for industry review	July 30, 2015
0.16	All	Change accordingly to "*2016-04-14_SafeSPI_Change_Requests.xlsx*"	March , 2016
0.17	All	Change accordingly to "*Rev016_SafeSPI_Change_Request_Review_Meeting_20-05-2016.xlsx*"	May 25, 2016
V1.0	§3.1	INFO_011 and DEF_012 updated accordingly to Fig 3	June 3, 2016
	INFO_024	Updated	June 3, 2016
	REQ_064	New wording	June 3, 2016