
**Road vehicles — Deployment and sensor
bus for occupant safety systems**

*Véhicules routiers — Bus de déploiement et de capteurs pour les
systèmes de sécurité des occupants*



Reference number
ISO 22896:2006(E)

© ISO 2006

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

© ISO 2006

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword.....	iv
1 Scope	1
2 Terms and definitions.....	1
3 Abbreviations	3
4 General.....	4
5 System architecture.....	5
5.1 General.....	5
5.2 Deployment bus	5
5.3 Sensor bus	5
5.4 Combined sensor and deployment bus	6
6 Physical Layer.....	6
6.1 Bus medium	6
6.2 Bus topology	6
6.3 Bus load.....	8
6.4 Bus signals.....	10
6.5 Bit coding	12
6.6 Fault tolerance	15
6.7 Use of analogue safing on a deployment bus	17
6.8 Bus signal parameters	18
7 Data Link Layer	22
7.1 Bus Idle	22
7.2 Addresses.....	22
7.3 Message frames	24
7.4 Bit fields within a frame.....	32
8 Application Layer.....	35
8.1 General.....	35
8.2 Common D-Frame commands.....	36
8.3 Memory layout of slaves	37
8.4 Application Layer for deployable devices.....	42
8.5 Application Layer for sensor devices.....	47
Annex A (informative) In-car address programming for daisy-chain systems	50
Annex B (informative) Guideline for definition of deviations from standard parameters	51
Annex C (informative) Rationale of functionality	52
Annex D (informative) Latency time analysis for interrupts from smart sensors	53
Annex E (informative) CRC examples	56
Annex F (informative) Deployable devices	57
Annex G (informative) Slave manufacturer identification codes.....	60

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 22896 was prepared by Technical Committee ISO/TC 22, *Road vehicles*, Subcommittee SC 3, *Electrical and electronic equipment*.

Road vehicles — Deployment and sensor bus for occupant safety systems

1 Scope

This International Standard is a specification of a serial communications bus protocol for automotive occupant restraint systems. It covers Physical Layer and Data Link Layer and those parts of the Application Layer that are not supplier-specific.

2 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

2.1

analogue safing

using a special *bus level* (*LS0-level*) for confirmation of deploy messages

2.2

bitmap addressing

method of addressing one or several slaves at a time by assigning each bit of the address field to a different *slave*

2.3

bus level

one out of four levels of the differential bus voltage, whereof one forms the *Power Phase* and the other three are used for representation of a data bit during the *Data Phase*

2.4

command

part of a *D-Frame*, transmitted by the master, defining the purpose of the frame

2.5

CRC field

part of a *D-Frame* or *S-Frame*

2.6

data field

part of a *D-Frame*

2.7

Data Phase

part of a data bit providing the bit value

2.8

deploy command family

four commands for control of *deployable devices*

2.9

deployable device

irreversible actuator

2.10

D-Frame

type of frame primarily used for diagnostic communication and actuation of *deployable devices*

2.11

differential bus voltage

differential voltage between the two bus wires

2.12

duty cycle

percentage of a bit time that is assigned to the *Power Phase*

2.13

E-bit

bit in a *D-Frame* indicating an error or a “read” command

2.14

half-rate

mode used for sensors that shall not reply in every *S-Frame*

2.15

hold-up capacitor

capacitor supplying power to a *slave* during the *Data Phase*

2.16

latency time

worst-case duration between the occurrence of an interrupt requesting event in the sensor and the actual start of an *S-Frame* polling message

2.17

LS0-level

bus level indicating an error, a bus interrupt or a “0” with *analogue safing*

2.18

L0-level

bus level indicating a “0”

2.19

L1-level

bus level indicating a “1”

2.20

master

device responsible for communication on the bus and for power distribution over the bus

2.21

Multi-Sharing

mode used in *S-Frames* for dynamic assignment of slave data to the first slot

2.22

node

master or slave

2.23

point-to-point addressing

addressing used for communication between the *master* and one *slave*

2.24**power level**

bus level forming the Power Phase

2.25**Power Phase**

part of a data bit during which the *master* transmits the *power level*

2.26**R-bit**

reserved bit in *D-frames* for future definition

2.27**SEL-bit**

bit used in *S-Frames* to control *slaves* configured for *half-rate* mode

2.28**S-Frame**

type of frame used by the *master* to collect dynamic data from *slaves* periodically

2.29**signal address**

address assigned to peculiar signals provided by *slaves*, used in *S-Frames* for *Multi-Sharing*

2.30**slave**

device that is connected to the bus and is not the *master*

2.31**slave address bitmap**

part of a *D-Frame* in which each bit corresponds to one *slave*

2.32**slot**

part of an *S-Frame* assigned to a certain *slave* to be filled with its data

2.33**Slot Length**

determines the number of data bits that a *slot* consists of

2.34**Sub-Slot**

sub-section of a *slot*

2.35**T-bit**

first bit of a frame, used to define the frame type (*S-* or *D-Frame*)

3 Abbreviations

ACU Airbag Control Unit

ASIC Application-Specific Integrated Circuit

CRC Cyclic Redundancy Check

ECU Electronic Control Unit

ISO 22896:2006(E)

HSD	High Side Driver
INT	Interrupt
LSB	Least Significant Bit
LSD	Low Side Driver
MSA	Multi-Sharing Address
MSB	Most Significant Bit
MTP	Multi Time Programmable
NVM	Non Volatile Memory
ORC	Occupant Restraint Controller
OTP	One Time Programmable
RAM	Random Access Memory
RCM	Restraint Control Module
ROM	Read Only Memory
SDM	Sensing and Diagnostic Module
SEL	Select
SOF	Start Of Frame
SSB	Slot Start Bit

4 General

Automotive occupant restraint systems are controlled by a Sensing and Diagnostic Module (SDM), also called Airbag Control Unit (ACU), Restraint Control Module (RCM) or Occupant Restraint Controller (ORC), which is connected to peripheral devices:

- dynamic sensors with high update rates, e.g. for remote front and side impact sensing;
- static sensors with low update rates, e.g. buckle switches, seat position and occupancy sensors;
- actuators, especially deployable devices, e.g. squibs.

The SDM is also referred to as “master”; the peripheral devices are also referred to as “slaves”.

The bus provides a two-wire connection between the SDM and the peripheral devices and supplies power to the slaves. It offers bi-directional communication. The master’s bus interface sends energy into the bus, the slave’s bus interface extracts power from the bus. The master determines the bus speed and initiates all communication by sending message frames on the bus. Slaves may transmit their data within these frames when requested by the master. Smart dynamic sensors (defined in 5.3) may send an interrupt to the master while the bus is idle or while there is diagnostic communication on the bus. The master’s reaction to the interrupt is application specific and typically lets the master stop diagnostic communication and start polling of impact data instead.

The data is usually coded using differential bus voltage. On a bus, where several transmitters are sharing the same wiring, using voltage as the data signal has a significant advantage over current, because it enables the transmitter to verify the data that it sent on the bus. This is the most reliable way to detect bus collisions, e.g. when two sensors are transmitting their data at the same time. For less critical data like diagnostics, reply data from slaves can be coded using current, which allows connection of deployable slaves to the bus via isolation resistors (see 6.6.4.2).

5 System architecture

5.1 General

The specification covers sensor busses, deployment busses and combined sensor/deployment busses.

The bus shall support 64 slave addresses, of which three shall be reserved for special purposes. The actual number of slaves that can be connected to one bus is limited by the supply current for the slaves and by the pin capacitance of the slaves (see also Clause 6). Bandwidth limits shall also be considered.

NOTE A single slave can incorporate the functionality of several slave addresses.

5.2 Deployment bus

The deployment bus shall support deployable devices and static sensors. The bus shall provide point-to-point messages for diagnostic communications between master and slaves. Since the deployment bus shall support fast selective deployment of several deployable devices, the bus shall also provide a special deploy message, which allows individual deployment control of up to 12 devices at a time. There shall be four deploy messages available, each controlling 12 device addresses:

- address range 0b000000 – 0b001011;
- address range 0b010000 – 0b011011;
- address range 0b100000 – 0b101011;
- address range 0b110000 – 0b111011.

In this way, up to $4 \times 12 = 48$ deployable devices can be controlled by one bus. The address 0b000000 should not be used as a slave address, because this address shall be the default address of all slaves that have not been programmed yet. See also 7.2.1, 7.3.2 and 7.4.8.

The deployment bus shall provide communication with and without a special “safing” signal, which may be used for additional differentiation between diagnostic communication and actual deploy commands.

5.3 Sensor bus

The sensor bus shall support static and dynamic sensors. There may be two types of dynamic sensors.

- **Raw-data sensors** send time-critical data periodically to the SDM.
- **Smart sensors** send time-critical data event-driven only.

Smart sensors can easily coexist with static sensors on the same bus.

NOTE Raw-data sensors usually occupy the bus bandwidth all the time, while smart sensors usually need the full bandwidth only for a short time during an event.

EXAMPLE In the absence of an event, the master can poll diagnostic data and/or static sensor data from all slaves. When an event occurs, a smart sensor can stop this communication by sending an interrupt to the master and to the other slaves. The master can then assign the full bus bandwidth for exclusive communication of time-critical data from smart sensors to the master.

The number of smart sensors that can be connected to the bus is usually limited by the ratio between the available bandwidth and the latency time requirements for this data transfer. Additional static sensors on the bus do not contribute to the latency time, but they contribute to the physical bus load, which also limits the number of slaves (see Clause 6) on the bus.

On a sensor bus, the “safing” signal, known from the deployment bus, shall be used for error indication and optionally for the interrupt capability of smart sensors.

Since raw-data sensors usually are not required to send bus interrupts, they may be implemented without this option. Devices (master and slaves) made for raw-data sensor busses should either not have bus interrupt capability or provide a means to disable the bus interrupt function in a reliable way.

5.4 Combined sensor and deployment bus

On a combined sensor and deployment bus, all types of slaves that are connected to the same bus would have to share the available bandwidth and the available bus power. This shall be taken into account when designing such a mixed system.

The “safing”-level LS0 shall be used on the one hand for confirmation of deploy messages (i.e. LS0 transmitted by the master), and on the other hand for signalling bus collisions (i.e. LS0 transmitted by a dynamic sensor during an S-Frame) or for interrupting communication (i.e. LS0 transmitted by a smart sensor during a D-Frame). The deployment and sensor bus protocol shall ensure that the relevant function of the LS0-level can be clearly identified by all nodes (see 8.5.4).

6 Physical Layer

6.1 Bus medium

The bus can use unshielded twisted pair or untwisted cable (see Table 3). The maximum bus length depends on the bus topology (see 6.2).

6.2 Bus topology

6.2.1 Parallel configuration

For a parallel bus configuration, each slave shall be directly connected to the two bus wires Bus-A and Bus-B (see Figure 1).

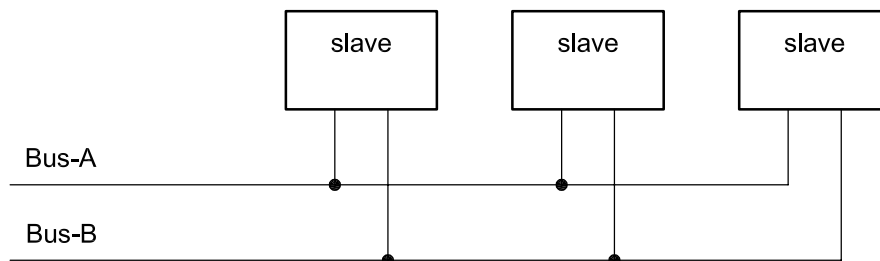


Figure 1 — Parallel connection of slaves to the bus

In parallel configuration, the wires may be routed in a bus, tree or ring structure (see Figure 2) or combinations of these. A ring may be implemented either by connecting both ends of the bus cable to a single bus output of the master, or by connecting each end of the cable to a separate output of the master (see also 6.6.3.2).

Parallel squibs can be implemented as polarized or non-polarized devices. For non-polarized devices, it does not matter which pin is connected to which bus wire. Dynamic sensors shall be polarized.

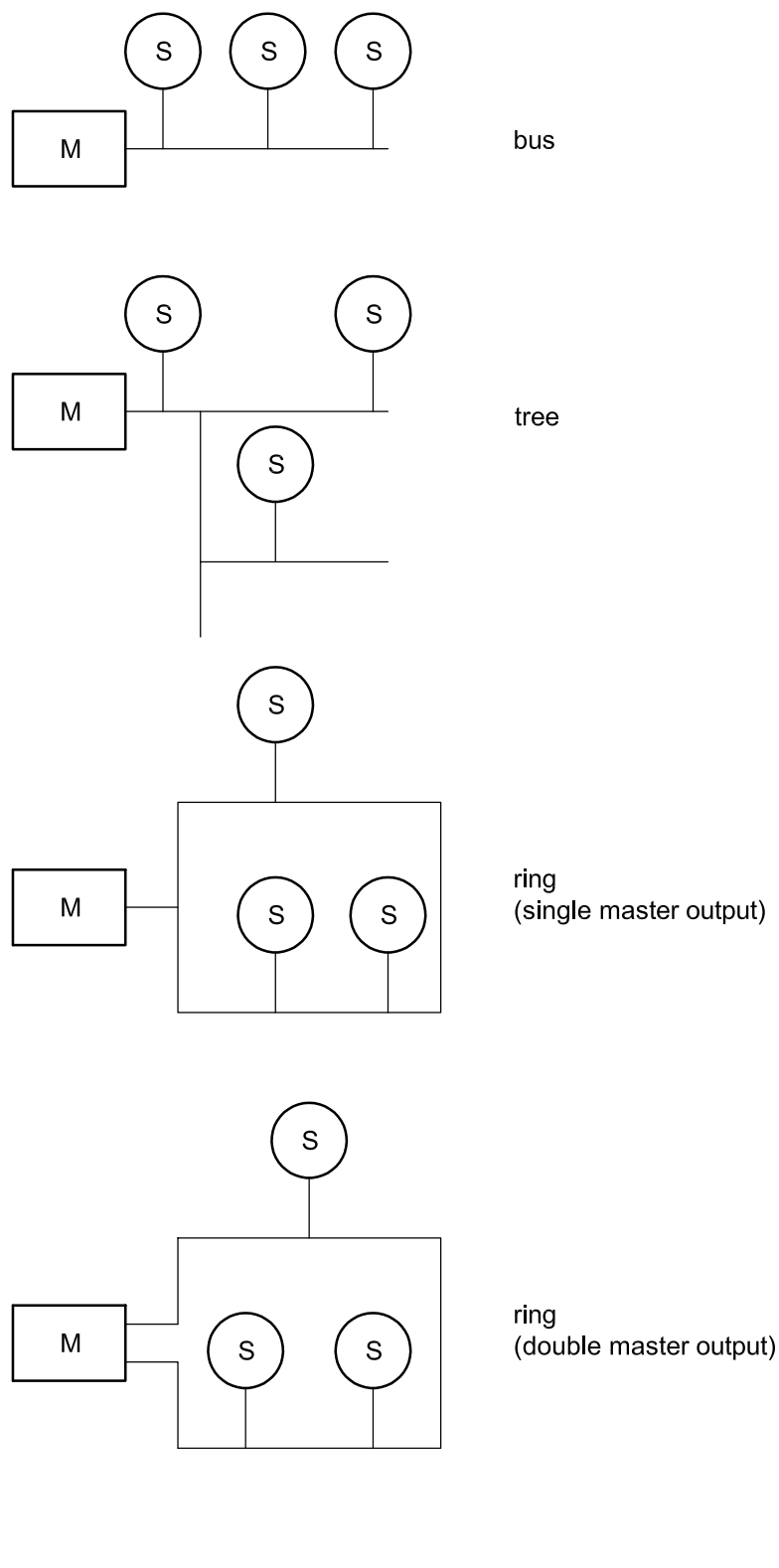


Figure 2 — Bus topologies for parallel configuration

6.2.2 Daisy-chain configuration

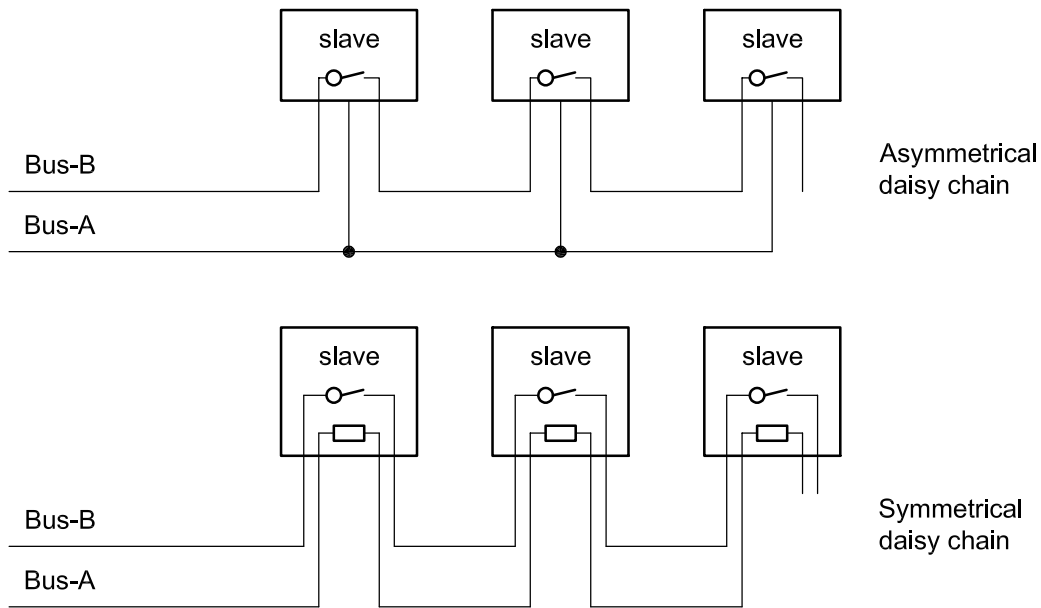


Figure 3 — Daisy-chain connection of slaves to the bus

For an asymmetrical daisy-chain configuration, each slave shall be directly connected to Bus-A. Bus-B shall be routed through a switch in each slave (see Figure 3). These daisy-chain switches shall split the bus into bus sections. Single switches can be opened in order to shut down individual bus sections, which can be used for the following:

- in-car address programming, where slaves are identified by their position in the daisy-chain (see Annex A);
- recovery from Bus-A to Bus-B shorts (see 6.6.4.3), where the faulty bus section is switched off.

For a symmetrical daisy-chain configuration, each slave shall switch Bus-B and insert an element into Bus-A. Ideally, the on-resistance of the switch and the resistance of the element should be identical, in order to keep the balance of the bus voltage behind the daisy-chain slave. Symmetrical daisy-chain configurations should be chosen when the bus is run permanently at high speed.

In a daisy-chain configuration, the wires may be routed in a bus or ring structure (see also 6.6.3.2). Between two daisy-chain slaves, parallel slaves may be connected in a bus or tree structure.

Daisy-chain slaves shall be polarized with respect to the exchange of Bus-A with Bus-B. For operation in a ring, they shall not be polarized with respect to the exchange of switch input with switch output.

6.3 Bus load

6.3.1 General

The master shall provide enough current to supply the slaves and to drive the signal edges. A slave replying with voltage modulation shall sink enough current to drive the signal edges.

6.3.2 Bus capacitance

Each slave, the bus wires and the master itself contribute to the capacitance that is limiting the slope of the signal edges. Since parallel slaves have less capacitance than daisy-chain slaves, a higher number of parallel slaves and/or a longer bus may be admitted.

The total bus capacitance shall be less than the value that master and slaves can drive. The latter can depend on the bus speed (i.e. higher bus speed requires lower capacitance).

EXAMPLE

Calculations for a deployment bus:

$$C_c = 61 \text{ pF/m}$$

$$C_{\text{mstr}} = 5 \text{ nF}$$

where

C_c is the cable capacity per metre;

C_{mstr} is the master pin output capacitance (effective capacitance between Bus-A and Bus-B, when one bus wire is shorted to ground), see Table 1.

NOTE This corresponds for instance to 2,2 nF capacitors (with a relative tolerance of $\pm 10 \%$) connected from Bus-A to GND and from Bus-B to GND at both master outputs driving a ring. If a single output is driving the bus, the capacitor values can be doubled.

*Calculation for **parallel** slave configuration*

$$C_{\text{ab}} = 250 \text{ pF}$$

$$l_{\text{par}} = 40 \text{ m}$$

$$n_{\text{par}} = 16$$

where

C_{ab} is the maximal slave capacitance of each slave in parallel configuration;

l_{par} is the maximal permissible cable length in parallel configuration;

n_{par} is the maximal number of slaves that may be connected to the bus in parallel configuration.

The total bus capacitance in parallel configuration, C_{totpar} is calculated as follows:

$$C_{\text{totpar}} = (C_c \times l_{\text{par}}) + (n_{\text{par}} \times C_{\text{ab}}) + C_{\text{mstr}}$$

$$C_{\text{totpar}} = (61 \text{ pF/m} \times 40 \text{ m}) + (16 \times 250 \text{ pF}) + 5 \text{ nF}$$

$$C_{\text{totpar}} = 2,44 \text{ nF} + 4 \text{ nF} + 5 \text{ nF} = 11,44 \text{ nF}$$

*Calculation for **daisy-chain** slave configuration*

$$C_{\text{dc}} = 500 \text{ pF}$$

$$l_{\text{dc}} = 25 \text{ m}$$

$$n_{\text{dc}} = 12$$

where

C_{dc} is the maximal slave capacitance of each slave in daisy-chain configuration ($2 \times C_{ab}$ because of two connections to the bus);

l_{dc} is the maximal permissible cable length in daisy-chain configuration;

n_{dc} is the maximal number of slaves that may be connected to the bus in daisy-chain configuration.

The total bus capacitance in daisy-chain configuration, C_{totdc} , is calculated as follows:

$$C_{totdc} = (C_c \times l_{dc}) + (n_{dc} \times C_{dc}) + C_{mstr}$$

$$C_{totdc} = (61 \text{ pF/m} \times 25 \text{ m}) + (12 \times 500 \text{ pF}) + 5 \text{ nF}$$

$$C_{totdc} = 1,525 \text{ nF} + 6 \text{ nF} + 5 \text{ nF} = 12,525 \text{ nF}$$

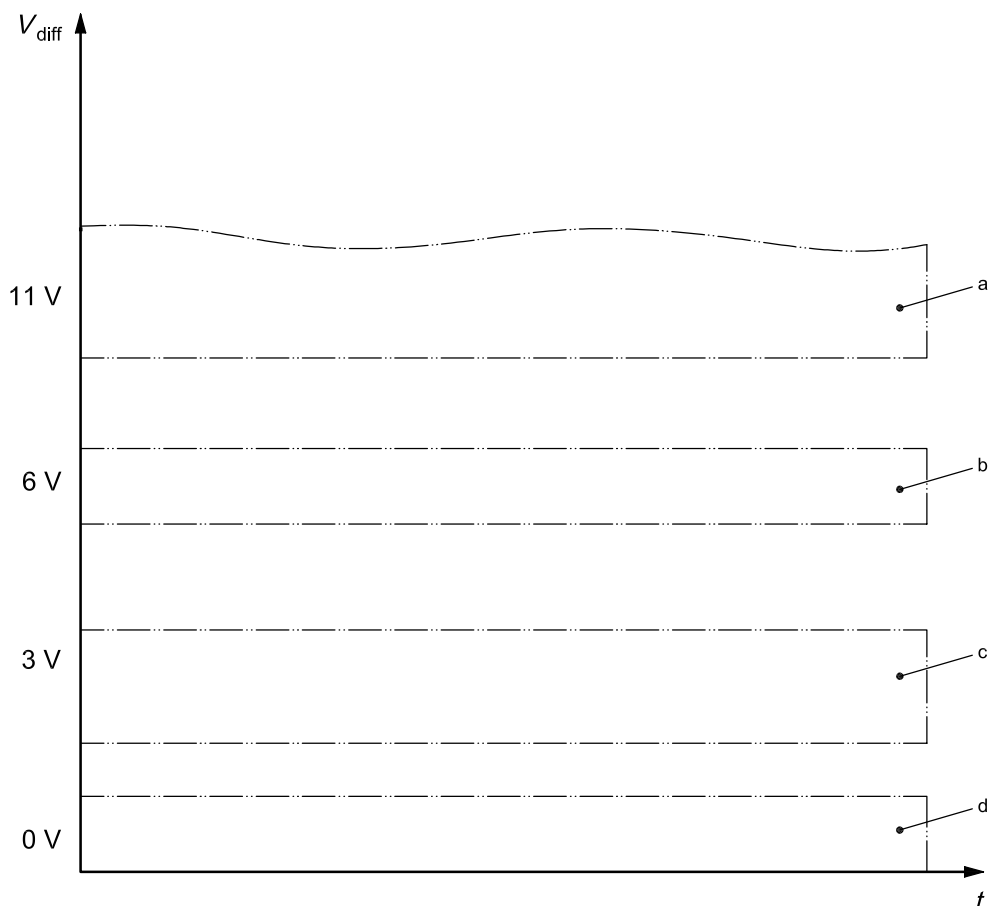
In these examples, the total bus capacitance is 12,525 nF for a 25 m bus with 12 daisy-chain slaves, or 11,44 nF for a 40 m bus with 16 parallel slaves.

6.3.3 Slave supply current

A slave shall not get a d.c. supply current from the bus. The bus shall provide power pulses with a duty cycle of about 50 %. A hold-up capacitor internal to each slave can be used to supply power to the slave in between power pulses. A slave shall extract power from the power pulses only, even during power-on, when its hold-up capacitor has not yet been charged up. The average surge current needed by the slave for power extraction is about twice its average d.c. supply current, because of the 50 % duty cycle. A built-in current limit in the slave shall balance the surge current over the length of the power pulse.

6.4 Bus signals

The bus shall use differential voltage signals for communications from master to slaves. Slave data shall be transmitted either by differential voltage as well, or by current modulation. All voltage signals shall have the same polarity $V_{Bus-A} - V_{Bus-B} > 0$. Differential bus voltages are shown in Figure 4.



- a Power distribution level "P".
- b L0 = normal data level "0".
- c L1 = normal data level "1".
- d LS0 = special data level "0" for safing, interrupt or error.

Figure 4 — Differential bus voltage levels

There shall be one bus level for power distribution and three levels for data exchange:

- P = power level;
- L0 = recessive data level for the bit value "0";
- L1 = dominant data level for the bit value "1";
- LS0 = dominant data level for either
 - the interrupt signal,
 - the bit value "0" with analogue safing (for deploy messages), or
 - an error indication at certain bit positions (see 7.4.9 and 7.4.10).

A slave shall only transmit when the master does not output the power level. In order to allow a slave to transmit, the master shall transmit a default "0" (L0-level), which a slave may overwrite by pulling the bus down to the L1-level for transmission of a "1". Dynamic sensors shall modulate the bus voltage for

transmission of data. This way they can read back the bus signal while transmitting and detect faults like collisions with other transmitting nodes. On a deploy bus, static sensors or deployable devices may not be able to modulate the bus voltage at high bus speeds, because their transmit current may be too weak to discharge the bus capacitance in time. In this case the master shall recognize the slave data by evaluating the bus current.

A dynamic sensor slave shall have the additional option to pull an L0- or L1-level down to the LS0-level, which means either that the slave can submit an interrupt to the master or that it can indicate a detected error during transmission (see 7.4.9 and 8.5.4).

The master shall use either the L0-level or the LS0-level for the representation of a “0”, depending whether the message is a normal message or a qualified one (“analogue safing”). See also 6.7.

The P-level shall be about 12 V for normal operation, or it may be raised up to 30 V for special purposes such as OTP programming in the slaves. Raising the P-level shall not be considered applicable in the system, but for off-line use only. Slaves not requiring an increased P-level for certain functions need not tolerate such a level. All slaves shall tolerate and shall not be damaged by a differential bus voltage up to 20 V. This takes into account failure modes in the airbag ECU that may lead to a temporarily increased bus level.

L0 can be overwritten by L1 or by LS0; L1 can be overwritten by LS0. The P-level cannot be overwritten by any other level. The exact bus levels are specified in 6.8.

NOTE Higher voltage at same slew rate level implies lower speed.

6.5 Bit coding

6.5.1 General

The time during which the master transmits the power level is called the Power Phase. The time during which the master transmits a data level is called the Data Phase. The master shall transmit a steady sequence of Power and Data Phases. During one Data Phase, one bit shall be transmitted (see Figure 5). The slaves shall extract power from the bus during the Power Phase only. The average duty cycle for the Power Phase should be about 50%.

Speed changes for high-speed deployment messages can be initiated by the master. For high-speed sensor polling messages, speed changes can be initiated by a sensor with interrupt capability.

Speed changes shall go into effect with the beginning of a new Power Phase (for details and restrictions, see 6.5.2).

When the master does not transmit a message (= Bus Idle) or when the master transmits a frame in which it expects a slave to transmit data, the master shall transmit the L0-level during Data Phases. The L0-level can be overwritten by another data level from a slave, when the slave is transmitting data. In order to transmit the bit value “0”, the slave shall leave the Data Phase as is. In order to transmit the bit value “1”, the slave shall pull the bus voltage down to L1-level. On a deploy bus during D-Frames, the master shall read the slave’s transmit current, instead of checking the bus voltage. Therefore, it is not necessary that the slave actually pull the bus voltage completely down to L1-level. In order to send an interrupt or to indicate an error during S-Frames, the slave shall pull the bus voltage down to LS0-level (applicable to dynamic sensors only).

NOTE Data bits are surrounded by power distribution level phases. The level “LS0” is used either for analogue safing of deploy messages, or for signalling an interrupt or error condition from a dynamic sensor to the master.

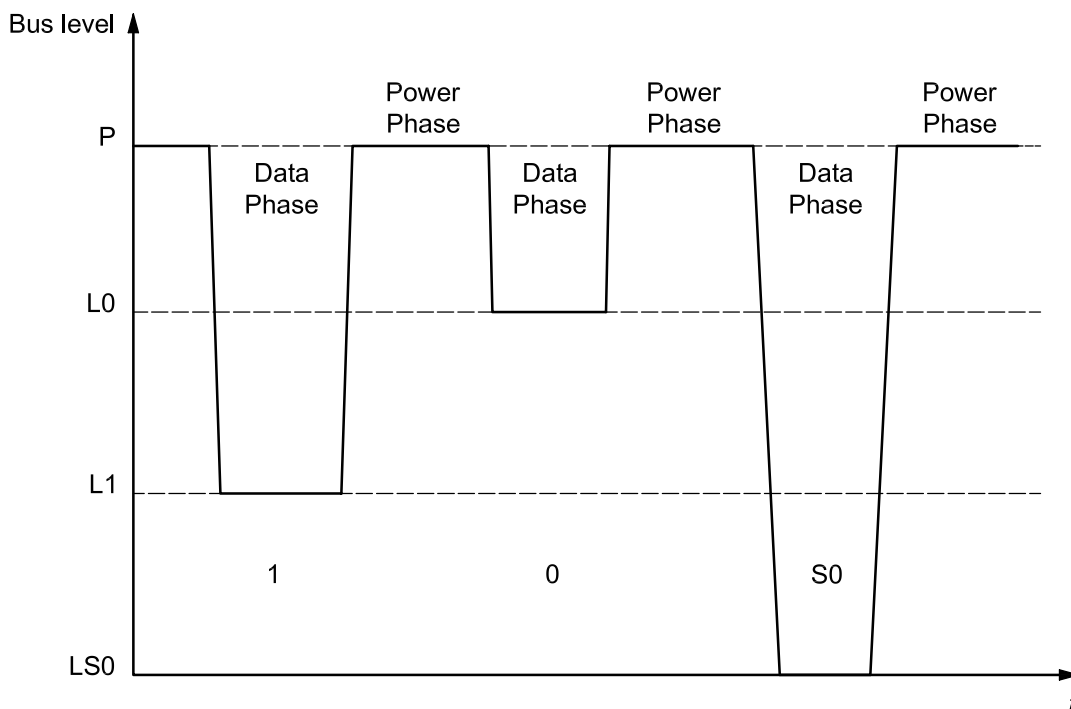


Figure 5 — Illustration of the bit coding

6.5.2 Start of frame (SOF)

The Physical Layer shall provide a special SOF symbol, which is characterized by a temporary doubling of the duration of the Power Phase (P) and of the following Data Phase (0) of one bit (see Figure 6).

The bit level within the SOF shall indicate the type ("T", see 7.4.2) of the subsequent frame.

The master shall start a new frame by sending the SOF symbol. Usually, this is done after completion of a previous frame or during Bus Idle. The master can also send a new SOF during a frame. This shall terminate the running frame and shall start a new one. For this case, the following rules apply.

- The master shall not send an SOF immediately after an SOF, there shall be at least one data bit in between.
- When the current frame is a D-Frame requesting data and the new frame is also a D-Frame (7.3.2), the master shall send the SOF twice, with at least one idle bit in between (see also Annex C). When the new frame is a deploy command with safing, the idle bit(s) between the dual SOF shall be transmitted with safing level as well (LS0).

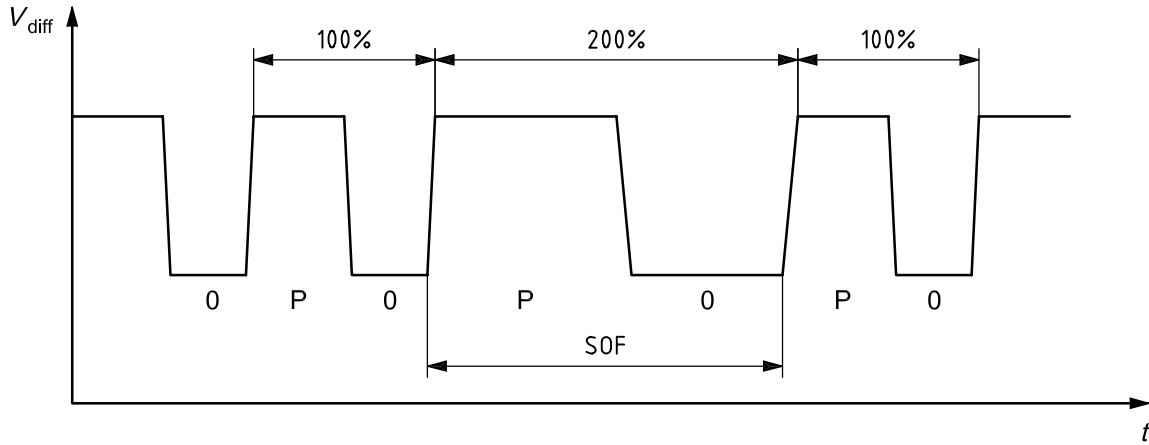


Figure 6 — Characterization of the SOF by a temporary doubling of the bit time

The start of a new message may also be used as the time to decrease the bus speed permanently (see Figure 7). Any lengthening of the bit time by a factor of two or more shall be recognized as the start of a new frame.

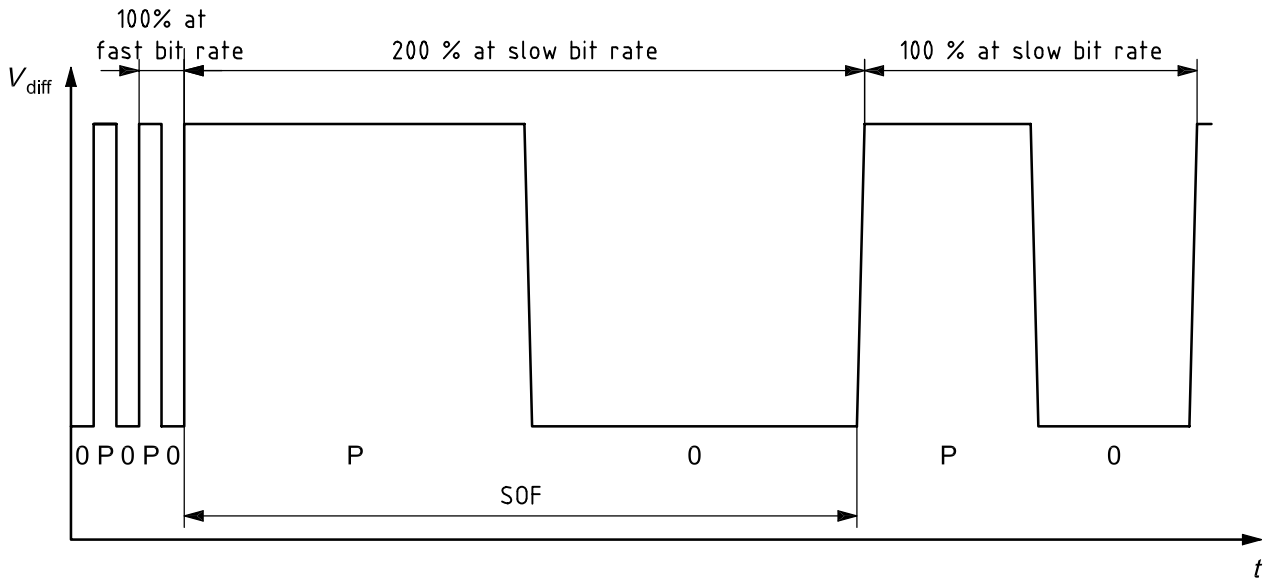


Figure 7 — Example for changing the bus speed from fast to slow with the start of a new frame

The bus speed may be increased at any time during a frame, except immediately before an SOF. If the request to increase speed and an SOF occur coincidentally, the master shall switch to high speed one bit before it sends the SOF (Figure 8).

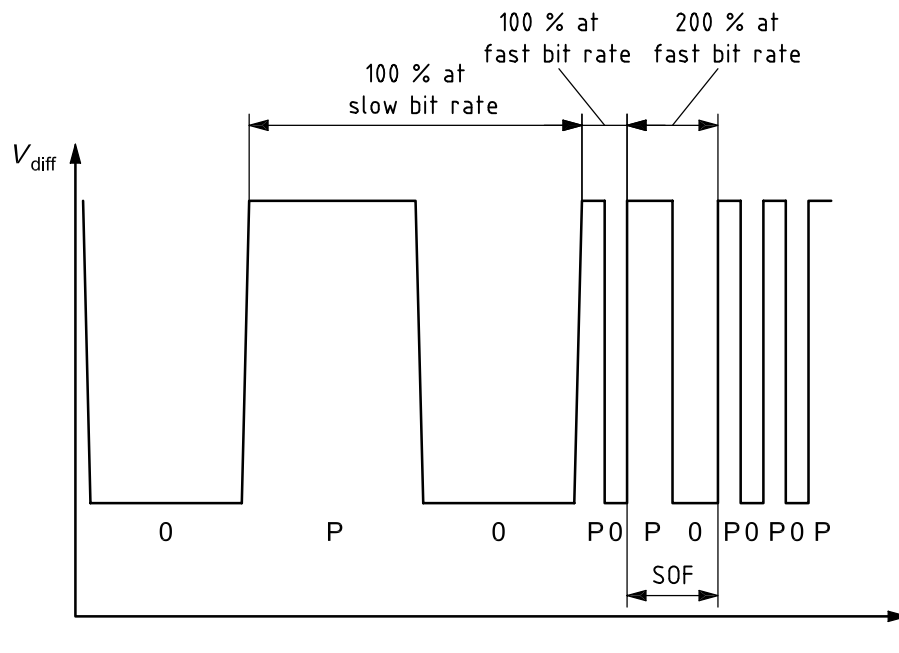


Figure 8 — Example for changing the bus speed from slow to fast one bit before the start of a new frame

NOTE For simplicity, the data values in Figure 6, Figure 7 and Figure 8 are all “0”.

6.6 Fault tolerance

6.6.1 General

This concept provides several options to implement fault tolerance. It shall cover

- shorts of one bus wire,
- open circuits,
- shorts between the bus wires (A-to-B shorts).

6.6.2 Shorts of one bus wire

The main approach for tolerance of single-wire shorts is to make the bus float. Due to the definition of a steady 50 % duty cycle even during Bus Idle, capacitor-based circuitry can be used to make the bus float, without having to use a transformer. This way, shorts of one wire to any voltage within the range of at least -2 V to $+16\text{ V}$ are tolerated, no matter whether this is a transient, intermittent or continuous short and no matter the resistance of the short.

6.6.3 Wire interruptions

6.6.3.1 General

Without a ring structure, the bus shall tolerate open bus wires by maintaining communication to all slaves between the master and the fault.

6.6.3.2 Ring structure

On a pure parallel bus, the ring can be driven by a single master output (see Figure 2). In this case, an open wire within the ring shall be completely tolerated. The master cannot check whether the ring is open or closed.

The ring can be driven from two master outputs as well. In this configuration the master can check whether the ring is open or closed. However, the electrical bus parameters on a closed ring differ from those on an open ring. Therefore some restrictions apply, which are explained below.

All voltage-specific bus signals shall be independent of the status of the ring. This includes all urgent messages like commands from master to slaves on a deploy bus and sensor data on a sensor bus. For these messages, the frames shall be transmitted by the master on both outputs. Slave data shall be decoded on both outputs, the actual slave data being the result of a bit-wise OR function of the individual signals. The bus transfer of such messages shall fully tolerate any opening or closure of the ring. No such message shall be lost.

All current-specific bus signals, such as reply data from slaves within D-Frames, are dependent on the status of the ring. When the master data bias current (I_{Dsrc}) (see Table 1) is driven from both sides at the same time, the ECU may not be able to receive such slave data. Therefore the following rules apply to a ring driven by two bus outputs.

- For D-Frames, I_{Dsrc} shall be switched off on **one** bus output during bits that carry slave data.
- The slave current shall be checked at the one output at which I_{Dsrc} is active.
- When there is no slave reply detectable, the message frame shall be repeated with I_{Dsrc} being active at the other output and checking the slave current from there. This could happen when the ring is open. The master shall locate the open wire by checking which slave is replying to which output.

6.6.4 Bus A-to-B shorts

6.6.4.1 General

The bus shall fully tolerate Bus A-to-B short circuit currents of up to 5 mA. On a parallel bus, the ability to send and receive messages with safing level shall be maintained even with short circuit currents of up to 20 mA. On a daisy-chain bus, short circuits of more than 5 mA can be isolated by reconfiguration of the daisy-chain.

6.6.4.2 Recovery from A-to-B shorts on a deployment bus in parallel configuration

Parallel deployable devices or static sensors on a deployment bus shall be connected to the bus through discrete resistors. These resistors prevent the slave from shorting the bus wires A-to-B. The resistors may be part of the slaves, or they are inserted in the wiring harness between master and slave. Both solutions may also be combined (see Figure 10). However, such resistors shall only maintain the capability of transmitting messages with analogue safing from master to slaves, in spite of an A-to-B short. This means that deploy messages can still be sent, but diagnostic communication or polling of sensor data can no longer proceed, because of the short.

6.6.4.3 Recovery from A-to-B shorts in daisy-chain configuration

There shall be two ways of recovering from an A-to-B short:

- real-time recovery by hardware;
- recovery by software.

Recovery by software may be chosen for all kind of problems where the origin of the problem is located by testing the individual bus sections. Once it has been identified that the occurrence of the problem is correlated

to the closure of one of the daisy-chain switches, the master can decide to leave this particular switch off in order to maintain communication in the remaining bus sections.

Hardware recovery shall be available for recovery from “true” shorts. A “true” short is defined as a low-ohmic short that prevents the master from generating a sufficient power level, V_{LP} , with the consequence that the bus signal does not exceed the V_{P0} threshold at one or several nodes (which can include the master itself).

A slave shall detect a “true” short by checking the time having elapsed since the last Power Phase. When this time is significantly longer than the duration of one bit at lowest bus speed, the slave shall assume the presence of a “true” short and open its daisy-chain switch.

A master shall detect a “true” short by checking the bus level during Power Phase. When the bus level is lower than the V_{P0} threshold, the master shall assume the presence of a “true” short and shut down the bus output.

When a slave has detected a “true” short and has opened its daisy-chain switch accordingly, it shall wait for a bus voltage higher than the V_{P0} threshold on one of the two sides of the daisy-chain switch. When this is available, it shall forward a test current to the other side during Power Phase. If the bus voltage on this side exceeds the V_{01} threshold again, it is assumed that the short is gone and the slave shall switch on the daisy-chain switch again.

When a master has detected a “true” short and has switched off the bus accordingly, it shall wait for one bit time and afterwards turn on a test current to check if the short is still present. When the bus voltage reaches the V_{01} threshold again, it is assumed that the short is gone and the master shall switch on the bus output again.

The power-up behaviour of a master or a slave shall be basically the same as described above for recovery from an A-to-B short. The bus shall be tested with a test current: if positive, the bus shall be switched on; if negative, the master can double-check by commanding the closure of a daisy-chain switch and simply testing whether it can still communicate afterwards.

During initial power-on of a slave, its daisy-chain switch shall be open. Immediately after power-on, the slave shall behave as after an A-to-B short detection: when V_{P0} is reached on one side, the test current shall be switched on and the daisy-chain switch shall be closed as soon as V_{01} has been crossed on the other bus side.

NOTE Even when the slave fails to close the switch automatically, the master can still command the closure of the daisy-chain switch to double-check whether there is really an A-to-B short behind that slave.

6.7 Use of analogue safing on a deployment bus

6.7.1 General

The master shall send out a message with analogue safing almost in the same way as without safing. It shall simply replace the L0-level by the LS0-level for representation of a “0”, and keep the L1-level for representation of a “1”.

A slave shall recognize both levels, L0 and LS0, as a data bit value “0”. The LS0-level shall additionally qualify a deployment message as a “serious” one.

A slave shall accept a message in two cases:

- when all bits of content “0” are represented with L0-level;
- when all bits of content “0” are represented with LS0-level.

When some bits of a frame are represented with L0-level and others with LS0-level (= inconsistent safing), the message shall be ignored and regarded as a bus error (see 8.4.7). This gives additional protection against corrupted messages. Slaves, for which this extra level of protection is not needed, may not be able to receive

messages transmitted with safing. These slaves need not check for consistent safing and may not even be able to recognize an LS0-level at all.

A message of the **deploy command** family (see 8.4.2) that is sent **without** safing level, shall be accepted by the deploy slaves (i.e. not regarded as bus error), but a command to turn on a deploy switch shall be refused in this case.

A slave shall not reply in a D-Frame that has been started with safing level.

Master and slave on a deploy bus should process the safing information as far as possible away from the circuitry for handling the binary bit patterns of bus messages. Then the probability is low that a fault in a device corrupts both the bit pattern and the safing information at the same time.

6.7.2 Immunity to “babbling idiots”

When a parallel slave ASIC is shorted or a slave is transmitting without having been asked to do so (“babbling idiot”), a message sent by the master can get corrupted, since the defective slave can pull the L0-level down to L1-level, though the slave cannot modify the L1-level significantly. The master can recognize such a fault by reading back the bits while transmitting. The master shall then make the message artificially invalid to avoid the slaves receiving wrong data. An appropriate way to make a message invalid can be to send L1-level during the E-bit slot. Alternatively, the master can interrupt the faulty message by sending an SOF and starting a new frame. In any case, the master can send out a message with analogue safing level, e.g. a “serious” deploy message. Since a slave on the deployment bus cannot influence the L1-level or the LS0-level, the message is transmitted and received without bit error.

6.8 Bus signal parameters

Table 1 summarizes the bus signal parameters that are common for all implementations of the automotive safety restraints bus. Other parameters depend on the application. These are listed in Table 2, for a standard deployment bus and for a standard sensor bus. For non-standard applications, these parameters may be varied. A guideline for such variations is given in Annex B.

Table 3 lists the recommended cable parameters. Deviations are possible, depending on the actual bus speed and bus topology specifics.

Data bits shall consist of a sequence of Power and Data Phases with a Power Phase duty cycle of at least 50 % (see Figure 9). This ensures that the average surge current during the Power Phase is not much higher than twice the average supply current of the slaves. The nominal duration of one bit (= 100 %) shall be

- between 5,5 µs (181,8 kbit/s) and 14 µs (71,4 kbit/s) at high speed,
- between 22 µs (45,5 kbit/s) and 28 µs (35,7 kbit/s) at mid-speed,
- between 44 µs (22,7 kbit/s) and 56 µs (17,9 kbit/s) at low speed.

The devices may support other bit rates as well. A master implementation may support only a few discrete bus speeds. A slave implementation shall support all bus speeds within the above-specified speed ranges. A slave shall automatically adjust to the bus speed set by the master. Modules not requiring all three speed ranges may support less than these three speed ranges.

EXAMPLE A side-impact acceleration sensor may be optimized for high-speed operation and not support lower speeds.

The slew rate shall be high enough to guarantee a minimum data-signal dwell time below the respective receiver threshold of 15 % of one bit time or 1 µs, whichever is longer. The Data Phase itself (i.e. the time for which the bus voltage is lower than the V_{P0} threshold) shall be at least 30 % of one bit time. See also Figure 9.

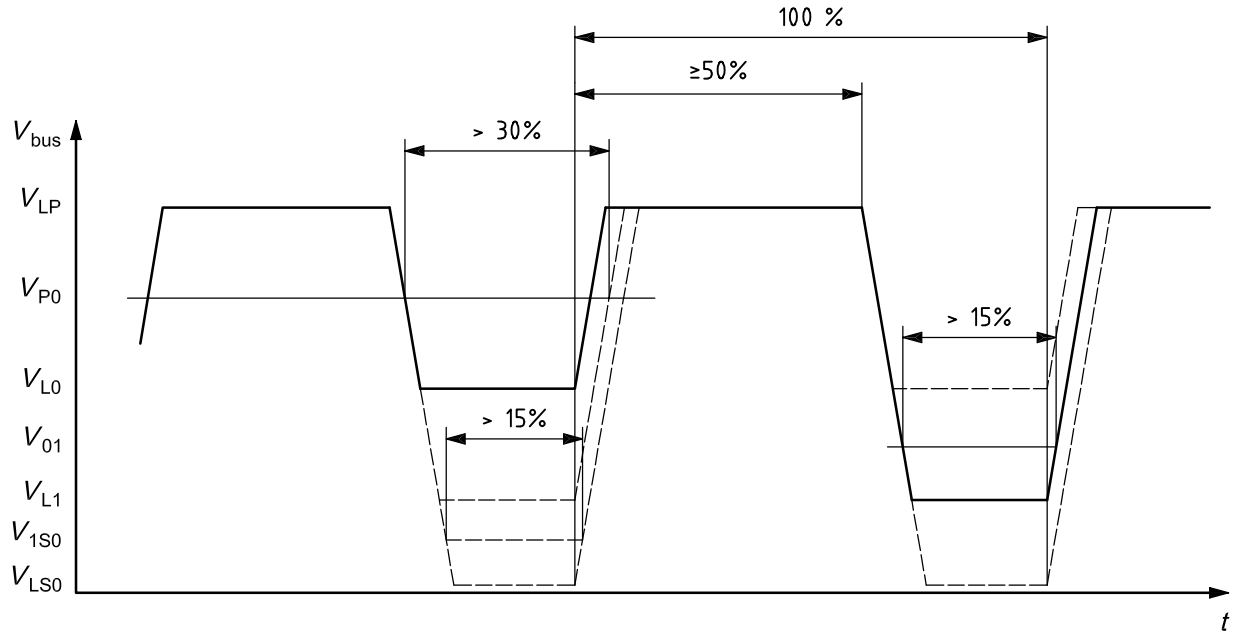


Figure 9 — Duty cycle of the Power and Data Phases

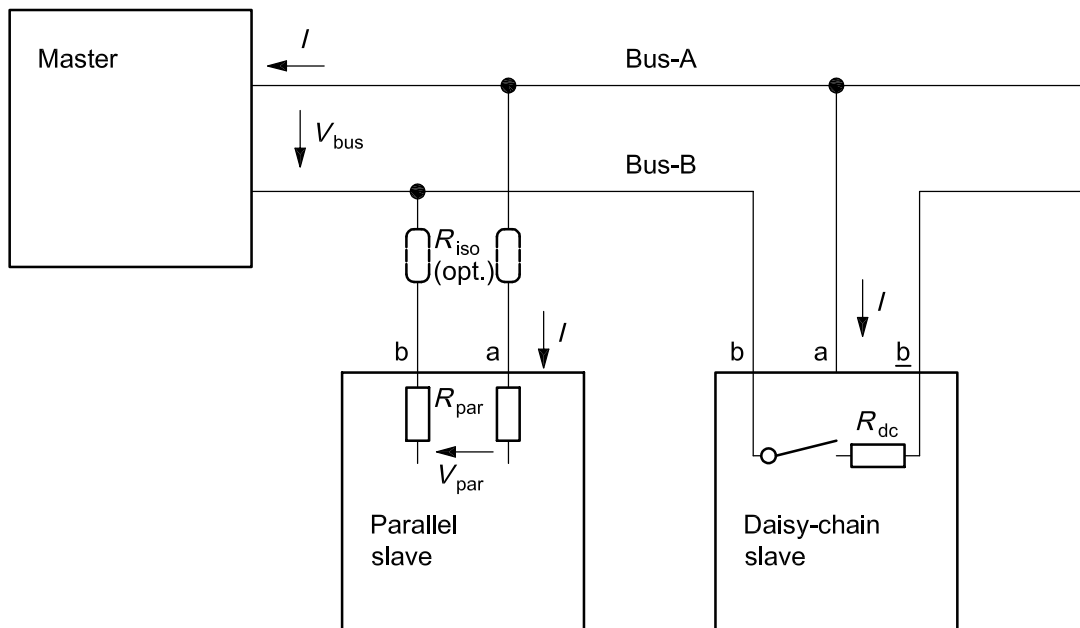


Figure 10 — Definition of parameters

Table 1 — Common bus parameters

Symbol	Parameter	Condition/remarks	Min.	Max.	Unit
V_{LP}	differential bus voltage at power level	normal operation OTP programming mode, max. 33 kbit/s	10,1 10,1	12,5 30	V V
V_{P0}	receiver threshold P/0		6,6	8,0	V
V_{L0}	differential bus voltage at data level 0		4,9	6,1	V
V_{01}	receiver threshold 0/1		3,6	4,4	V
V_{L1}	differential bus voltage at data level 1, master is transmitting		2,5	3,1	V
V_{L1slv}	differential bus voltage at data level 1, slave is transmitting (slaves with voltage transmitter, e.g. sensors replying to S-Frames)	measured at slave, $0 < I < 12$ mA, ($I_{Dsaf} = 0, R_{par} = 0, R_{iso} = 0$)	2	2,9	V
V_{1S0}	receiver threshold 1/S0		1,2	1,5	V
V_{LS0}	differential bus voltage at data level LS0	measured at transmitter (master or slave)	0	0,5	V
I_{Dsrc}	master data bias current	master sends L0-level, $V_{bus} < V_{L0}$	- 5	- 10	mA
I_{01}	D-Frame slave data current threshold		- 5	- 10	mA
I_{lkg}	tolerated leakage current during Data Phase	Bus-A to Bus-B and Bus to ground (bus isolation at master)		5	mA
I_{Dsaf}	master data safing current (only active when parallel deployable devices are on the bus)	master sends L0- or L1-level, $V_{bus} < V_{L1}$	- 20	- 40	mA
V_{par}	bus voltage at parallel slave ASIC while transmitting during D-Frames	$I = 12$ mA		0,9	V
I_{Dtxd}	slave data transmit current (daisy-chain deployable devices or sensors with current transmitter, i.e. sensors not replying to S-Frames)	$3,9$ V < V_{bus}	12	24 ^a	mA
		$2,5$ V < V_{bus} < $3,9$ V	0	24 ^a	mA
		V_{bus} < $2,5$ V measured at slave	0	1	mA
C_{mstr}	effective capacitance between Bus-A and Bus-B at master because of external components	one bus wire shorted to ground		5	nF
I_{tst}	test current during Power Phase for daisy-chain recovery from A-to-B shorts (used by master before switching on the bus, or used by slave before closing daisy-chain switch)	$V_{bus} < 5$ V	2,2	4,4	mA
		5 V < V_{bus} < $7,5$ V	0	4,4	mA
		$7,5$ V < V_{bus} measured at tested bus section	0	0,1	mA
t_{ko}	time-out for detection of an A-to-B short by a daisy-chain slave	time between end of previous Power Phase ($V_{bus} < V_{P0}$) and beginning of next Power Phase ($V_{bus} > V_{P0}$), measured at slave	60 ^b	240	µs
NOTE Unless specified differently, all bus voltages are measured at the master.					
^a The slave transmit current during the falling edge may be higher.					
^b Allows bus speeds down to 20 kbit/s with a tolerance of - 10 %. For non-standard, lower bus speeds, this number shall be increased accordingly.					

Table 2 — Application-specific parameters

Symbol	Parameter	Condition/remarks	Min.	Max.	Unit
Standard deployment bus					
I_{srg}	Surge current of a slave	$V_{\text{bus}} > 8 \text{ V}$ $V_{\text{bus}} < 8 \text{ V}$		4 ^a 0,1 ^b	mA mA
R_{par}	value of bus protection resistor at parallel slave	pure parallel bus or mixed parallel/daisy-chain bus, two resistors of this value are used	75 – 5%	75 + 5%	Ω
R_{iso}	value of bus protection resistor in harness between master and one parallel slave	pure parallel bus, two resistors of this value are used	75 – 5%	75 + 5%	Ω
R_{dc}	on-resistance of daisy-chain switch in slave			5	Ω
C_{ab}	capacitance of external components applied between pins a and b or a and <u>b</u> of a slave			250	pF
n_{par}	number of slaves in a pure parallel bus			16	1
n_{dc}	number of slaves in a in a pure daisy-chain or mixed parallel/daisy-chain bus			12	1
n_{dcmx}	number of daisy-chain slaves between master and parallel slaves in a mixed parallel/daisy-chain bus	reference configuration: MDPPDPPDPPDPP		4	1
$n_{\text{par dc}}$	number of parallel slaves between two daisy-chain slaves	M = master D = daisy-chain slave P = parallel slave		2	1
Standard sensor bus					
I_{srg}	surge current of a slave	$V_{\text{bus}} > 8 \text{ V}$ $V_{\text{bus}} < 8 \text{ V}$		45 ^a 0,1 ^b	mA mA
R_{dc}	on-resistance of daisy-chain switch in slave			5	Ω
C_{ab}	capacitance of external components applied between pins a and b or a and <u>b</u> of a slave			500	pF
n_{par}	number of slaves in a pure parallel bus		master shall be able to drive $n_{\text{par}} \times I_{\text{srg}}^{\text{c}}$		
n_{dc}	number of slaves in a pure daisy-chain bus			4	1
Extended sensor bus					
I_{srg}	surge current of a slave	$V_{\text{bus}} > 8 \text{ V}$ $V_{\text{bus}} < 8 \text{ V}$		30 ^a 0,1 ^b	mA mA
R_{dc}	on-resistance of daisy-chain switch in slave			3	Ω
C_{ab}	capacitance of external components applied between pins a and b or a and <u>b</u> of a slave			500	pF
n_{par}	number of slaves in a pure parallel bus		master shall be able to drive $n_{\text{par}} \times I_{\text{srg}}^{\text{c}}$		
n_{dc}	number of slaves in a pure daisy-chain bus			6	1
NOTE	“Non-standard” systems with different values are possible.				
a	For daisy-chain operation, this limit shall be implemented in the slave’s power extractor. See also Annex A.				
b	A slave shall not extract power during Data Phase.				
c	At high speed, further restrictions apply (depending on actual speed and on implementation of devices).				

Table 3 — Recommended cable parameters

Symbol	Parameter	Condition/remarks	Min.	Max.	Unit
l_{par}	total bus cable length for pure parallel configuration	sum of lengths of all branches		40 ^a	m
l_{ee}	cable distance between any two cable ends			25 ^a	m
l_{dc}	total bus cable length for pure daisy-chain or mixed parallel/daisy-chain configuration	sum of lengths of all branches		25 ^a	m
Z	specific cable impedance		95	140	Ω
C_c	cable capacitance	one bus wire shorted to ground		61	pF/m
^a At high speed, further restrictions apply (depending on actual speed and on implementation of devices).					

7 Data Link Layer

7.1 Bus Idle

Bus Idle shall be represented by a continuously repeated sequence of levels P-L0-P-L0-.... This ensures that a maximum latency time for a sensor interrupt can be specified.

During power-on, also a long Power Phase may be applied to the bus in order to accelerate power-on of the slaves.

7.2 Addresses

7.2.1 Slave address

The bus shall support 64 slave addresses, 0 through 63. However, the actual number of slaves that may be connected to one bus is limited by the supply current for the slaves, the pin capacitance of the slaves (see also Clauses 5 and 6) and the bandwidth. A single slave may incorporate the functionality of several slave addresses.

The slave address 0b000000 (0x00) should be reserved for slaves that have not yet been programmed with an address. Slave addresses shall be programmed into non-volatile memory in slaves with parallel configuration.

NOTE When slaves are used in daisy-chain configuration, their slave address can be stored in their volatile or non-volatile memory.

With the address a parity bit shall be stored (even parity). When a slave detects a parity error for its programmed address, it shall ignore the programmed address, set the “internal error” status (see Table 27) and adopt the reserved address 62 = 0x3e for communication with the master.

The slave address 63 = 0x3f shall be reserved for optional broadcast commands from the master. All slaves, independent of their actual programmed slave address, shall execute a command sent to slave address 63. All slaves shall ignore a message requesting data from slave address 63.

Usage of slave addresses is given in Table 4.

Table 4 — Usage of slave addresses

Slave address			Usage
decimal	hex	binary	
63	0x3f	11 1111	broadcast commands: all slaves shall accept commands sent to this address (optional feature)
62	0x3e	11 1110	address error: slaves that detect a parity error in their programmed address shall switch to this address
60 – 61	0x3c – 0x3d	11 1100 – 11 1101	for sensors
48 – 59	0x30 – 0x3b	11 0000 – 11 1011	for sensors or deployable devices
44 – 47	0x2c – 0x2f	10 1100 – 10 1111	for sensors
32 – 43	0x20 – 0x2b	10 0000 – 10 1011	for sensors or deployable devices
28 – 31	0x1c – 0x1f	01 1100 – 01 1111	for sensors
16 – 27	0x10 – 0x1b	01 0000 – 01 1011	for sensors or deployable devices
12 – 15	0x0c – 0x0f	00 1100 – 00 1111	for sensors
1 – 11	0x01 – 0x0b	00 0001 – 00 1011	for sensors or deployable devices
0	0x00	00 0000	default address of an unprogrammed slave

7.2.2 Memory address

Each slave has internal memory, which may consist of volatile and/or non-volatile (register, RAM, OTP, MTP or ROM) memory cells of 8-bit size. Each cell shall have a slave-specific **memory address**.

7.2.3 Signal address

A slave with Multi-Sharing capability (see 7.3.3.4) shall use an individual 6-bit **signal address** for each signal that it can transmit during S-Frames. Like the slave address, each signal address shall be unique for a given bus. When the slave serves only one signal, this signal address may be identical to the slave address of that slave.

The signal address shall have an attribute “sample” or “no sample”, depending on its numerical value (Table 5). The sampling shall be performed by successful acceptance of any Multi-Sharing address with the sampling attribute set to “sample”. This shall be used for synchronization of signal samples (see 7.3.3.6).

Table 5 — Signal address attributes

Signal address	Attribute
0x00 – 0x1f (i.e. MSB = 0)	sample
0x20 – 0x3f (i.e. MSB = 1)	no sample

Slave address, memory address and signal address of a sensor slave are illustrated in Figure 11.

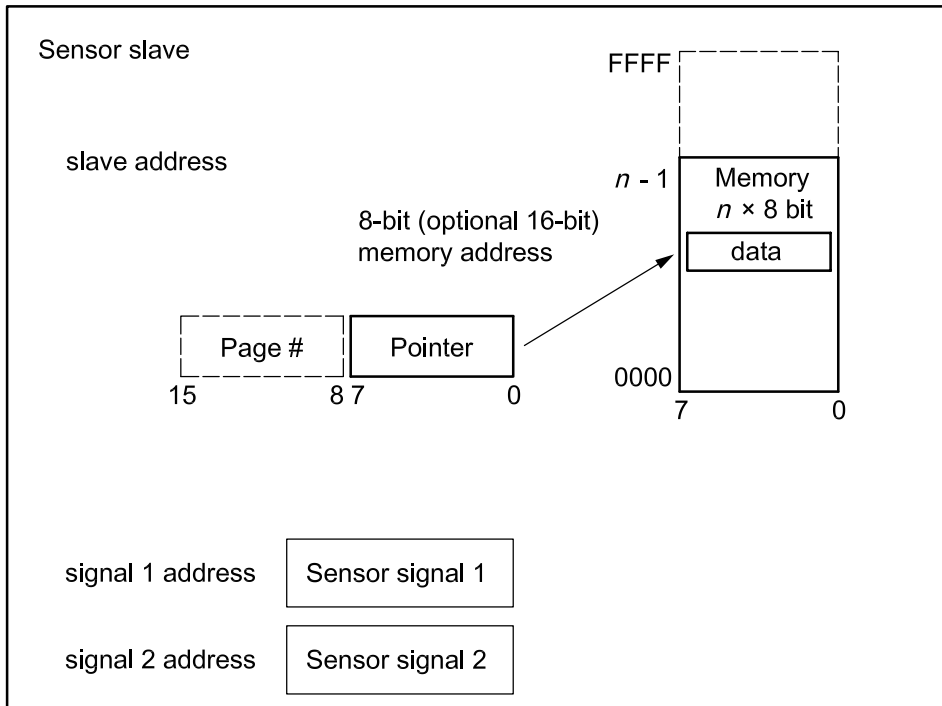


Figure 11 — Illustration of slave address, memory address and signal address of a sensor slave

7.3 Message frames

7.3.1 General

A message frame shall be transmitted exclusively by the master. Frames shall be used either to transfer data from the master to one or several slaves, or to transfer data from one or several slaves to the master. In the latter case, the master shall send the frame with L0-level at all bit positions that will contain slave data. The slave(s) shall overwrite these bits according to the data to be sent to the master.

A message frame shall start with an SOF symbol (see 6.5.2), which shall be transmitted by the master. A message shall end when the specified number of data bits of the frame has been transmitted. After the end of a frame, the master may either start a new frame or continue with the Bus Idle signal. An already started message frame can be cancelled by sending a new SOF symbol before completion of the frame.

There are two different message frame types available.

- The **D-Frame** shall be mainly used for diagnostic data and deploy commands.
- The **S-Frame** shall be mainly used for fast polling of sensor data (crash severity data or raw impact sensing data).

The data level within the SOF shall indicate the frame type ("T", see 7.4.2).

7.3.3 S-Frames

7.3.3.1 General

S-Frames have been defined for high throughput of sensor data to the master. The master shall start a frame with SOF and the appropriate T-bit (see 7.4.2). Then the slaves shall fill in their data in the respective slots of the frame, according to their set-up. Each slave shall append a CRC (see 7.4.9) to its data. Slaves shall modulate the bus voltage for data transmission.

In general, the S-Frame can support any number of slots, any length of data and individual data length for each sensor. However, the master shall know which sensor is using which data length and how many sensors are sending data within one frame. The data length should be eight or ten bits for raw-data sensors, or four bits for smart sensors sending crash severity levels (see Tables 7 and 8). For other sensors with short data length, see also 7.3.3.5. Accordingly, the bit number where the sensor's data slot starts shall be programmed into each slave. A master implementation should have a programmable number of slots, at least up to three slots.

The CRC-length used in S-Frames shall be either 3-bit or 8-bit. It shall be selectable as a global parameter for one bus.

EXAMPLE 1 A master implementation may be programmable to use the 3-bit CRC or the 8-bit CRC for all S-Frames.

EXAMPLE 2 A raw-data sensor slave implementation may support 3-bit CRC only.

EXAMPLE 3 Slaves sending data that does not need CRC protection may not send a CRC at all, which means that the CRC field may be left empty (all 0).

Table 7 — Example layout of an S-Frame with 4-bit crash severity data from three sensors

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
Header			Slot 0					Slot 1					Slot 2											
transmitted by master			transmitted by slave no. 0					transmitted by slave no. 1					transmitted by slave no. 2											
T	M	S	slave-0 data				CRC			slave-1 data				CRC		slave-2 data			CRC					
SEL																								
Slot bit no. (hex)			1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	10	11	12	13	14	15	

Table 8 — Example layout of an S-Frame with 8-bit raw-data from three sensors

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	31	32	33	34	35	36	37					
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	30	31	32	33	34	35	36						
transmitted by master										transmitted by slave no. 0																	
T		M S A		S E L		slave-0 data						CRC		transmitted by slave no. 2													

If an LS0-level is found within an S-Frame, the master shall not regard this as an interrupt.

7.3.3.2 Slot-Sharing

The field “SEL” shall select one out of two slaves sharing the same slot.

This way, sensors with fast sample rates can send their data with **each** S-Frame. Two sensors with half that sample rate can share the same slot in the S-Frame, with the SEL-bit controlling which of the two transmits.

EXAMPLE Sensors no. 1 and no. 2 are fast ones, no. 3, no. 4, no. 5 and no. 6 have half the sample rate, and the S-Frame has four slots.

First S-Frame, SEL = 1: no.1, no. 2, no. 3, no. 5.

Second S-Frame, SEL = 0: no. 1, no. 2, no. 4, no. 6.

Third S-Frame, SEL = 1: no. 1, no. 2, no. 3, no. 5.

And so on.

7.3.3.3 Collision-Signalling

The transmitting slave shall verify with its receiver circuitry that the right bus level actually appears on the bus. If this check has failed, the slave shall send LS0-level during the CRC field in order to notify the master about invalid data. This way the master can reliably detect multiple bit errors caused by overlay of two or more slaves erroneously transmitting during the same slot (“bus collision”). Slaves not requiring this additional error detection method may be implemented without collision signalling, i.e. without a transmitter stage for LS0-level.

7.3.3.4 Multi-Sharing

In addition to sharing one slot between two sensors (see 7.3.3.2), there is an option to share the first slot of an S-Frame between multiple sensors. Slaves supporting this Multi-Sharing option shall reply in the S-Frame only when addressed by the master. The master shall address (“call”) a Multi-Sharing slave by sending an S-Frame with MSA = 1 and sending (one of) the slave’s 6-bit signal address(es) in the first slot. After having been called successfully (i.e. received signal address matches (one of) its signal address(es) and CRC is valid), a Multi-Sharing slave shall be prepared to reply one time to an S-Frame, provided that the master has not called a different signal address in the meantime. The slave shall handle a matching signal address with invalid CRC like a non-matching signal address (see Figure 12, Table 10 and Table 12).

Multi-Sharing can be applied with full-rate (SEL ignored) or half-rate [slave addressing by the master and slave reply only for a certain value of SEL (see Table 9 and Table 11)].

When the master transmits the signal address, it shall leave the first two bits of the first slot empty (bit value 0), then transmit the slave address, MSB first. Accordingly, in a 10-bit slot the master shall leave the two bits after the signal address empty (bit value 0) as well.

Table 9 — Example assignment of six sensor signals to three slots

Signal	Slot no.	SEL-Filter	Multi-Sharing
A	0	yes, SEL = 0	no
B	1	no	no
C	2	yes, SEL = 0	no
D	0	yes, SEL = 1	yes
E	2	yes, SEL = 1	no
F	0	yes, SEL = 1	yes

Table 10 — Example sequence of S-Frames for configuration of Table 9

S-Frame count	MSA	SEL	Slot 0	Slot 1	Slot 2
1	0	0	Signal A	Signal B	Signal C
2	1	1	address of Signal D	Signal B	Signal E
3	0	0	Signal A	Signal B	Signal C
4	0	1	Signal D	Signal B	Signal E
5	0	0	Signal A	Signal B	Signal C
6	1	1	address of Signal F	Signal B	Signal E
7	0	0	Signal A	Signal B	Signal C
8	0	1	Signal F	Signal B	Signal E

Table 11 — Example assignment of six sensor signals to three slots

Signal	Slot no.	SEL-Filter	Multi-Sharing
A	0	no	yes
B	1	no	no
C	2	yes, SEL = 0	no
D	0	no	yes
E	2	yes, SEL = 1	no
F	0	no	yes

Table 12 — Example sequence of S-Frames for a configuration as in Table 11

S-Frame count	MSA	SEL	Slot 0	Slot 1	Slot 2
1	1	0	address of Signal A	Signal B	Signal C
2	0	1	Signal A	Signal B	Signal E
3	1	0	address of Signal D	Signal B	Signal C
4	0	1	Signal D	Signal B	Signal E
5	1	0	address of Signal A	Signal B	Signal C
6	0	1	Signal A	Signal B	Signal E
7	0	0	no reply ^a	Signal B	Signal C
8	1	1	address of Signal D ^a	Signal B	Signal E
9	1	0	address of Signal F	Signal B	Signal C
10	0	1	Signal F	Signal B	Signal E
11	0	0	no reply ^a	Signal B	Signal C
12	0	1	no reply ^a	Signal B	Signal E

^a These are examples illustrating the slave's behaviour when the master does not keep the usual order of messages.

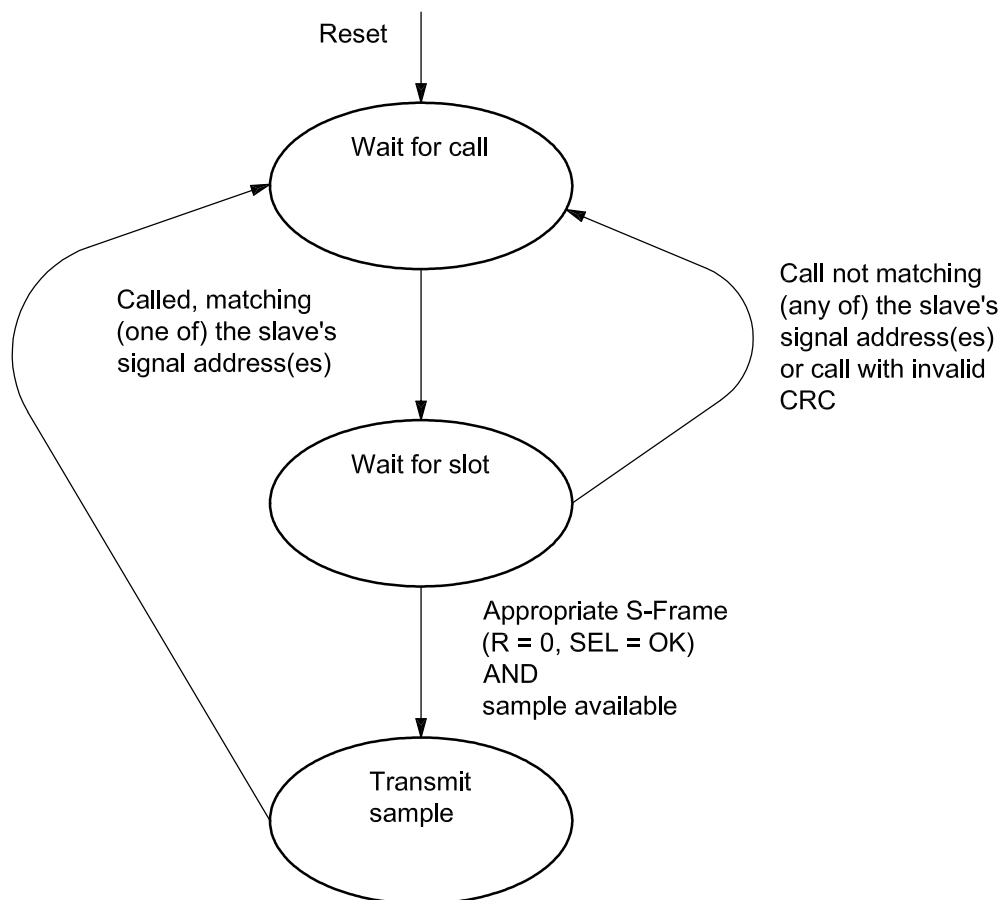


Figure 12 — State diagram for a slave in Multi-Sharing mode (see also Figure 13)

7.3.3.5 Sub-Slots

Slaves in Multi-Sharing mode may send a lower number of bits than available in the S-Frame slot. The slot may be divided into Sub-Slots, and each Sub-Slot shall be assigned to a different slave, while all these slaves shall be replying within the very same S-Frame slot (see Table 13, Table 15 and Table 16). Slaves sending in Sub-Slots shall not send anything in the CRC field of the slot. Usage of parity bits in a Sub-Slot is optional. The signal addresses for all slaves sending data in Sub-Slots within the very same S-Frame, shall have the same MSBs and differ only in terms of the LSBs (see Table 14). The master shall call all slaves assigned to the Sub-Slots of the S-Frame by sending the signal address of one of the slaves. All these slaves shall check only the MSBs of the signal address sent by the master for a match with their own signal address. The LSBs shall be ignored. The number of LSBs to be ignored may be 1, 2 or 3, depending on the number of slaves replying within the very same slot.

Table 13 — Example of an S-Frame with 2-bit Sub-Slots in an 8-bit slot

Header			Slot 0								Slot 1												
T	M S A	S E L																					
0	0	0/1	Sub-Slot 0	Sub-Slot 1	Sub-Slot 2	Sub-Slot 3	000		Data								CRC						
Slot bit no. (hex)			1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	10	11	12	13	14	15

Table 14 — Example: Four slaves using Sub-Slots

Slave	Signal Address	Signal addresses accepted for being called
A	0x10 = 010000	0x10 up to 0x13 = 0100XX (X = "don't care")
B	0x11 = 010001	
C	0x12 = 010010	
D	0x13 = 010011	

Table 15 — Example assignment of seven sensor signals to 3 slots

Signal	Slot no.	SEL-Filter	Multi-Sharing	Sub-Slots, no.
A	0	no	yes	yes, 0
B	0	no	yes	yes, 1
C	0	no	yes	yes, 2
D	0	no	yes	yes, 3
E	0	no	yes	no
F	1	no	no	no
G	2	no	no	no

Copyrighted material

Table 16 — Example sequence of S-Frames for a configuration as in Table 15

S-Frame count	MSA	SEL	Slot 0	Slot 1	Slot 2
1	1	0	address of Signal A	Signal F	Signal G
2	0	0	Signals A, B, C, D	Signal F	Signal G
3	1	0	address of Signal E	Signal F	Signal G
4	0	0	Signal E	Signal F	Signal G
5	1	0	address of Signal C	Signal F	Signal G
6	0	0	Signals A, B, C, D	Signal F	Signal G
7	0	0	no reply ^a	Signal F	Signal G
8	1	1	address of Signal A	Signal F	Signal G
9	0	1	Signals A, B, C, D	Signal F	Signal G
10	1	0	address of Signal A ^a	Signal F	Signal G
11	1	0	address of Signal E	Signal F	Signal G
12	0	0	Signal E	Signal F	Signal G
13	0	0	no reply ^a	Signal F	Signal G

^a These are examples illustrating the slave's behaviour when the master does not keep the usual order of messages.

7.3.3.6 Sensor sampling synchronization

When processing sensor data, it is desirable to have synchronous sampling by all sensors belonging to the same function, or at least to know at which times the individual samples have been taken.

Slaves shall take samples of time-variant signals (to be transmitted within S-Frames) right after the SOF of an S-Frame. This means that the data of all slots in the same S-Frame would represent samples taken at the same point of time. Exception: An implementation with one A/D converter and several analogue inputs connected via a multiplexer converts one signal after the other, so that there would be a certain delay between the sample points of the individual signals.

When slaves belong to the same function (e.g. four strain gauge sensors for occupant weight sensing) and are sharing one slot in Multi-Sharing mode, their data shall be transferred via several S-Frames. Nevertheless, their sample points can be synchronized. This is achieved by taking (a) new sample(s) only with the first S-Frame **after** an S-Frame calling a Multi-Sharing signal address with attribute "sample" (see 7.2.3), whether the signal address matches its own signal address or not. See Figure 13.

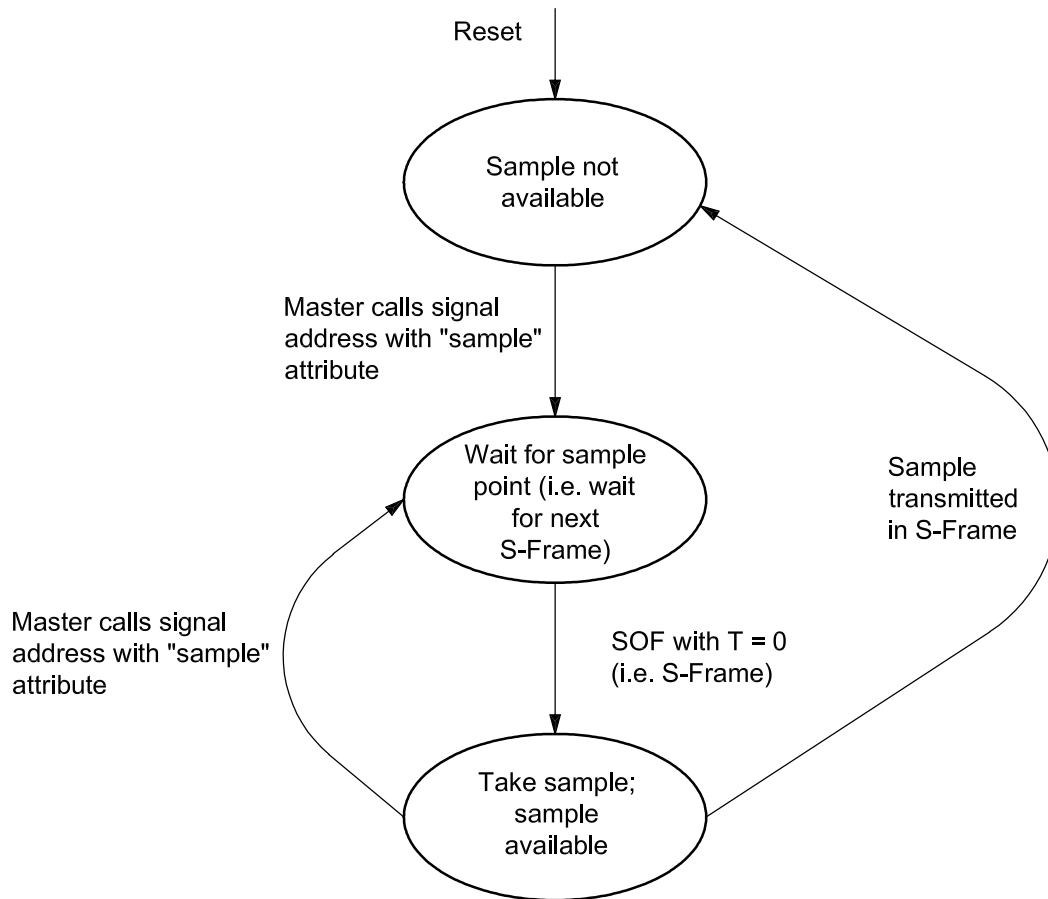
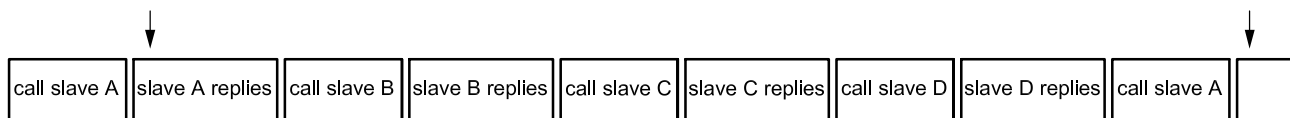


Figure 13 — State diagram for signal sampling by a slave in Multi-Sharing mode

Figure 14 shows an example sequence of S-Frames with Multi-Sharing where only the address of slave A has the attribute “sample”.



NOTE The arrows indicate the sample points of all slaves A – D.

Figure 14 — Example sequence of S-Frames with Multi-Sharing where only the address of slave A has the attribute “sample”

7.4 Bit fields within a frame

7.4.1 Bit order

Within all bit fields of two or more bits, the MSB is transmitted first.

7.4.2 T (Type)

The T-bit shall be a single bit defining the type of the frame (see Table 17). It shall be sent as an SOF symbol, which usually needs as much time as two normal data bits. Therefore the T-bit occupies two columns in Table 6, Table 7 and Table 8.

Table 17 — Meaning of T-Bit-level

Value of T	Meaning
0	This is an S-Frame.
1	This is a D-Frame.
S0	This is an S-Frame that has been initiated by an interrupt during SOF, or that has been started by the master before finishing a running frame.

When an interrupt occurs during SOF, then automatically an SOF for an S-Frame shall be generated. If the master was intending to send an S-Frame anyway, then the interrupt shall not change anything. If the master was intending to send a D-Frame, then the master shall recognize the interrupt and switch over to an already started S-Frame, i.e. it shall continue the frame, as if it would have wanted to send an S-Frame anyway.

The master can stop a D-Frame at any time by starting a new frame, without having to take into account pending slave data, which would overwrite its SOF. The “1” in the SOF of a new D-Frame cannot be overwritten by slave reply data (0 or 1), and the master uses the “LS0” for indication of a new S-Frame that cannot collide with pending slave data either. See also 6.5.2.

7.4.3 R (Reserved)

This bit has been reserved for future extensions of the bus system. For the D-Frames described in this International Standard, the master shall transmit it as a “0”. Slaves shall accept D-Frames only when they receive a “0” (i.e. L0-level or LS0-level) at this bit position. Slaves shall ignore D-Frames with R = 1, but this shall be not regarded as a bus error.

7.4.4 MSA (Multi-Sharing Address)

In S-Frames, the master shall send either MSA = 0 or MSA = 1. Slaves without Multi-Sharing shall ignore the MSA-bit and shall reply only based on the SEL-bit. When the master sends an S-Frame with MSA = 1, it shall send a signal address in the first slot and all slaves in Multi-Sharing mode shall then check if this signal address matches (one of) their own signal address(es). Slaves in Multi-Sharing mode (see 7.3.3.4) shall reply only in S-Frames with MSA = 0 and only after the appropriate signal address has been transmitted.

7.4.5 Command

This 4-bit field shall provide 16 different commands. These are specified in Clause 8.

7.4.6 Slave address

This field shall specify the node address of the slave involved in this frame’s communication. See also 5.2, 7.2 and 7.3.2.

7.4.7 Address MSBs

This field shall specify the two most significant address bits of the slave addresses that are selected in the following slave address bitmap.

7.4.8 Slave address bitmap

For D-Frames with bitmapped addressing, this field shall not specify a single slave address. Depending on the value in the field “Address MSBs”, it shall specify a bitmap for slave addresses 0x0 to 0xB, 0x10 to 0x1B, 0x20 to 0x2B or 0x30 to 0x3B. The layout of the bitmap is indicated in Table 6.

NOTE A single multi-stage deployable device can be controlled by more than just one bit in the bitmap (see also 8.3.2.1).

7.4.9 CRC

In **D-Frames**, the 8-bit CRC shall cover all preceding 20 bits of the frame, including the T-bit which was sent during SOF and the reserved bit R-bit. For data sent by the master, the master shall send the CRC of the frame. For data sent by a slave, the slave shall send the CRC (which covers the whole frame), including the T-bit, the R-bit and the command and address bits, which have been sent by the master. The generator polynomial of the 8-bit CRC shall be

$$x^8 + x^4 + x^3 + 1,$$

which shall include a parity check. The start value for the CRC shall be 0xff. The CRC shall be transmitted MSB first.

EXAMPLE 1

- A D-Frame (T = 1, R = 0) with all command, address and data bits being “0” carries the CRC 0xda.
- A D-Frame (T = 1, R = 0) with all command, address and data bits being “1” carries the CRC 0x19.

For D-Frames with bitmapped slave response, the master shall send out 0x00 for the CRC, but slaves shall not send a CRC. The CRC shall be “don’t care” for master and slaves in this case.

In **S-Frames**, the 3-bit CRC shall be a checksum for the T-bit, the R-bit, the SEL-bit and the according data bits sent by the respective slave, or for the T-bit, the MSA-bit, the SEL-bit and the signal address sent by the master (in the first slot, when R = 1). The generator polynomial shall be

$$x^3 + x + 1.$$

The start value of the CRC shall be 0x07 (This start value ensures that data = 0 does not produce CRC = 0). The CRC shall be transmitted MSB first.

EXAMPLE 2

- 8-bit S-Frame reply data (T = 0) with MSA, SEL and all eight data bits being “0” carries the 3-bit CRC 0x4.
- 10-bit S-Frame reply data (T = 0) with MSA, SEL and all 10 data bits being “1” carries the 3-bit CRC 0x0.

Alternatively, S-Frames can use the 8-bit CRC that is known from D-Frames (polynomial and start value described above). Like the 3-bit CRC, it shall cover T-bit, MSA-bit, SEL-bit and the data bits sent by the slave.

During S-Frames, each slave shall monitor the data it is transmitting bit-by-bit. If there is a mismatch between the bit value found on the bus and the one it was intended to transmit (= “collision”), the slave may not append a CRC to its data but transmit the LS0-level instead in all bits of the CRC field (= “collision signalling”, see also 7.3.3.3). The presence of one or more LS0-bits in the CRC field shall tell the master that the data of that slot is invalid.

Slaves shall not care about the CRC transmitted by other slaves and shall transmit data in their respective slots regardless of the CRC field contents of previous slots. This implies that the master shall continue with the S-Frame after such error.

The “start value of the CRC” shall be the CRC result for n bits with $n = 0$. For implementations of the CRC generator where the CRC register directly reflects the current CRC result after a data bit has been shifted in, the “start value of the CRC” shall be equivalent to the “start value of the CRC register”.

Further CRC examples are listed in Annex E.

7.4.10 E (Error)

This bit shall be used within D-Frames by the master to indicate any detected mismatch between the data seen on the bus, and the data that it actually intended to transmit. Such mismatch shall be indicated as an error by sending a “1” at the E-bit position. Slaves shall ignore commands when the E-bit is “1”. For all D-Frames that are used to get data from slaves, the E-bit shall be “1”. This ensures that other slaves do not mix it up with a command message.

NOTE 1 The addressed slave would have sent its data before the E-bit was received.

NOTE 2 A shorted squib on a parallel bus would turn all L0-level into L1-level, including the E-bit. Deploy messages with safing can still be sent, since they use the LS0-level instead of the L0-level in order to indicate a valid message. The master can indicate a faulty deploy message by transmitting L1-level at the E-bit position.

For D-Frames requesting slave data, the master shall ignore the slave data if the master detects data transmission by slaves during the command and address bits.

8 Application Layer

8.1 General

The Application Layer shall support the architecture of slaves with all slave resources being accessible via the bus through memory in the slave (see also 7.2.2). The master can access the memory of a slave by first setting the memory pointer of the slave and then reading or writing the data. This can be executed with a sequence of D-Frames, using the following commands:

- Write Page Number - optional for slaves with memory size > 256 bytes;
- Read Page Number - optional for slaves with memory size > 256 bytes;
- Write Pointer;
- Read Pointer;
- Write Memory;
- Read Memory.

These commands shall all use point-to-point addressing of the slaves. For faster access to frequently used control or status bits, three short-cut commands shall be available that do not need prior setting of the pointer:

- Status Change - controlling the slave;
- Read Status 1 - reading status or error information from the slave;
- Read Status 2 - reading status or error information from the slave.

These commands shall also use point-to-point addressing of the slaves.

There shall be seven commands with bit-mapped addressing for exclusive use with deployable devices (see 8.4). Table 18 lists the various commands within D-Frames.

Table 18 — Command set for deployable devices

Code	Command	Type	DD ^a	Remarks
0000	No Deploy	bitmapped	W	The command code "00xy" requests a change in status of the two deploy switches HSD and LSD of a deployable device (see also Figure 15), where "x" refers to HSD and "y" refers to LSD, with "1" and "0" requesting to switch on and off, respectively.
0001	Test LSD	bitmapped	W	
0010	Test HSD	bitmapped	W	
0011	Deploy	bitmapped	W	
0100	Deploy Enable	bitmapped	W	
0101	Enable Status	bitmapped response	R	
0110	Deploy Status	bitmapped response	R	
0111	Write Page Number	point to point	W	Devices with less than 257 bytes of memory may use this command differently.
1000	Read Status1	point to point	R	
1001	Read Status2	point to point	R	
1010	Status Change	point to point	W	
1011	Read Page Number	point to point	R	Devices with less than 257 bytes of memory may use this command differently.
1100	Write Pointer	point to point	W	
1101	Read Pointer	point to point	R	
1110	Write Memory	point to point	W	
1111	Read Memory	point to point	R	

^a Data Direction: W = Write, i.e. from master to slave(s); R = Read, i.e. from slave(s) to master.

8.2 Common D-Frame commands

8.2.1 "Write Pointer" message

The internal memory of a slave shall be accessible by indirect addressing only. Two messages are necessary in order to do one operation on one memory address. With the data field of the "Write Pointer" message the address is specified for which the memory operation shall be executed. The memory operation itself shall be requested with a following "Write Memory" or "Read Memory" message.

Indirect addressing has the advantage that it supports a big memory size of 256×8 bits by keeping message lengths the same as other messages, thus reducing complexity in the communication ASICs.

The addressing capabilities can even be extended to cover memory sizes of up to 64 kbyte, see 8.2.5.

8.2.2 "Read Pointer" message

The slave shall tell the master which memory address is currently selected.

8.2.3 "Write Memory" message

The data field holds the new value to be written to the memory location at the selected memory address.

8.2.4 "Read Memory" message

The slave shall transmit the actual value of the memory location at the selected memory address. If the data is its slave address, it shall send the address and the parity bit of the address (see 7.2 and 8.3.2.2).

8.2.5 “Write Page Number” message

Some slaves may have more than 256 bytes of memory. The “Write Page Number” message shall allow 16-bit addressing to be used, if available by the slave. The data field of this message shall hold the 8-bit page number of the slave memory. The 16-bit memory address shall comprise the 8-bit page number and the 8-bit address (see also Figure 11), the latter being selected with the “Write Pointer” message (see 8.2.1). For slaves with less than 257 bytes of memory, this message may be used for device-specific non-standard commands to the slave.

8.2.6 “Read Page Number” message

The slave shall tell the master which memory page number is currently selected. For slaves with less than 257 bytes of memory, this message may be used for reading device-specific non-standard data from the slave.

8.3 Memory layout of slaves

8.3.1 General

The internal memory of a slave shall be organized in $n \times 8$ bits. A memory location (i.e. one byte) is also referred to as “register”. The contents of the memory shall be structured into three sections.

- Section I: slave identification and set-up data with a common layout for all slaves (see 8.3.2).
- Section II: module-specific set-up and calibration data and control registers, intended for use by the application (see 8.5.2).
- Section III: implementation-specific registers, not intended for use by the application (not covered by this International Standard).

8.3.2 Slave memory Section I

8.3.2.1 General

All slaves should have the same layout for the lower part of their internal memory (Section I, address range 0x00 to 0x2f). This part of the memory shall define the slave address, traceability data and the slave’s configuration for S-Frame service and bus interrupt. Table 19 lists its contents (R = readable over bus, W = can be written or programmed, NVM = non-volatile memory, ROM = read-only memory, RAM = random access memory).

Table 19 — Layout of the lower part of a slave’s internal memory (Section I)

Addr. (hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R/W ^a	Memory type (typical)
00	Slave Address Configuration								R/W	NVM
	Lock NVM	Parity	Slave Address							
01	IC Device Type								R	ROM
02	IC Manufacturer ID								R	ROM
03	IC Revision Level								R	ROM
04	Module Type L								R/W	NVM
05	Module Type H								R/W	NVM
06	Module Manufacturer ID								R/W	NVM
07	Module Revision Level								R/W	NVM
08	Serial Number Byte 0 (LSBs)								R/W	NVM
09	Serial Number Byte 1								R/W	NVM
0a	Serial Number Byte 2								R/W	NVM
0b	Serial Number Byte 3								R/W	NVM
0c	Serial Number Byte 4 (MSBs)								R/W or R	NVM or ROM
0d – 0f	Reserved (3 bytes)									
10	Slave Status 1 (Error Status) 0x00 = no error all other values = error (details are module specific)								R	VM
11 – 1f	Reserved (15 bytes)									
20	Signal 0 Assignment to S-Frames								R/W	NVM
	MSA									
	0	S-Frame Slot Start Bit Number								
	1	SSB								
		0	Signal Address for Multi-Sharing							
1	Slave Address Mask Length 00 = no mask 01 = 1 LSB 10 = 2 LSBs 11 = 3 LSBs	Sub-Slot Start Bit Number 0000 = slot not used 0001 = Sub-Slot starts at bit no.1 ... 1010 = Sub-Slot starts at bit no.10 1011 – 1111 = slot not used								
21	Signal 1 Assignment to S-Frames								R/W	NVM
	MSA									
	0	S-Frame Slot Start Bit Number								
	1	res	Slave Address for Multi-Sharing							
22	Signal 2 Assignment to S-Frames (see Signal 1...)								R/W	NVM

Table 19 (continued)

Addr. (hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R/W ^a	Memory type (typical)
23	Signal 3 Assignment to S-Frames (see Signal 1...)								R/W	NVM
24	Signal 4 Assignment to S-Frames (see Signal 1...)								R/W	NVM
25	Signal 5 Assignment to S-Frames (see Signal 1...)								R/W	NVM
26	Signal 6 Assignment to S-Frames (see Signal 1...)								R/W	NVM
27	Signal 7 Assignment to S-Frames (see Signal 1...)								R/W	NVM
28	Signal 0&1 Configuration								R/W	NVM
	Signal 1				Signal 0					
	Data Length		SEL Filter		Data Length		SEL Filter			
	00 = 8 bit		00 = ignore SEL		00 = 8 bit		00 = ignore SEL			
	01 = 10 bit		01 = ignore SEL		01 = 10 bit		01 = ignore SEL			
10 = 4 bit		10 = [SEL=0] only		10 = 4 bit		10 = [SEL=0] only				
11 = reserved		11 = [SEL=1] only		11 = reserved		11 = [SEL=1] only				
29	Signal 2&3 Configuration (see Signal 0&1...)								R/W	NVM
2a	Signal 4&5 Configuration (see Signal 0&1...)								R/W	NVM
2b	Signal 6&7 Configuration (see Signal 0&1...)								R/W	NVM
2c	Slot Configuration								R or R/W	ROM or NVM
	CRC-length	res	res	Bus-INT	res	Sub-Slot Length				
	0: 3 bits			0 = no		000 = no Sub-Slots				
	1: 8 bits			1 = yes		001 = 1 bit				
						010 = 2 bits				
						...				
						111 = 7 bits				
2d	S-Frame Length (only for sensors with bus interrupt capability)								R/W	NVM
2e-2f	Reserved (2 bytes)									

^a When NVM is locked, it can no longer be written. OTP memory can be written (i.e. programmed) only once.

8.3.2.2 Slave Address Configuration

8.3.2.2.1 General

This memory location shall contain the slave address under which this slave can be accessed using D-Frames with point-to-point addressing. The slave address shall also determine the relevant bit position in the bitmap of a D-Frame with bitmapped addressing (see 7.4.8).

NOTE When, for example, a multi-stage deployable device is controlled by more than one bit in the bitmap, this corresponds to an assignment of several slave addresses to that slave, of which only one slave address (the one that is intended to be used for D-Frames with point-to-point addressing) would be stored in this memory location. The other slave addresses would either be defined implicitly (i.e. an n -stage device would occupy n addresses, starting with the one stored in this location) or explicitly (e.g. a list of used addresses, stored in Memory Section II).

The default slave address (i.e. when the slave has not yet been programmed) shall be 0x00. See also 7.2.1.

8.3.2.2.2 Address Parity

This bit shall be “0” when the slave address consists of an even number of bits of value “1”. The parity bit shall be programmed together with the slave address. When the value of the parity check does not match the slave address, this shall be regarded as an internal error and the slave shall adopt the reserved slave address 62.

8.3.2.2.3 Lock NVM

When this bit is programmed to “1”, the contents of the NVM shall be frozen.

8.3.2.3 IC Device Type

This byte shall be defined by the manufacturer of the integrated circuit.

8.3.2.4 IC Manufacturer ID

This byte shall identify the manufacturer of the integrated circuit (see also Annex G).

8.3.2.5 IC Revision Level

This byte shall be defined by the manufacturer of the integrated circuit.

8.3.2.6 Module Type L and Module Type H

This shall be a 16-bit value identifying the module’s function in the car.

8.3.2.7 Module Manufacturer ID

This byte shall identify the module manufacturer.

8.3.2.8 Module Revision Level

This byte shall be defined by the manufacturer of the module.

8.3.2.9 Serial Number Byte 0–4

The serial number shall consist of five bytes. Byte 0 shall contain the LSBs.

NOTE Several billion devices may have the same contents in Byte 4, so that a ROM implementation of Byte 4 may be considered.

8.3.2.10 Slave Status 1

The Slave Status 1 Register can also be read with the short-cut D-Frame command “Read Status 1”. It should contain error bits only. The other details of this register are implementation specific.

8.3.2.11 Signal 0–7 Assignment to S-Frames

A sensor slave may have up to eight signals that can be transmitted in S-Frames. For each signal one of these registers may define the signal’s slot assignment. If a signal does not exist, the corresponding register content shall be 0x00. Accordingly, for a device that does not reply to S-Frames at all, these register contents shall be zero (0x00).

The MSA-bit shall determine whether the slave transmits this signal in Multi-Sharing (see 7.3.3.4) or in normal mode.

8.3.2.16 Configuration check sum

The memory locations defining the S-Frame configuration and other data critical for a proper system set-up shall be protected by appropriate means, e.g. a check sum, so that single bit failures of the memory can be detected. The slave shall permanently check for such bit failures and in case of a failure it shall not reply to S-Frames and shall not send bus interrupts.

8.4 Application Layer for deployable devices

8.4.1 General

Application Layer specifics for deployable devices (see also Annex F) are specified in 8.4.2 to 8.4.8, which may also be applicable to actuators in general, with “deploy” meaning “actuate”. Deployable devices shall use D-Frames exclusively. They shall ignore S-Frames.

8.4.2 Deploy command family

The four commands “No Deploy”, “Deploy”, “Test LSD” and “Test HSD” shall be treated as one command where the two least-significant command bits (LSBs) define the requested operation of two deploy switches (Table 21), a low-side driver (LSD) and a high-side driver (HSD), see example in Figure 15. The “Test LSD” and “Test HSD” commands shall allow diagnostic testing of the individual switches by a method that is similar to a deployment command.

Table 21 — Deploy command family

Code	Meaning
0000	Switch off both HSD and LSD (= No Deploy)
0001	Switch off HSD, switch on LSD (used to Test the LSD)
0010	Switch on HSD, switch off LSD (used to Test the HSD)
0011	Switch on both HSD and LSD (used to Deploy)

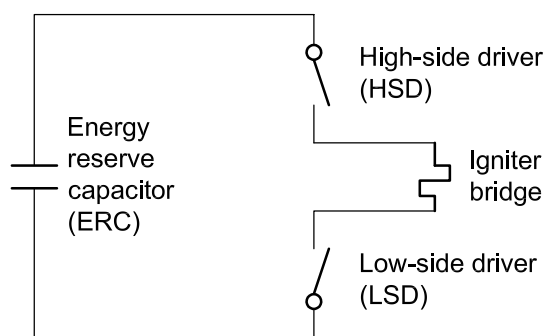


Figure 15 — Example of a deployable device with two deploy switches

These messages shall use the address and data fields together as a 12-bit bitmap for 12 deployable devices as defined in Table 22. The code for this command shall be 00xy. The two LSBs “x” and “y” shall refer to the two deploy switches HSD and LSD. A “1” shall indicate that the respective switch shall be switched on. A “0” shall indicate that the respective switch shall be switched off.

Deploy switches shall not be switched **on** unless

- deployment has been enabled before (see 8.4.3),
- the “Deploy” command has been transmitted with safing level (LS0) for all “0”s in the message,
- the CRC matches the data,
- the E-bit shall be “0”.

Deploy switches shall be automatically switched **off**, when

- an according command has been received with or without safing level,
- a “Deploy Disable” command has been received.

Execution of the “Test LSD” or “Test HSD” shall be executable only if the device is not ready to fire. Execution of such a command shall be in the following order:

- a) open the other switch, if it was closed;
- b) close the switch under test.

Table 22 — Value assignment for the bitmap of a “No-Deploy”, “Deploy”, “Test LSD” or “Test HSD” command message

Value	Meaning
0	Execute the command
1	Do not execute the command

For resettable actuators, the “No Deploy” command may be used to reset the actuator.

A single multi-stage deployable device should be controlled with more than one bit in the bitmap.

8.4.3 “Deploy Enable” message

This message shall use bitmap addressing for 12 deployable devices. The bit value assignment shall be according to Table 23. The “Deploy” and “Deploy Enable” messages shall use opposite value assignments for the bitmap. This means that typically the “Deploy” message shall carry the **inverted** bitmap of the “Deploy Enable” message in order to cause deployment of the selected squibs. In turn, any sequence of a “Deploy Enable” and “Deploy” message with **identical** bitmap can not cause any deployment.

Table 23 — Value assignment for the bitmap of a “Deploy Enable” message

Value	Meaning
0	Deploy Disable
1	Deploy Enable

8.4.4 “Enable Status” message

This message shall use bitmap addressing for the reply of 12 deployable devices. The slaves shall send their deploy status in the corresponding slot within the address bitmap. The bit value assignment shall be according to Table 23.

8.4.5 “Deploy Status” message

This message shall use bitmap addressing for the reply of 12 deployable devices. The slaves shall send their deploy status in the corresponding slot within the address bitmap. The value assignment shall be according to Table 24. The assignment shall be made in such a way that a non-responding slave produces the same data as a responding slave, indicating that it did not execute a deploy command. This is opposite polarity to that used for deploy commands.

Table 24 — Value assignment for the bitmap in the “Deploy Status” message

Value	Meaning
0	No deploy command executed
1	Deploy command executed

The deploy status shall reflect the status of the control lines to the deploy switches. When both control lines are in the state to switch on the respective switch, the value of the deploy status shall be “1”. In all other cases the value of the deploy status shall be “0”.

8.4.6 “Status Change” message

This message shall be transmitted to a slave in order to change its internal state. The requested state shall be coded in the data field (see Table 25).

Activation of the “High-side Driver Status Test” or “Low-side Driver Status Test” shall not cause any operation of the switches. It shall only apply certain test currents to the switches in order to be able to evaluate the actual position of the switches, on or off.

The commands “Squib Resistance Test”, “High-side Driver Status Test” and “Low-side Driver Status Test” may not be necessary, if the device performs these actions automatically, when a “Read Status 2” message is processed (see 8.4.8).

Table 25 — Example for the data byte layout of a “Status Change” message to a deployable device

Bit no.	Squib status	Value assignment
7	ERC	1 = charge 0 = discharge
6	Squib Resistance Test (optional)	1 = do test 0 = no test
5	High-side Driver Status Test (optional)	1 = do test 0 = no test
4	Low-side Driver Status Test (optional)	1 = do test 0 = no test
3	Reserved	
2	Reserved	
1	Reset Error Status (see 8.4.7)	1 = reset 0 = no reset
0	Daisy-chain switch (if available)	0 = off = open 1 = on = close

NOTE In some cases, the new status may be kept by the slave for a restricted time only. For instance, the “Squib Resistance Test” request may be executed by the squib and the squib would automatically return to “no test” after successful execution of the test.

8.4.7 “Read Status 1” message

All functional blocks relevant for deployment shall be covered by diagnostics.

The “Read Status 1” message shall be used to retrieve the actual primary status of the slave. The data field of the frame shall be transmitted by the slave (see Table 26).

Table 26 — Example for the data byte layout of a “Read Status 1” message to a deployable device

Bit no.	Squib status	Value assignment
7	HSD deploy switch status ^a	1 = on 0 = off
6	LSD deploy switch status ^a	1 = on 0 = off
5	Error level bit 1	see Table 27
4	Error level bit 0	see Table 27
3	Daisy-chain switch status ^a	1 = on 0 = off
2	Deploy Enable Status	1 = deployment enabled 0 = deployment disabled
1	ERC all-fire level	1 = all-fire 0 = not all-fire
0	ERC no-fire level	1 = no-fire 0 = not no-fire
^a This is more the status of the control lines to the switch(es) than the actual status of the switch(es).		

The error level bits 0 and 1 shall form a two-bit error level number, which shall indicate the quality of bus communication (see Table 27). The device shall latch the highest error level number that has been entered during operation. The error level can be reset to “0” by the “Reset Error Status” command of a “Status Change” message (see 8.4.6). At power-on, the device shall start with error level 0.

Table 27 — Definition of error level bits

Error level bit 1	Error level bit 0	Meaning	Explanation
0	0	Error level 0 (ok)	normal operation
0	1	Error level 1 (noise info)	a CRC error ^a has been detected, or a message with both L0 and LS0 level has been detected
1	0	Error level 2 (warning)	a deploy command ^b has been received without safing level
1	1	Error level 3 (error)	an internal error has been detected, e.g. address parity error (see also 7.2)
^a A CRC error is regarded as error only if the E-bit is “0”.			
^b Command code 0011.			

8.4.8 “Read Status 2” message

This message shall be used to retrieve the actual secondary state of the slave. The data field of the message shall be transmitted by the slave.

Table 28 — Example for the data byte layout of a “Read Status 2” message to a deployable device

Bit no.	Squib status	Value assignment
7	High-side driver ^a	1 = on 0 = off
6	Low-side driver ^a	1 = on 0 = off
5	Reserved	
4	Reserved	
3	Bridge resistance 1	00 = ok
2	Bridge resistance 0	01 = too high 10 = too low 11 = not tested
1	Bridge leakage 1	00 = no significant leakage detected
0	Bridge leakage 0	01 = leakage to Low detected 10 = leakage to High detected 11 = not defined

^a This is the actual status of the switches, which is only valid when the appropriate test signals have been turned on, see “Status Change” command in 8.4.6

For some of the status bits in this register, it may be necessary to turn on internal test signals. Depending on the implementation, it may be required to activate these tests with an appropriate “Status Change” message before reading (see 8.4.6). Other implementations may activate these tests automatically upon reception of a “Read Status 2” command. In that case they would reply with the status that was evaluated during a previous “Read Status 2” message and at the same time they would do a new test. The result of this test would then be sent with the next “Read Status 2” message (see Figure 16).

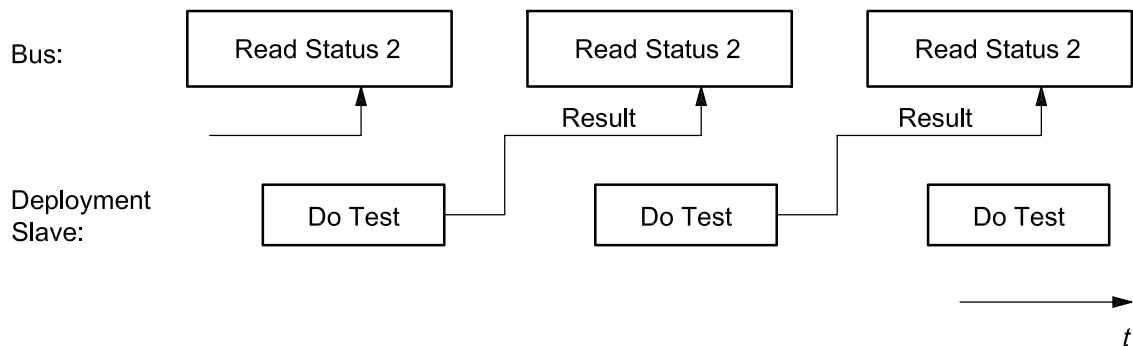


Figure 16 — Illustration of automatic execution of diagnostic tests upon reception of “Read Status 2” messages

8.5 Application Layer for sensor devices

8.5.1 General

Application Layer specifics for sensors are specified in 8.5.2 to 8.5.4. Sensors shall use the D-Frame messages described in 8.2. The meaning of the data byte in “Read Status 1” / “Read Status 2” and “Status Change” messages is sensor specific. Sensors shall ignore deploy command family messages, “Deploy Enable”, “Deploy Status” and “Enable Status” messages. Sensors shall send their signal data into S-Frames according to the set-up information stored in their memory.

8.5.2 Sensor slave memory Section II

8.5.2.1 Memory map Section II

Section II of a sensor’s memory should include all information about the sensor necessary for its use in the application. It may also include registers for control of the sensor via the bus.

Table 29 — Example for Section II of the memory in a sensor slave

Addr. (hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	R/W	Memory type
0x30 – 0x31	Sensor configuration								R/W or R	NVM or ROM
0x32 – 0x37	Sensor calibration data								R/W	NVM
0x38 – 0x39	Sensor control								R/W	VM
0x3a – 0x3f	Reserved									

8.5.2.2 Sensor configuration

These optional registers may store static control data for the sensor module, if available. They may contain bits for adaptation of the bus interface to the sensor.

8.5.2.3 Sensor calibration data

These memory locations should be used to store sensor characteristics like sensitivity, offset, etc., which are needed to process the sensor’s signal data correctly. Like the S-Frame set-up, these registers should be covered by an integrity check (e.g. check sum), see 8.3.2.16. For sensors not needing all six bytes, the implemented bytes should be available on the lower address locations (0x32 and higher).

8.5.2.4 Sensor control

These optional registers should store dynamic control data for the sensor module, if available. They may contain bits for control of the sensor’s test mode. Frequently used sensor control bits should also be accessible directly with the “Status Change” command.

8.5.3 Usage of S-Frame sensor data

When raw-data sensors are polled with S-Frames at high sample rates, there is usually no time available for diagnostic communication between master and sensors. Nevertheless, there shall be possibility for the slave to signal errors to the master.

- One of the values that can be represented by the data slot (e.g. 255 = 0xff in the case of 8-bit data) should not be used for sensor signal data. Instead, it should represent an indication to the master that an internal error condition would not allow sensor signal data to be sent. If the master received this value several times in a row, it would recognize that the sensor slave is not healthy.

- When the slave detects an inconsistency (e.g. a check-sum error) in its S-Frame set-up or in its calibration data, it shall not reply to S-Frames.

8.5.4 Bus interrupt

When a smart sensor has recognized a critical impact, it shall request an interrupt (LS0-level) until it gets the opportunity to send its data during an appropriate S-Frame without errors (right-hand state diagram in Figure 17). In order to make sure that the interrupt is not lost (see also case study in Annex D), it may keep the interrupt active until three interrupt bits have been transmitted. When a time-out is reached before the interrupt request has been executed, the interrupt request shall be cancelled and the corresponding error information shall be given to the attached microcontroller.

Actual interrupt transmission shall get disabled during S-Frames ($T = 0$ or $T = LS0$) or after an LS0-bit has been detected on the bus during a D-Frame (i.e. S-Frames or D-Frames with safing shall not be interrupted).

NOTE 1 This includes the detection of the LS0-bit when transmitting an interrupt (i.e. only one interrupt bit will be transmitted per D-Frame). Interrupt transmission gets enabled again when a frame has been finished, or when an SOF announcing a D-Frame (i.e. $T = 1$) is on the bus.

This implies the following.

- During Bus Idle the slave may transmit up to three interrupt bits in a row, but this shall be stopped immediately after an SOF. If all three interrupt bits have not yet been sent, the slave may continue to send interrupt bits after the frame that has been initiated by this SOF.
- During an S-Frame, a slave shall not send an interrupt. If the interrupt request was in time, it can already transmit data in the appropriate slot with the data causing the interrupt.
- During a D-Frame the slave shall transmit only one interrupt bit, assuming that it could read back the LS0-bit it intended to send. However, if an LS0-bit had already been received during the D-Frame, no interrupt shall be sent during that frame. In this way, a deploy command that was sent with the safing level can not be interrupted.

NOTE 2 An interrupt may be sent during the R-bit of a D-Frame with a deploy command. Since the R-bit in such a D-Frame (transmitted with safing) has the LS0-level anyway, the master cannot recognize the interrupt.

- During an SOF, the slave may send one interrupt bit. If the SOF is starting an S-Frame, the S-Frame shall continue without interruption. If the SOF is starting a D-Frame, the interrupt changes the frame into an S-Frame. In this case the master shall either
 - switch directly over to transmission of an S-Frame (e.g. by making use of the SOF that has been turned into an LS0 by the slave), or
 - re-start the D-Frame (recommended action when the D-Frame was a deploy frame) and send the S-Frame afterwards.

NOTE 3 The sensor recognizes the end of a frame by counting the number of bits that have been on the bus since the last SOF. For D-Frames, it counts 28 bits after the SOF (see Table 6). For S-Frames, the number of bits to wait for is specified in its memory (see 8.3.2.14).

The latency times for interrupts from smart sensors are discussed in Annex D.

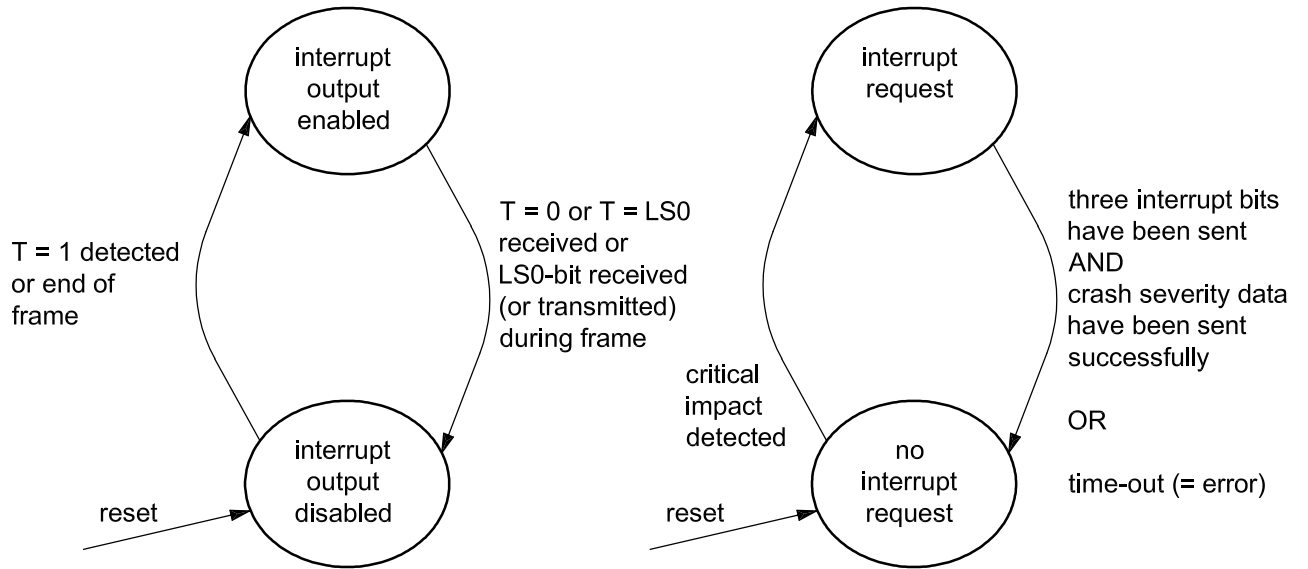


Figure 17 — Example state diagrams for control of the interrupt output of a smart sensor

Annex A (informative)

In-car address programming for daisy-chain systems

Daisy-chain systems allow the slave addresses to be programmed after installation of the slaves in the car. By opening and closing certain daisy-chain switches, the master can select individual slaves without using their address. The process of programming one, several or all slaves in the system is described below.

- Step 1: Power-up the bus.
- Step 2: Send command “Open Daisy-chain Switch” to address 0. Now all unprogrammed slaves open their switches. Only the one unprogrammed slave that is closest to the master remains on the bus.

NOTE Address 0 is the default address of unprogrammed slaves

- Step 3: The master now communicates to the unprogrammed slave by using the address 0. It sets the memory pointer to the location of the slave address and writes the new slave address with the correct parity bit into that location. After waiting the specified time that is needed to execute address programming in the slave, the master confirms successful programming by reading the slave address from this memory location.
- Step 4: Send command “Close Daisy-chain Switch” to the last slave that has just been programmed.

Steps 2 through 4 can be repeated until all slaves have their new addresses. In a daisy-chain ring, the bus is driven from one side only during this process.

It is assumed that the slave’s memory holds detailed non-volatile information about the slave’s functionality and its properties. The master would compare this information with the expected information corresponding to a certain position in the daisy-chain. In this way, the master confirms the proper execution of the address programming process.

50

Annex B (informative)

Guideline for definition of deviations from standard parameters

The parameters in Table 2 mutually influence each other. For example, a higher surge current of the slave supply causes a higher voltage drop across the daisy-chain switches. This can be compensated for by reducing the on-resistance of the switches, for instance. However, the surge current has also an impact on the voltage drop across the protection resistors in front of parallel slaves. These mutual dependencies are quite complex and therefore a checklist is given below to validate a non-standard set of parameters.

- **Surge current voltage drop across daisy-chain switches and protection resistors:**
The total voltage drop across resistive elements (daisy-chain switches or external protection resistors and internal protection resistors) between master and any slave during Power Phase shall be less than 1,6 V.
 - **Voltage drop across daisy-chain switches caused by a transmitting parallel slave on a deployment bus:**
Considering that the master holds the bus voltage at L1-level and the parallel slave is transmitting basically by shorting the bus through the internal protection resistors, the voltage drop across all daisy-chain switches between master and slave shall not exceed 0,6 V.
 - **Voltage drop across daisy-chain switches caused by a transmitting daisy-chain slave on a deployment bus:**
On a deployment bus, the resistance of the daisy-chain switches between master and transmitting slave has to be low enough to allow a minimum current of 12 mA at a voltage difference of 1 V.
 - **Voltage drop across daisy-chain switches caused by a data transmitting slave on a sensor bus:**
Considering, that the slave is transmitting data by pulling the bus to L1-level, the bus level at the master is higher because of the daisy-chain switches in between. The voltage drop across all switches caused by the bias current from the master shall not exceed 0,6 V.
 - **Voltage drop across daisy-chain switches caused by an interrupting slave on a sensor bus:**
Considering that the slave is transmitting data by pulling the bus to LS0-level, the bus level at the master is higher because of the daisy-chain switches in between.
- NOTE LS0-level is also used for signalling bus errors. The voltage drop across all switches caused by the bias current from the master should not exceed 0,2 V.
- **Voltage drop across daisy-chain switches and protection resistors when a parallel switch is transmitting:**
The resistance consisting of daisy-chain switches and protection resistors between master and parallel slave has to be low enough that the parallel slave is able to produce a transmit current of at least 12 mA while the bus is at L0-level.
 - **Current limit by protection resistors:**
On a deployment bus, the value of the protection resistors has to be high enough that a short behind these resistors produces a lower current than 20 mA, when the bus is at L1-level.

Annex C (informative)

Rationale of functionality

C.1 Frame interruption with new SOF

On a parallel bus, a deployable device or a static sensor is connected to the bus via serial resistors. Therefore, it may not be able to read the exact bus level while transmitting. When the master is sending a D-Frame requesting data from such a slave, but interrupts this frame by sending a new SOF while the slave is transmitting, the slave would recognize the SOF but would miss the value of the T-bit sent by the master during the SOF. Therefore, the slave would ignore the new frame that has been started with that SOF. If the new frame is an S-Frame, there is no need to repeat the SOF, since the deployable device would ignore an S-Frame anyway, and because dynamic sensors are capable of monitoring the bus level while transmitting (see also 7.4.2). If the new frame is a D-Frame that this slave should listen to, the SOF has to be sent twice. From this slave's point of view, the first SOF only stops the old frame, and the second frame actually starts the new D-Frame to be received. Of course, the second SOF can only be recognized when there has been at least one idle bit right before. If the new frame is transmitted with safing (i.e. shall not be interrupted), the idle bit before the second SOF must be transmitted with the safing level as well. This ensures that smart slaves with pending interrupts do not try to overwrite the second SOF.

C.2 “Deploy Enable” command

Some systems have significant time between crossing the first threshold and crossing the critical threshold. They can send “Deploy Enable” after the first and “Deploy” after the critical one. Systems for which this does not apply simply send “enable” by default.

C.3 Deploy verification

Deploy verification can be done by normal diagnostics commands. However, since some squibs may die a short time after a deployment, a quick scan of all squibs is preferred in order to log the information. (However, those who did deploy are not critical, and we would need diagnostics for those who did not deploy!)

C.4 Deploy command code

The deploy command code is “0011”. It includes “0”s, because only “0”s can carry the safing level. It also includes “1”s, because leakage that is higher than the L1-clamping current may create a sequence of “0”s with safing level as well. The first command bit is a “0”, because in this way interrupts from smart sensors are disabled (provided that it has been sent with safing level). The commands “0001” and “0010” are used for optional test deploy commands, which each affect only one of the two deploy switches inside of the squib driver. If these commands are implemented in the squib driver, it should recognize the MSBs “00” as the actual deploy command and the LSBs “11”, “10” or “01” as the selection of both switches or of only one of the two. Then the deploy command can be tested for each switch separately without danger of deploying (using the combinations “10” and “01”), which should give high confidence that the actual deploy command (combination “11”) really causes deployment.

Annex D (informative)

Latency time analysis for interrupts from smart sensors

D.1 General

Latency time is defined as the worst-case duration between the occurrence of an interrupt requesting event in the sensor and the actual start of an S-Frame polling message. This takes into account that the request to send an interrupt may be asynchronous to the bus clock and therefore the sensor may have to wait for the start of the next Data Phase to send the interrupt.

In the following case study it is shown that the worst-case latency time is two (2) nominal bit times.

If S-Frames are sent at higher speed than D-Frames, then the worst-case latency time is two (2) nominal bit times at low speed plus one (1) nominal bit time at high speed.

The timing diagrams use the following symbols:

- P Power Phase;
- D L0- or L1-level during Data Phase;
- S LS0-level during Data Phase;
- a clock tick of half the nominal bit time at the selected bus speed.

EXAMPLE

P-D-P-D means a sequence of two normal data bits (power, data, power, data) at high speed.

P-P-D-D means an SOF (double-length power, double-length data) at high speed.

PPP-DDD means a sequence of one normal data bit (power, data) at low speed.

PPP-PPP means a double-length Power Phase at low-speed, as can occur during an SOF.

D.2 Case: Sensor sends interrupt during SOF

- a) If the SOF was sent by the master as a D-Frame start ($T = 1$), the interrupt turns it into an S-Frame start ($T = LS0$), which is recognized by the master so that the master can immediately continue with sending an S-Frame.

Interrupt latency is 0,5 bit times.

bus level sent by master	...P-D-P-D-P-P-1-1-P-D-...
event at sensor	[
earliest time to send interrupt	[
actual bus level because of interrupt	...P-D-P-D-P-P-S-S-P-D-...
effective start of S-Frame	[
latency time	+--+ (0,5 bit times)

- b) If the SOF was sent by the master as an S-Frame start ($T = 0$), the interrupt does not change this. The master simply continues to send the S-Frame and recognizes that a slave has sent an interrupt.

Interrupt latency is 0,5 bit times.

```

bus level sent by master          ...P-D-P-D-P-P-0-0-P-D-...
event at sensor                   [
earliest time to send interrupt   [
actual bus level because of interrupt ...P-D-P-D-P-P-S-S-P-D-...
effective start of S-Frame        [
latency time                       +-+ (0,5 bit times)
    
```

D.3 Case: Sensor sends interrupt right after SOF

Interrupt latency is two (2) bit times.

```

bus level sent by master          ...P-D-P-D-P-P-1-1-P-D-P-P-S-S-P-D...
event at sensor                   [
earliest time to send interrupt   [
actual bus level because of interrupt ...P-D-P-D-P-P-1-1-P-S-P-P-S-S-P-D...
effective start of S-Frame        [
latency time                       +-----+ (2 bit times)
    
```

D.4 Case: Sensor sends interrupt during other bit within D-Frame or during Bus Idle

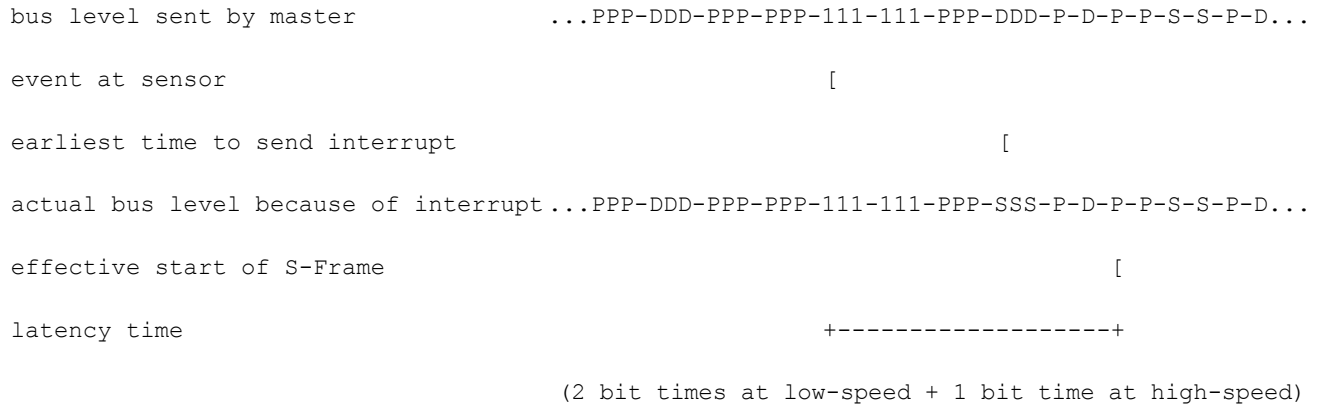
Interrupt latency is 1,5 bit times.

```

bus level sent by master          ...P-D-P-D-P-D-P-P-S-S-P-D-...
event at sensor                   [
earliest time to send interrupt   [
actual bus level because of interrupt ...P-D-P-D-P-S-P-P-S-S-P-D-...
effective start of S-Frame        [
latency time                       +-----+ (1,5 bit times)
    
```

D.5 Case: Sensor sends interrupt right after SOF, which causes a bus speed increase

The master recognizes the interrupt, then switches to high speed for one bit time and then starts the S-Frame. Interrupt latency is two bit times at low speed plus one bit time at high speed.



D.6 Case: Sensor sends interrupt right after S-Frame

This is identical to case D.2 or case D.4: During the S-Frame a request for an interrupt may pop up. The sensor knows that there is an S-Frame in progress and therefore waits for the end of that frame by counting the bits and comparing it with the specified frame length, which is stored in its OTP memory. The sensor sends the interrupt during the first Data Phase *after* the S-Frame, which is either an SOF (case D.2) or an idle bit (case D.4).

D.7 Case: Interrupt and deploy messages

When deploy messages are sent with safing level, interrupt gets disabled with the first bit after SOF, because the next bit after SOF is the R-bit, which is transmitted as a “0” (see 7.4.3). This means that an already started deploy message cannot be interrupted by a smart sensor. On the other hand, a pending interrupt request has higher priority than the start of a deploy frame, as explained in the following example.

EXAMPLE When an interrupt request occurs during an S-Frame (e.g. the impact condition has been recognized too late to send the new impact data already in that frame), the sensor will send out the interrupt with the next bit after the S-Frame. If this bit is an SOF that was intended to start a deploy message, the interrupt will change it to another S-Frame. This means for the application that transmission of this deploy message is postponed until the updated impact information has been received as well.

Annex E (informative)

CRC examples

Table E.1 — CRC examples for D-Frames

T	R	Command					Slave address + data or address MSBs + bitmap												CRC									
																			1	1	0	1	1	0	1	0		
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	1	0
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	1	0	0	1
1	0	0	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	1	1	0	0
1	0	0	0	1	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	1	1	0
1	0	1	0	1	0	1	1	1	1	0	0	0	0	1	1	0	0	1	0	0	1	1	1	1	0	1	1	0
1	0	1	0	1	0	0	0	0	0	0	1	1	1	1	0	0	1	1	0	1	1	0	1	1	0	0	0	0
1	0	1	1	0	0	0	1	0	0	1	1	0	0	0	1	1	1	1	0	0	0	0	0	1	0	0	1	1
1	0	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	1	1	0	0	0	0

Table E.2 — CRC examples for S-Frames

Data length	CRC-length	T	MSA	SEL	Data (binary)	CRC (binary)
4	3	0	0	0	1001	001
		0	0	1	1001	000
	8	0	0	0	1001	0100 0010
		0	0	1	1001	1100 1011
8	3	0	0	0	1001 0110	111
		0	1	0	0010 1001	111
		0	0	1	1001 0110	001
		0	1	1	1001 0110	110
	8	0	0	0	1001 0110	0001 0010
		0	0	1	1001 0110	0100 1010
10	3	0	0	0	10 0101 1001	010
		0	1	0	00 1010 0100	001
		0	0	1	10 0101 1001	111
		0	1	1	10 0101 1001	110
	8	0	0	0	10 0101 1001	0101 0001
		0	0	1	10 0101 1001	0010 1000

Annex F (informative)

Deployable devices

NOTE 1 This International Standard is a bus specification only and the description of the Application Layer covers functions associated with bus communication only. For illustration purposes, their relation to the actual application is included, but this is not to be understood as a complete description of a device's functionality.

The bus system provides several safety mechanisms to avoid inadvertent deployment. The properties of the Physical Layer (analogue safing) and the definition of the messages in the Application Layer allow to control these mechanisms and to diagnose their functionality individually. The names of the messages and of the bits in the messages and their explanations (e.g. Figure 15) point to a specific implementation of these mechanisms.

NOTE 2 Other implementations are also allowed, as long as the effect of the messages and of the bits in the messages is basically the same and the messages can be used in the same way.

This bus specification assumes that deployable devices are processing several signals for the decision to deploy. Their interaction is illustrated in Figures F.1, F.2, F.3 and F.4. Table F.1 lists the correlation between the terms used in these figures and the terms used for the message definitions.

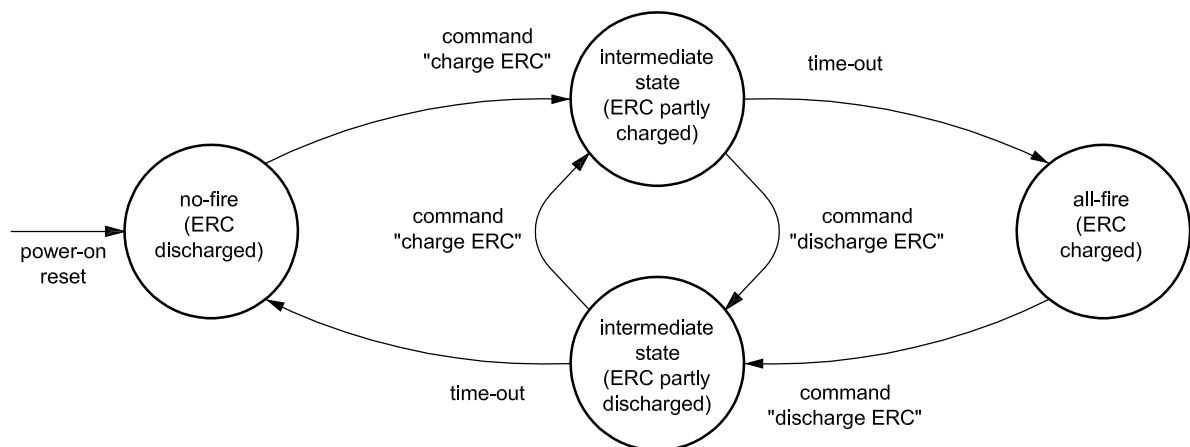


Figure F.1 — State diagram illustrating the control of the energy reserve capacitor (ERC) in a deployable device, which stores the deployment energy

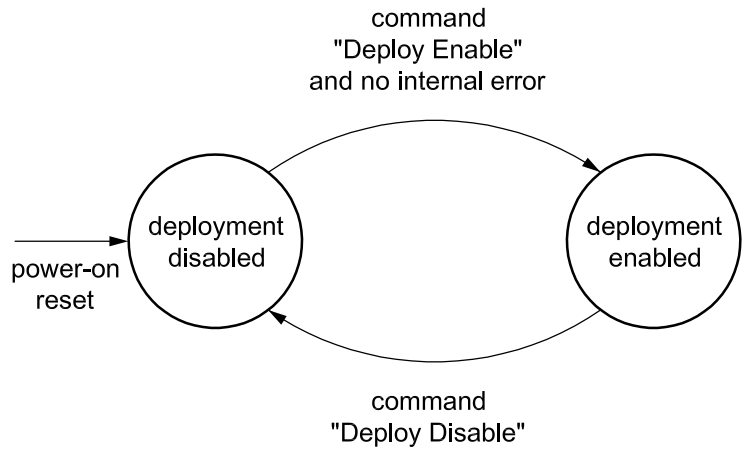


Figure F.2 — Example state diagram illustrating the usage of “Deploy Enable” and “Deploy Disable” commands

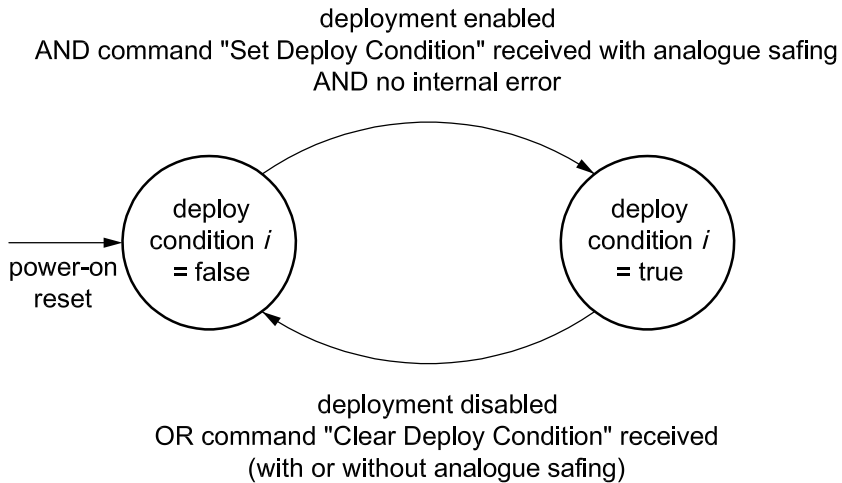
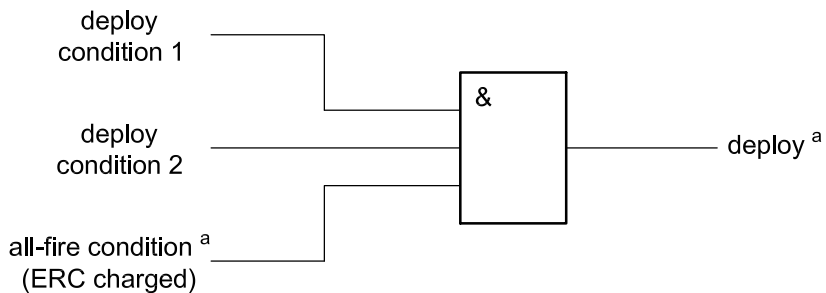


Figure F.3 — Example state diagram for deployment condition i ($i = 1, 2$)



^a When the ERC is in an intermediate state, deployment can neither be guaranteed nor excluded.

Figure F.4 — Model of the deployment actuator in the deployable device

Table F.1 — Correlation between terms used in the abstract model (Figure F.4) and those used for the example implementation (Figure 15)

Abstract model	Example implementation
Deploy condition 1	HSD
Deploy condition 2	LSD
Set/clear deploy condition 1, 2	Switch on/off HSD, LSD

Annex G
(informative)

Slave manufacturer identification codes

In 8.3.2, several slave manufacturer identification (ID) codes were introduced. The assignment of such codes to manufacturers is outside the scope of this International Standard.

© ISO 2006

.....

ICS 43.040.80

Price based on 60 pages