PLCopen Technical Committee 5

Safe Motion

PLCopen Technical Specification

PLCopen Document, Version 1.0, Official Release

Date: July 21, 2017

Total number of pages: 45

This paper on Safe Motion is a document as created by the PLCopen Technical Committee 5, TC5 - Safe Software.

It summarises the results of the Task Force Safe Motion and several meetings of PLCopen TC 5, containing contributions of all its members.

The present specification was written thanks to the following members:

Name	Company
Ulf Schuenemann	3S Smart Software Solutions
Engelbert Meier	Baumüller
Guido Beckmann	Beckhoff
Roland Wagner	B&R Automation
Andreas Hager	B&R Automation
Burchardt Hagen	Bosch Rexroth
Wolfgang Zeller	Hochschule Augsburg
Makoto Okuda	Kanagawa Research Institute
Karl Pickan	Kollmorgen
Michael Niehaus	Lenze
Josep Plassa	Omron
Takashi Matsukuma	Omron
Harry Koop	Phoenix Contact Software
Kevin Hull	Yaskawa
Eelco van der Wal	PLCopen

Change Status List:

Version number	Date	Change comment
V 0.1	April 3, 2015	First version done by EvdW as result of several webmeetings
V 0.2	April 30, 2015	As result of feedback, the webmeeting on April 10, and further work
V 0.3	June 17, 2016	Added CC-Link and MECHATROLINK specification
V 0.31	July 14, 2016	First changes added E. Meier and decisions webmeeting July 11
V 0.32	August 30, 2016	As result of the webmeeting on Aug. 24 and additional feedback Ch. 4.5
V 0.99	September 26, 2016	As result of the webmeeting on Sept. 21
V 0.99A	January 25, 2017	As a result of feedback and the webmeetings on Dec.14 & Jan 25
V 0.99B	March 6, 2017	As result of the Face2Face meeting on Feb 21 & 22 at Bosch
V 1.0	July 21, 2017	Official release

Contents

Introduction

The PLCopen Technical Committee 5, TC5, deals with software for safety environments with a focus to the operating personnel. The Committee has already published several specifications, like the Technical Specification Part 1: Concepts and Function Blocks Version 1.0 of January 31, 2006 as well as consecutive parts.
Inside these published specifications the safety aspects around motion have been described only partially. This document fills this gap.

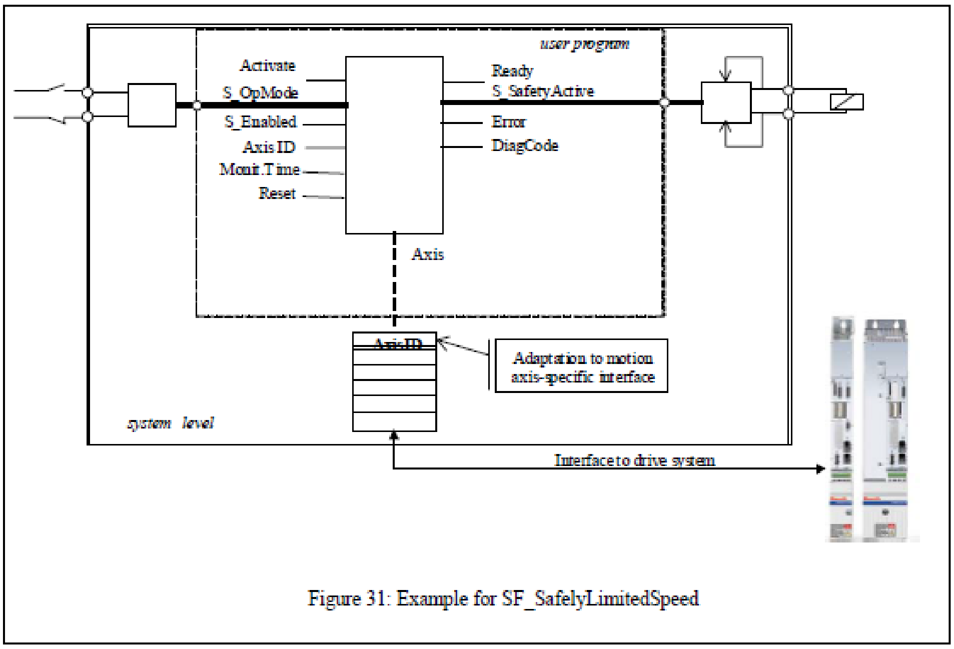


Figure 1: Example of Part 1 concerning Safety Limited Speed

1.1. Rationale

The emergence of (fast) digital networks made it possible to link many motors to a controller. In many cases servo technology replaces the large single motor solution, making the functionalities locally available in the machine and also replacing the previously mechanical solutions by software control. This solution is also referred to as "mechatronic solutions".

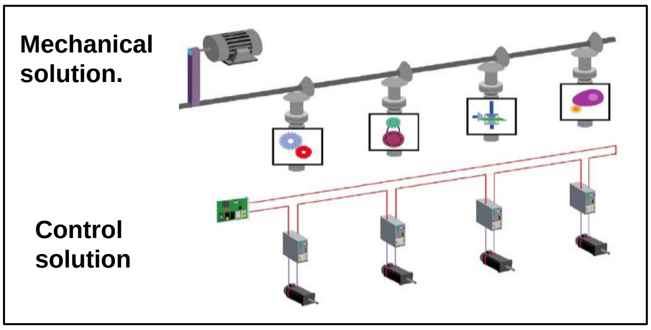


Figure 2: From mechanical to mechatronics

With multiple motors, the safety connections have changed also: from a hardwired safety solution to a solution over a safety network with a dedicated safety controller added to the functional controller (PLC).

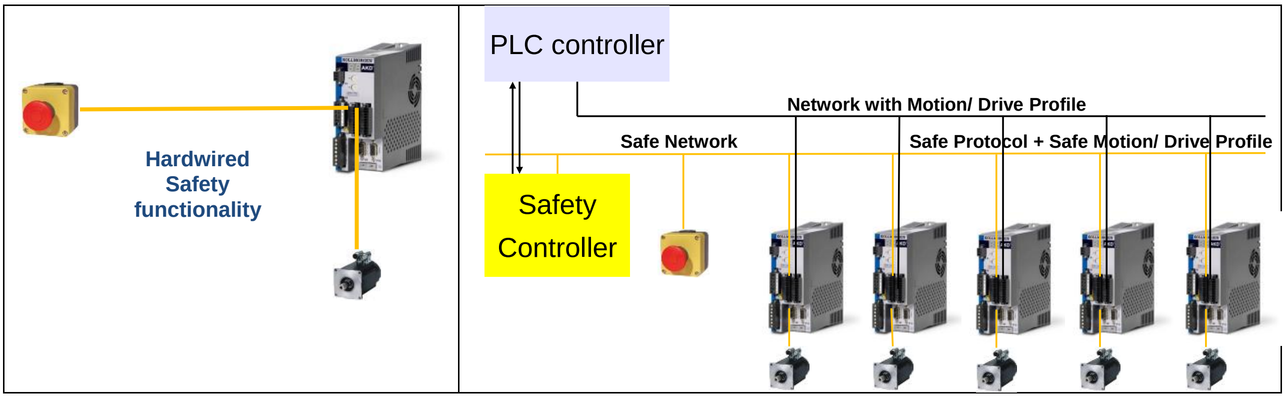
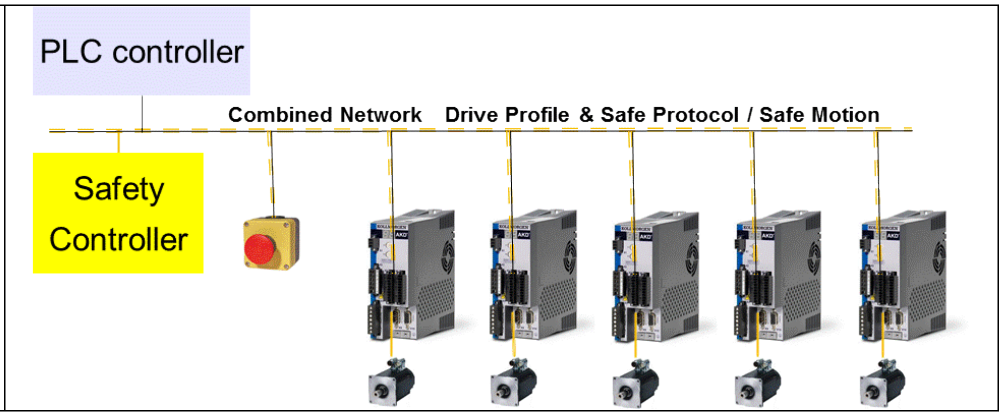


Figure 3: From hardwired to networked safety functionalities

The next step was to integrate the safety data transfer into the standard network, along with the logic and motion data. As a result only one network is necessary for the whole machine. Nowadays this sort of network types is widely available on the market



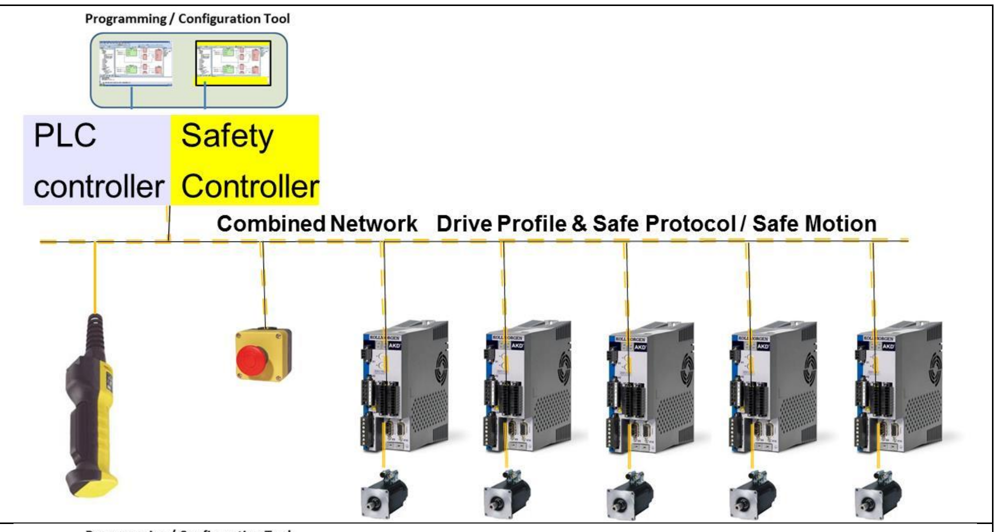
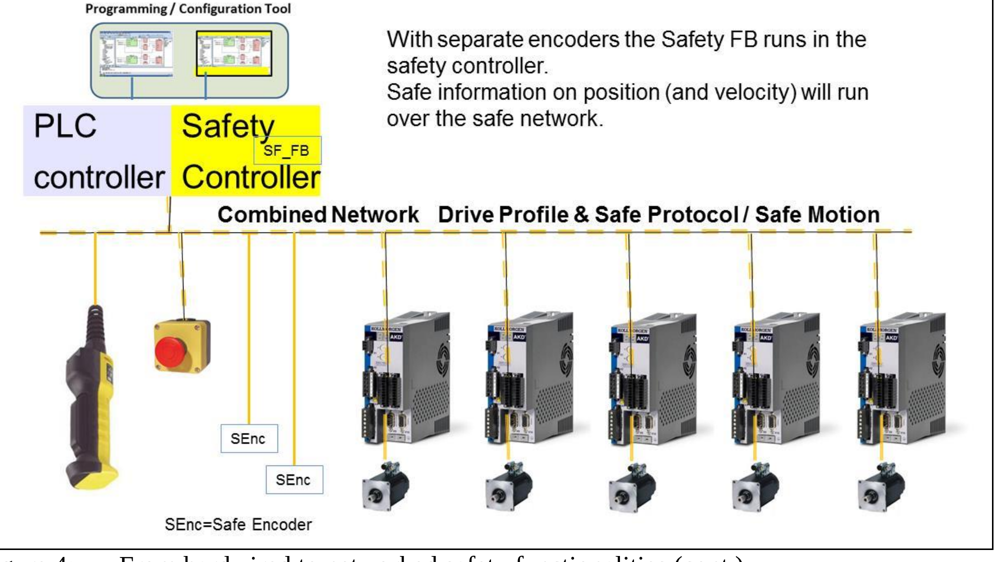
The next logical step is to integrate the controllers in one platform too. In this way the development environment for safety and non-safety can be integrated in one common system, which creates the possibility to include the safety functionality in the project from the start.	
Looking to the future it will be able to transfer the information on movements per axis over the safety protocols too, and use safe motion sensors which can provide accurate safe positions in a digital form. In that way it will be possible to calculate the overall behavior of a machine. For instance the movements of the working pieces in a robot arm can be limited to a safe speed.	

Figure 4: From hardwired to networked safety functionalities (cont.)

In order to support these trends it is necessary to reflect the safety functionality in the software environment. This paper identifies user friendly solutions for this.

[Note from PLCopen: the last picture in Figure 4:above does reflect the future and is not covered in this document. Also dynamic changing of safety related parameters is not covered in this document. This can be covered in future updates or extensions of this work.]

1.2. Positioning of work

When we refer to safe motion in this document we consider the safety of the operating personnel in machines with moving parts. An example of such a machine is a mechanical press or a packaging machine.

Note that a “safety drive” is not to be confused with “safe drive [system]” which is referred to in the automotive sector.

Figure 5: Example of safe driving in automotive

2 General overview

When we refer to safe motion in this document we mean the safety of the operating personnel in machines with moving parts. This includes the motion of the different motors, combined with their drives, in the machine in a safe way.
In accordance with the Machinery Directive, when the drive is brought to a standstill, the operating status must be safely monitored and maintained.
In many cases this is done via Safe Motion Monitoring Functionalities (SMMF) - whether external or drive-integrated - like defined in IEC 61800-5-2. Safe motion monitoring can be implemented through speed monitoring and as a result potentially hazardous movements are safely detected and appropriate countermeasures can be initiated in time.

In Figure 6: Overview of the safe motion definitions and area this is depicted with the separation between the safety drives and safe motion control (or alternatively called safe motion monitoring).

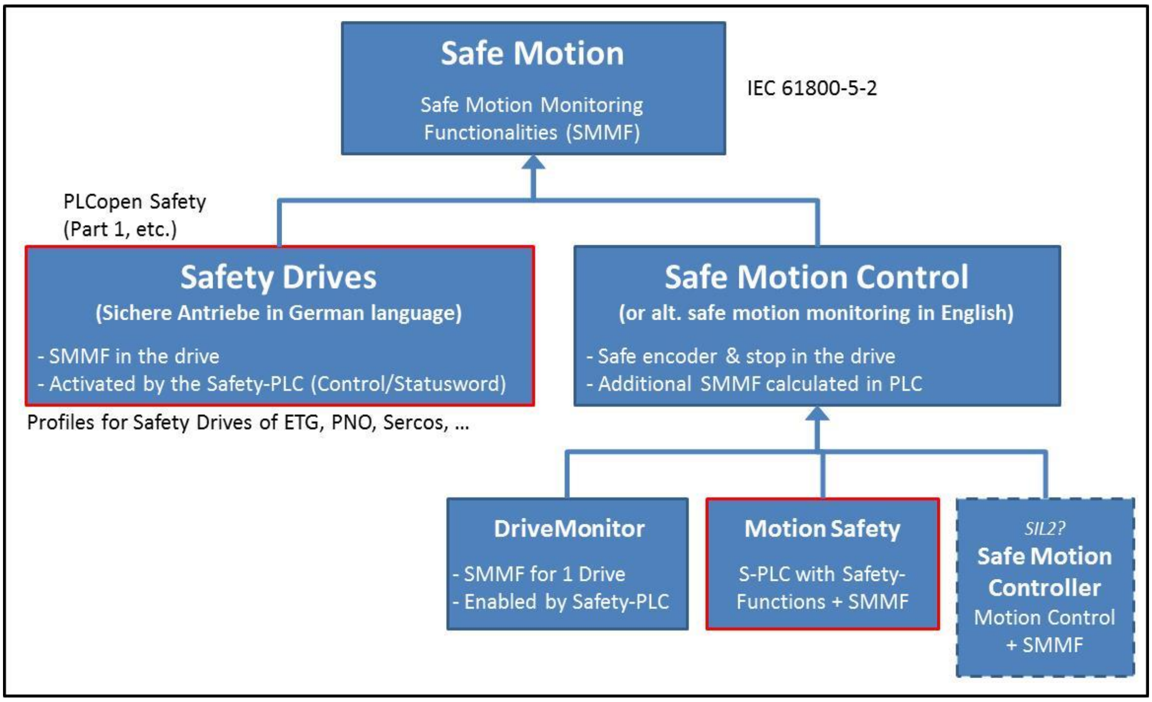


Figure 6: Overview of the safe motion definitions and area

The term “safe” is understood in terms of functional safety from the relevant (machinery) safety standards IEC 61508, IEC 62061, and ISO 13849-1.

2.1. Operational aspects of the safety function

In line with the hardwired safety functionality, the first step with multiple drives and motors is to include the safety monitoring function in the drive. As show in Chapter 1.1 a next option is to monitor this overall in a safety controller.

Safety Drives: Safe Motion monitoring (SMMF) in Drive
Safety Drives are drives with built-in safety functions. … the drive can be run in different between modes [SMMF active | inactive]
Various safety functions are available in safety mode. The transitions between the safety functions normally get handled by a drive internal state machine and might underlay different priorities [and] drive … parameters ….
[Conform PLCopen Safety, Part 2]

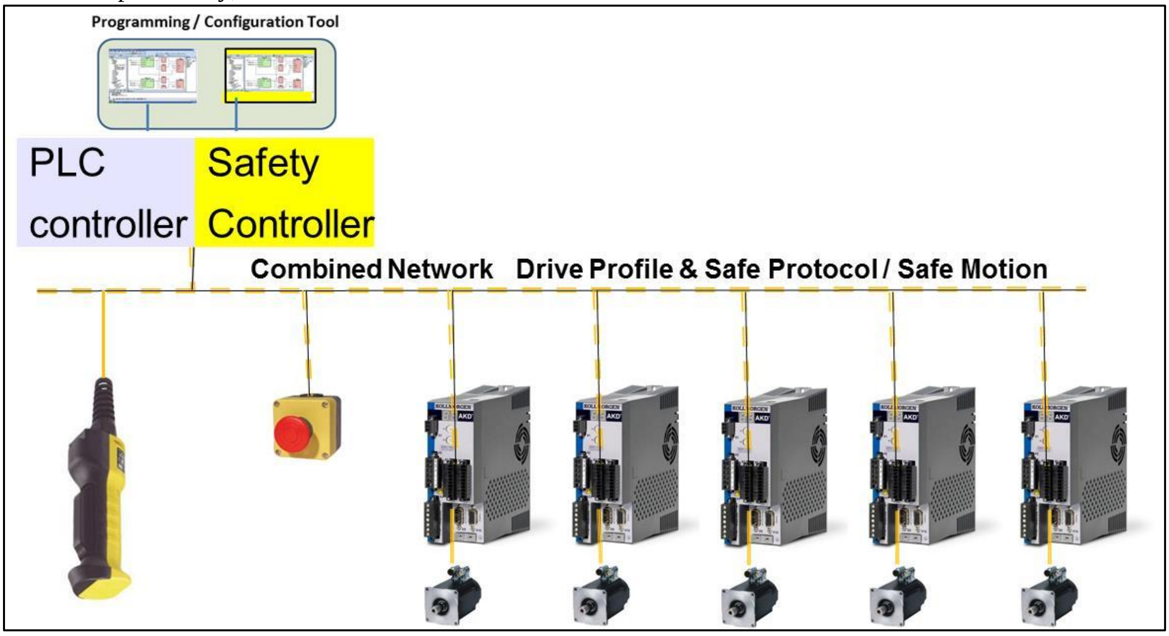


Figure 7: Safe Motion monitoring in Drive

Motion Safety: Safe Motion monitoring (SMMF) in Safety-PLC

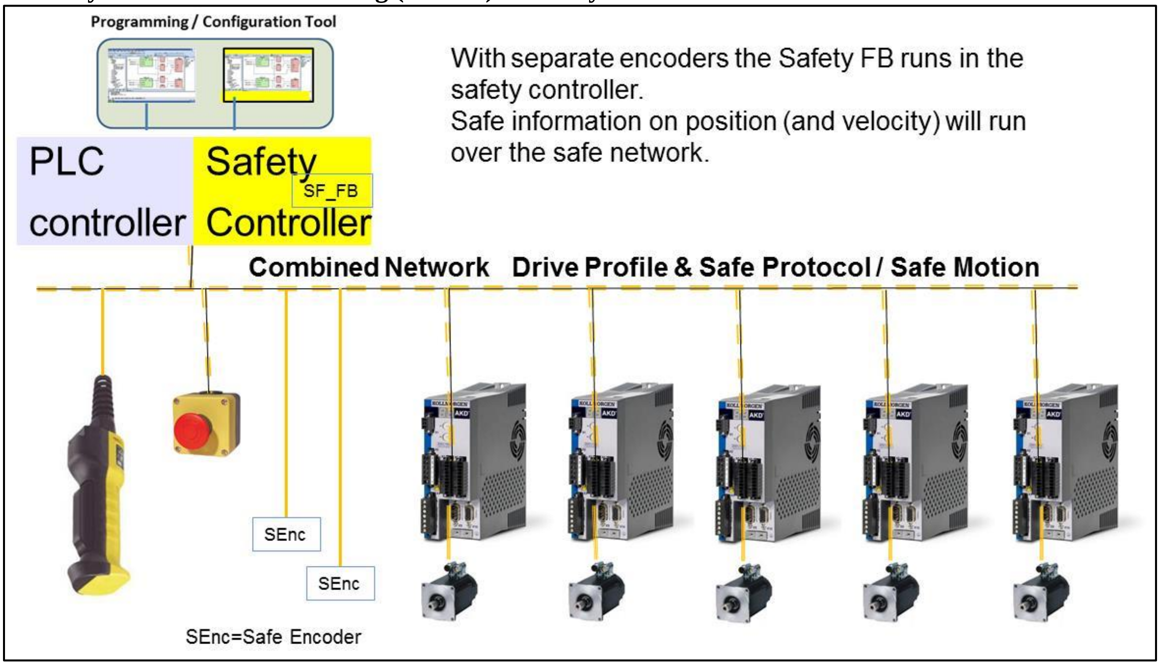
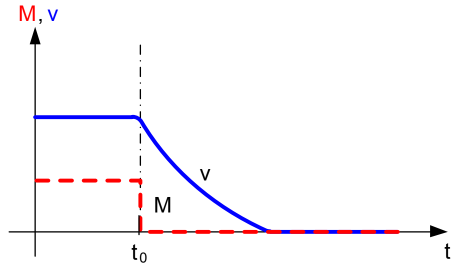
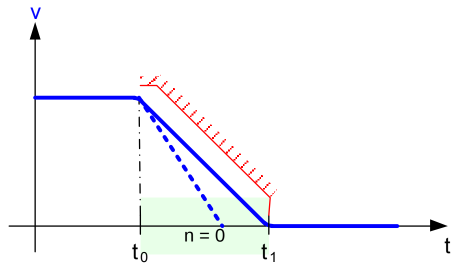
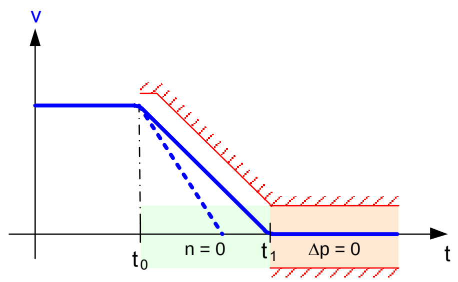
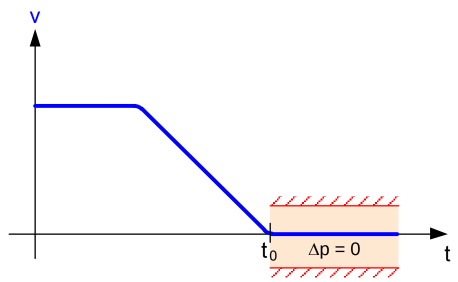
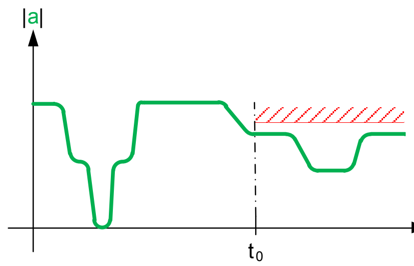


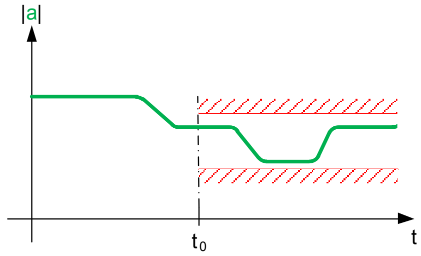
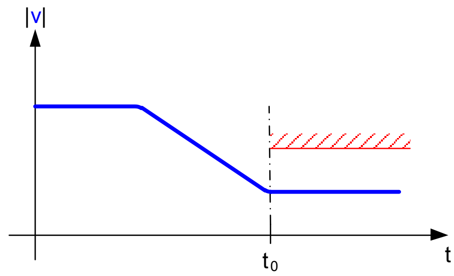
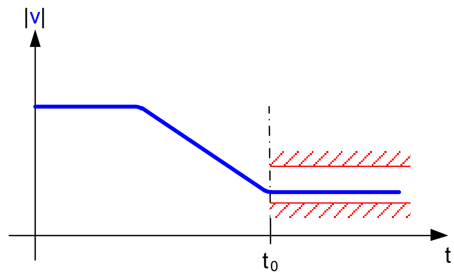
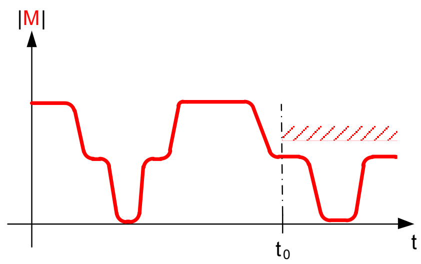
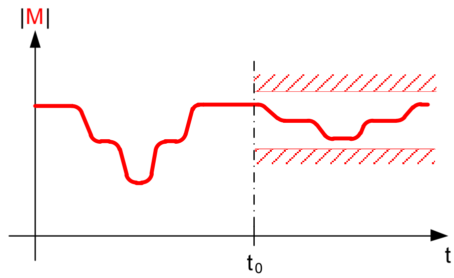
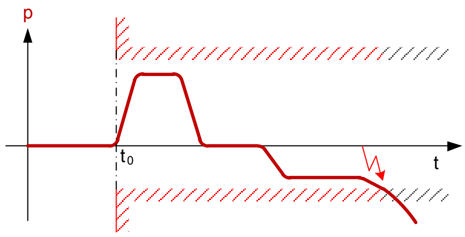
Figure 8: Safe Motion monitoring in Safety PLC

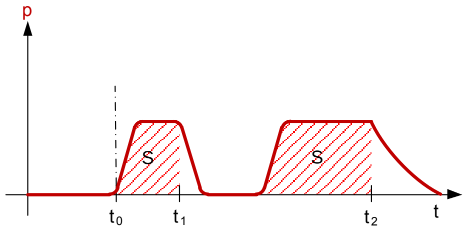
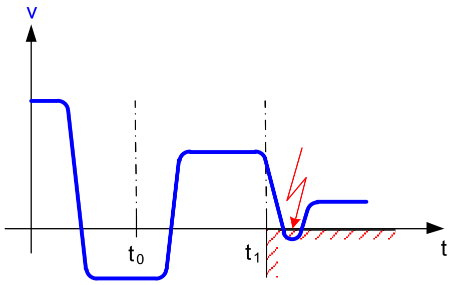
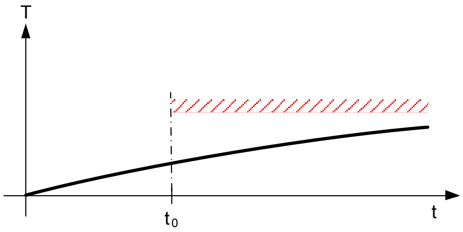
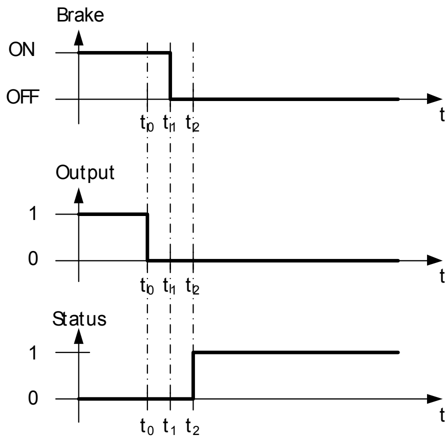
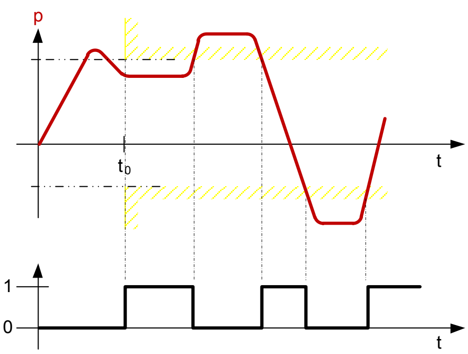
Note: the configuration in Figure 8: Safe Motion monitoring in Safety PLC is not yet covered in this specification.

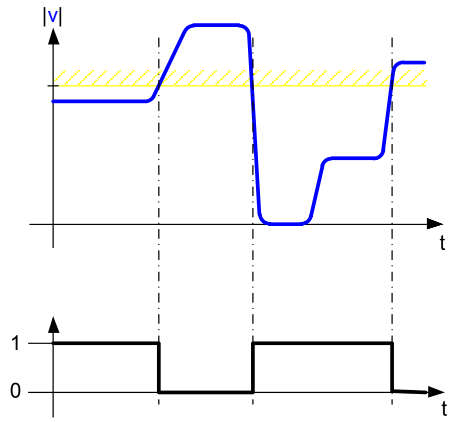
2.2. Overview common safe motion functionalities

Conform to the relevant safety standard today (IEC 61800-5-2) the following motion related safety functions can be identified:

Function (EN61800-5-2:2007 Chapter)	English (German)	Icon	Description
STO 4.2.2.2	Safe torque off (Sicher abgeschaltetes Moment)		Power energy which, can cause any motion is not applied to the motor.
SS1 4.2.2.3	Safe stop 1 (Sicherer Stopp 1)		Motor Deceleration. STO after Deceleration.
SS2 4.2.2.4	Safe stop 2 (Sicherer Stopp 2)		Motor Deceleration. SOS after Deceleration.
SOS 4.2.3.1	Safe operating stop (Sicherer Betriebshalt)		Prevents the Motor from deviating more than a defined distance from the stopped position
SLA 4.2.3.2	Safely-limited acceleration (Sicher begrenzte Beschleunigung)		Prevents the Motor from exceeding the acceleration limit

Function	English (German)	Icon	Description
SAR 4.2.3.3	Safe accelaration range (Sicherer Beschleunigungs-bereich)		Keeps the motor acceleration / deceleration within specified limits.
SLS 4.2.3.4	Safely-limited Speed (Sicher begrenzte Geschwindigkeit)		Prevents the motor from exceeding the specified speed limit
SSR 4.2.3.5	Safe speed range (Sicherer Geschwindigkeits-bereich)		Keeps the motor speed within specified limits
SLT 4.2.3.6	Safely-limited torque (Sicher begrenztes Moment)		Prevents the motor from exceeding the specified torque / force
STR 4.2.3.7	Safe torque range (Sicherer Momenten-bereich)		Keeps the motor torque / force within the specified limits
SLP 4.2.3.8	Safely-limited position (Sicher begrenzte Position)		Prevents the motor shaft from exceeding the position limits

Function	English (German)	Icon	Description
SLI 4.2.3.9	Safely-limited increment (Sicher begrenztes Schrittmaß)		Prevents the motor shaft from exceeding the limit of position increment
SDI 4.2.3.10	Safe direction (Sichere Bewegungs-richtung)		Prevents the motor shaft from moving in the unintended direction
SMT 4.2.3.11	Safe motor temperature (Sichere Motortemperatur)		Prevents the motor temperature from exceeding a specified upper limit.
SBC 4.2.3.12	Safe Brake Control (Sichere Bremsen-ansteuerung)		Provide safe output signal(s) to control the externals brake(s) [Note: Safe Brake Control is often linked to Safe Break Test (SBT). However the latter functionality is not covered in this document]
SCA 4.2.3.13	Safe cam (Sicherer Nocken)		Safe output Signal to indicate the motor shaft is within a specified range

Function	English (German)	Icon	Description
SSM 4.2.3.14	Safe speed monitor (Sichere Geschwindigkeits-überwachung)		Safe output signal to indicate whether the motor speed is below a specified limit

Abbreviations in the pictures above

v	… Speed	M	… Torque, Force
n	… Rotation Speed	p	… Position
a	… acceleration	T	… Temperature

Alternatively:

STO	Safe Torque Off	Safe Stop and brake functions
SS1	Safe Stop 1	Safe Stop and brake functions
SOS	Safe Operating Stop	Safe Stop and brake functions
SS2	Safe Stop 2	Safe Stop and brake functions
SBC	Safe Brake Control	Safe Stop and brake functions
SLA	Safety Limited Acceleration	Safe Motion Functions
SAR	Safe Acceleration range	Safe Motion Functions
SLS	Safety Limited Speed	Safe Motion Functions
SSR	Safe Speed Range	Safe Motion Functions
SLT	Safely-Limited Torque	Safe Motion Functions
STR	Safe Torque Range	Safe Motion Functions
SLP	Safely-Limited position	Safe Motion Functions
SLI	Safely-Limited Increment	Safe Motion Functions
SDI	Safe Direction	Safe Motion Functions
SCA	Safe Cam	Safe Motion Functions
SSM	Safe Speed Monitor	Safe Motion Functions
SMT	Safe Motor Temperature	Safe Motion Functions

These functions can be grouped in the following order:

Safe Stop and brake functions	
STO	Safe Torque Off
SS1	Safe Stop 1
SOS	Safe Operating Stop
SS2	Safe Stop 2
SBC	Safe Brake Control

Safe Motion Basics Functions	
SLS	Safety Limited Speed
SS1	Safety Stop 1. maintaining power and after certain time delay OFF
SS2	(Stop function) like Stop Cat.2 of EN60204, Controlled shut down while maintaining the supply of power, and after Standstill Can includes SOS
SLP	Safely-Limited position
…	SafeEncoder. Combine to encoders

Safe Motion Advance Functions	
SLA	Safety Limited Acceleration
SAR	Safe Acceleration range
SSR	Safe Speed Range
SLT	Safely-Limited Torque
STR	Safe Torque Range
SLI	Safely-Limited Increment
SDI	Safe Direction
SCA	Safe Cam
SSM	Safe Speed Monitor
SMT	Safe Motor Temperature

Figure 9: Grouped overview of safety functions

Concerning the safety parameters, the following statement can be made:

Parameter settings of the monitoring functions
<< Ref. Pilz 2011 [The Safety Compendium Chapter 6 SafeMotion]>>
Safe motion monitoring requires not just safe motion detection but also the opportunity to specify limit values safely. The way in which this is achieved depends on the level of dynamics and the flexibility within the machine.

Limit values	Description	Dynamics
Constant	Fixed during commissioning and cannot be amended during operation	-
Selectable	Possible to select/change the appropriate value from a fixed set of limit values during operation	O
Dynamic	Limit values are calculated and adjusted during operation	+

As example of dynamic limit values one can look at robotics: the safe speed of the robot can be reduced depending on the distance of the operator from the hazard zone: the closer the operator comes to the danger zone, the slower the robot moves This involves the safety logic and is calculated in the software

<< Note: Safety Controller is often seen as a separate unit. Safety Logic can be part of the safety controller or in the safety drive >>

3 Mapping to the safe profiles in the networks

In order to use the safety integrated networks the safety functions have to be mapped. This chapters combined with 5 Safe Profiles provides an overview; however this can be incomplete and is not intended for implementations. For this purpose the applicable network standards have to be used. Also note that there is no preference or order applicable here: it is just a small overview.

What is generic in the current status of the safety integrated networks is that there are basically two bytes or a word (double bytes) used for communicating the safety commands and status. However there are slight differences in the content and meaning of the words at bit level.

Overview Control Byte

Bit	OMAC	ProfiSafe	EtherCAT	OpenSafety	CIP / Sercos	CC-Link IE	MECHATRO LINK
0	STO	STO	STO	Reset	Mode	STO	STO
1	SS1	SS1	SS1	Activate	Emergency Stop	SS1	SS1
2	SS2	SS2	SS2	STO	Enabling	SS2	SS2
3	SOS	SOS	SOS	SBC	SMM1	SOS	SOS
4	SLS	SLS	SSR	SS1	SMM2	SSR	SSR
5	UserD	SLT	SDIp	Reserved	SMM3	SDIp	SDIp
6	UserD	SLP	SDIn	Vendor	SMM4	SDIn	SDIn
7	Error Ack	Internal	Error Ack	Vendor	SMM5	Error Ack	Error Ack

More detailed information on the different networks can be found in Chapter 5 Safe Profiles .

4 Proposal for Safety Drives

This proposal is based on the architecture that the safety functionality is in the drive, incl. all relevant parameters, as shown in Figure 7: Safe Motion monitoring in Drive.

4.1. SF_SafetyRequest for activating & monitoring the drive

In PLCopen Safety Part 1 the Function Block SF_SafetyRequest is defined. The graphical representation is:

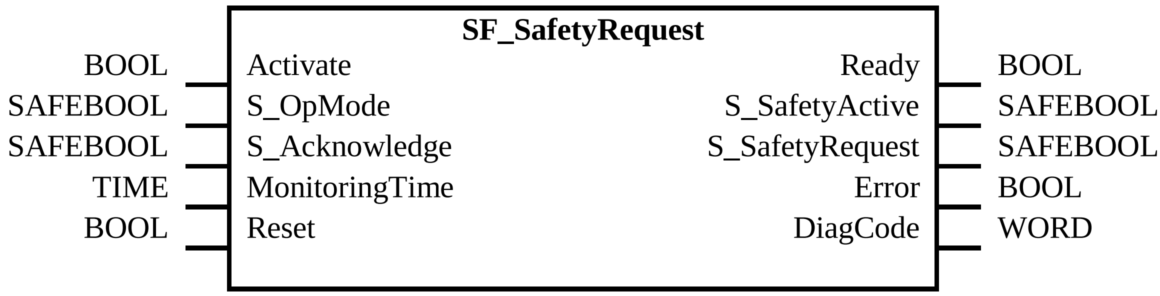


Figure 10: Graphical representation of SF_SafeRequest

The relevant inputs and outputs are shown in Figure 11: Description of SF_SafetyRequest with reduced set of inputs and outputs, providing a basic overview of the functionality.

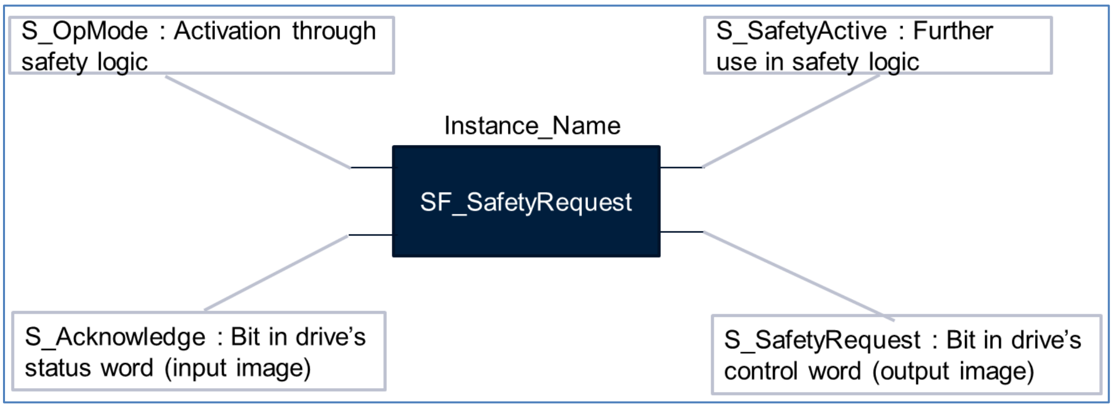


Figure 11: Description of SF_SafetyRequest with reduced set of inputs and outputs

The main inputs and outputs of SF_SafetyRequest shall be linked as follows:

- Top-left (*Activation through safety logic*): Input **S_OpMode** is linked to the logic determining whether the safety drive function (e.g. safely limited torque) needs to become active within the time specified on input MonitoringTime (not shown). S_OpMode may be connected to the output value of an FB monitoring a device (e.g. SF_ESPE monitoring a light curtain, or SF_ModeSelector monitoring a mode selecting device), or some combination of conditions.
- Top right (*Further use in safety logic*): Output **S_SafetyActive** provides the feedback whether the safety drive function has become active *within the specified monitoring time* after its need has been determined (input S_OpMode). If this output is active it can be used to activate the next step, like e.g. opening a gate.
- Bottom left (*Bit in drive's status word*): Input **S_Acknowledge** reflects the status of the drive relevant to the safety drive function requested by this FB (e.g. whether the limitation of torque is currently active), and is the reflection of the process input image w.r.t. to the drive.
- Bottom right (*Bit in drive's control word*): Output **S_SafetyRequested** is the bit that goes to the drive in the safe control word for activating the function in the drive. Note that the control word can be a byte.

If the input on the top left is FALSE, there is no activation so output top right is FALSE. However if there is a safety request and the input top left is TRUE, the top right output reflects the status of the drive or its exceeding of the monitoring time. The bottom right bit reflects the operation by setting the relevant bit in the control word for safety active, and after acknowledge of this in the safety drive, the bottom left input reflects this change (to TRUE).

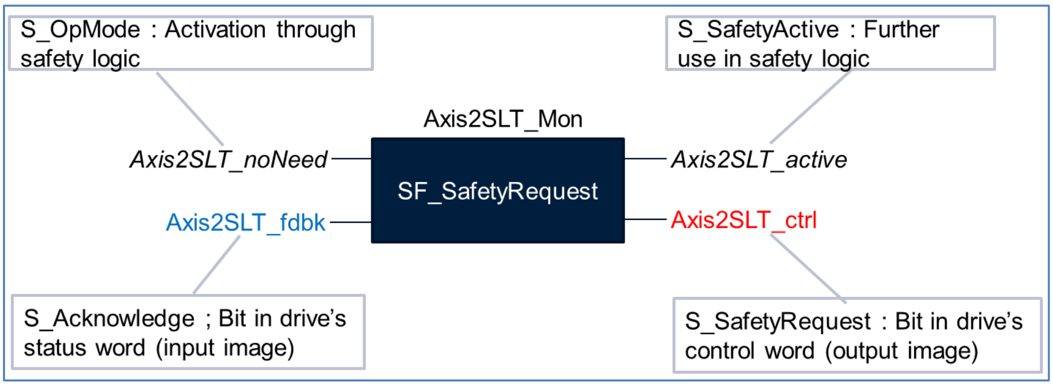


Figure 12: Extended description of SF_SafetyRequest with reduced set of inputs and outputs

This SF_SafetyRequest functionality can be used in a broad safety sense, incl. the safe motion functionalities. For instance, in order to use this FB for SLT (Safely Limited Torque) one combines the inputs Axis2SLT_noNeed and Axis2SLT_fdbk to generate the relevant safety outputs Axis2SLT_active and Axis2SLT_ctrl, as depicted in Figure 12: Extended description of SF_SafetyRequest with reduced set of inputs and outputs, providing the required safety functionality. Multiple instances of SF_SafetyRequest may be used to cover all safety drive functions (IEC61800, profiles, vendor-specific).

In this way most of the safety drive functionalities can be mapped easily, especially by providing a "guideline" with defined naming conventions, containing a generic scheme how to name signals related to the functions supported by a safety drive as described in Par. 4.2 The Safety Drive Naming Scheme. Combined with an I/O or Drive configurator these names can be generated automatically for the bits in the drive's status / control word, matching the drive's profile (symbolic names).

In this way PLCopen provides all these SafeMotion functionalities, while it covers multiple instances of the same function on the same drive, e.g. 2 x SLS with different velocities. And it allows the application to distinguish the activation of SS1 from the activation of STO, and also the activation of SS2 from the activation SOS, in drives that supports these functions separately (*Note, the specialized PLCopen FB SF_SafeStop1 does not guarantee activating SS1, but might activate STO, depending on the drive, and SF_SafeStop2 does not guarantee activating SS2, but might activate SOS. The application cannot choose which one should be activated in the current situation.)*
Also for the supplier the certification process gets much simpler, while the user gets a consistent user interface that supports all needed functionalities, and even can group the same functionalities easily to provide a better overview of the safety application program.

4.2. *The Safety Drive Naming Scheme*

In order to have a consistent interface, the following naming scheme is proposed. For every safety drive *d* and every safety drive function *f* supported by the drive according to its profile and configuration, the corresponding FB instance and I/O signals shall be given the following names:

	if *d* has one instance of *f*	if *d* has multiple instances of *f*
Name of the SF_SafetyRequest instance	SF_<*d*><*f*>**_Mon**	SF_<*d*><*f*><*k*>**_Mon**
Symbolic name of the bit in the input image indicating that the drive function is now active	S_<*d*><*f*>**_fdbk**	S_<*d*><*f*><*k*>**_fdbk**
Symbolic name of the bit in the output image by which the drive function is requested	S_<*d*><*f*>**_ctrl**	S_<*d*><*f*><*k*>**_ctrl**

where

d is the name given to the safety drive in the application;

f is the acronym for the safety drive function; for standard safety drive functions the following common acronyms shall be used:
f ∈ { STO, SS1, SS2, SOS, SBC, SLA, SLI, SLP, SLS, SLT, SAR, SSR, STR, SDI, SEL, SCA, SSM, SMT };

k is the number of the instance of the safety drive function (if the drive is configured to have multiple instances of the same drive instance.

Hereunder an overview of the different names for the different safety functionalities as example:

Function	**English**	**Instance Name**	**InputNames**	**OutputNames**
STO	Safe torque off	SF_Axis1STO_Mon	S_Axis1STO _fdbk	S_Axis1STO _ctrl
SS1	Safe stop 1	SF_Axis2SS1_Mon	S_Axis2SS1_fdbk	S_Axis2SS1_ctrl
SS2	Safe stop 2	SF_Axis3SS21_Mon	S_Axis3SS21_fdbk	S_Axis3SS21_ctrl
SOS	Safe operating stop	SF_Axis1SOS2_Mon	S_Axis1SOS2_fdbk	S_Axis1SOS2_ctrl
SLA	Safely-limited acceleration	SF_Axis2SLA_Mon	S_Axis2SLA_fdbk	S_Axis2SLA_ctrl
SAR	Safe acceleration range	SF_Axis1SAR2_Mon	S_Axis1SAR2_fdbk	S_Axis1SAR2_ctrl
SLS	Safely-limited Speed	SF_Axis2SLS_Mon	S_Axis2SLS_fdbk	S_Axis2SLS_ctrl
SSR	Safe speed range	SF_Axis1SSR1_Mon	S_Axis1SSR1_fdbk	S_Axis1SSR1_ctrl
SLT	Safely-limited torque	SF_Axis2SLT_Mon	S_Axis2SLT_fdbk	S_Axis2SLT_ctrl
STR	Safe torque range	SF_Axis1STR_Mon	S_Axis1STR_fdbk	S_Axis1STR_ctrl
SLP	Safely-limited position	SF_Axis3SLP_Mon	S_Axis3SLP_fdbk	S_Axis3SLP_ctrl
SLI	Safely-limited increment	SF_Axis4SLI2_Mon	S_Axis4SLI2_fdbk	S_Axis4SLI2_ctrl
SDI	Safe direction	SF_Axis2SDI_Mon	S_Axis2SDI_fdbk	S_Axis2SDI_ctrl
SMT	Safe motor temperature	SF_Axis1SMT_Mon	S_Axis1SMT_fdbk	S_Axis1SMT_ctrl
SBC	Safe Brake Control	SF_Axis2SBC1_Mon	S_Axis2SBC1_fdbk	S_Axis2SBC1_ctrl
SCA	Safe cam	SF_Axis1_SCA_Mon	S_Axis1_SCA_fdbk	S_Axis1_SCA_ctrl
SSM	Safe speed monitor	SF_Axis2SSM_Mon	S_Axis2SSM_fdbk	S_Axis2SSM_ctrl

Abbreviations

S … Speed and/or Safety (as first letter) T … Torque, Force or Temperature

Note: the feedback input for SS1 and SS2 may not be implemented by a direct mapping to a corresponding status bit in the different profiles, but need to be assembled from different status signals.

4.3. An example of the mapping to the safe profile

As an example on how to map this functionality to a drive / axes via the applicable safe profile, we look at a CIP-safety-of-Sercos drive "MyAxis" that has its Safe Motion Monitoring function #1 configured to be the "safely limited speed" function. Then the SF_SafetyRequest instance monitoring this function on MyAxis shall have the name "MyAxisSLS_Mon", bit #2 of MyAxis's status word in the input image shall have the symbolic name "MyAxisSLS_fdbk", and bit #3 of MyAxis's control word in the output image shall have the symbolic name "MyAxisSLS_ctrl".

4.4. Example of using one instance for multiple axes

There can be linked more than one axis to the same instance of SF_SafetyRequest and in this way provide the same safety functionality to all related axes. As shown in Figure 13: Programming example with multiple axes for the SOS functionality, the different axes are linked to the same output S_SafetyRequest, and the corresponding feedback signals are grouped via the AND functionality to the S_Acknowledge, closing the safety loop. In case the one of the drives does not confirm the correct status within the specified monitoring time, the function block will set the safe mode.

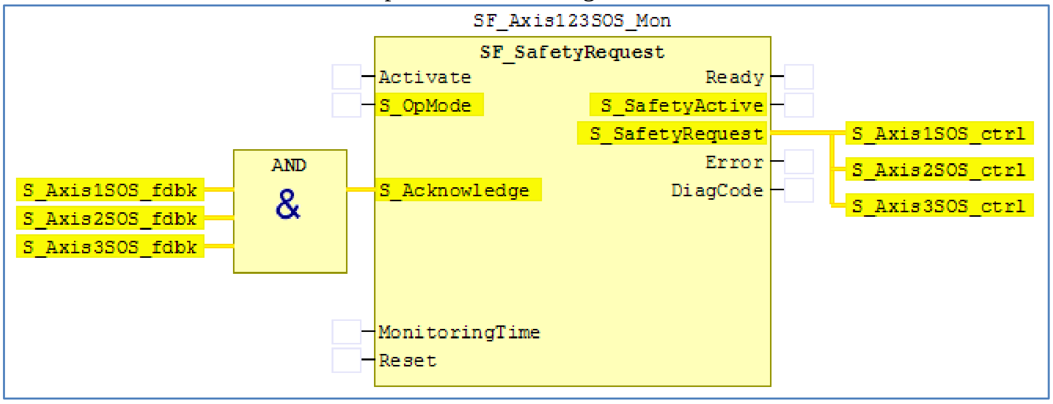


Figure 13: Programming example with multiple axes with same SafeMotion functionality

With this principle all axes that will have the same SafeMotion functionality can be grouped, and thus limiting the size of the program while enhancing the overview.

A safety drive can include multiple SafeMotion modes which can be switched during operation, meaning that they are not hard-wired. In the case of multiple function requests, priorities have to be set, which shall be according to the following table, starting with the highest priority:

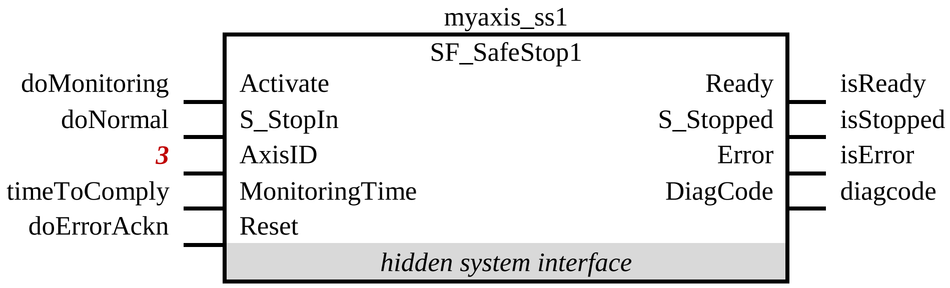
Priority	Function
1	STO
2	SS1
3	SS2
4	SLS
	SOS
	SDI
	SLI

4.5. *Example of safe stop and safely limited speed*

PLCopen Safety Software part 1 specifies three motion-specific function blocks with a hidden system-level interface to the drive. These too may be substituted by SF_SafetyRequest in accordance with the naming scheme.

4.5.1. Substituting SF_SafeStop1

SF_SafeStop1 initiates a controlled stop of an electrical drive in accordance with stop category 1 - with power available to the machine actuators to achieve the stop and then removal of the power when the stop is achieved.



A use of the SF_SafeStop1 instance as above can be substituted by a use of SF_SafetyRequest as follows:

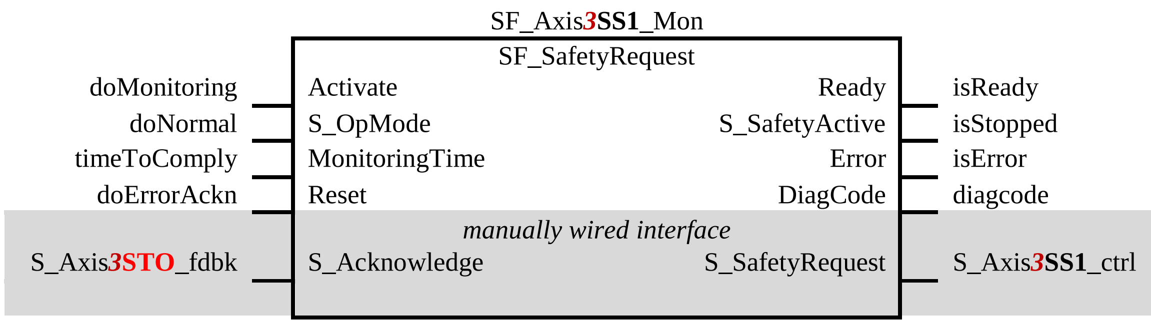


Figure 14: Mapping of SF_SafeStop1

Note: Drive profiles do not provide a specific status bits for the successful completion of the *safe stop 1* request. The request is completed successfully when the target state "safe torque off" of the *safe stop 1* request has been reached. Hence SF_SafetyRequest has to wired to the STO status bit. (Profisafe has an SS1_Active status, but it indicates the beginning of safe stopping, not its successful completion).

4.5.2. Substituting SF_SafeStop2

Safe stop 2 differs from safe stop 1 in the target status. The *safe stop 2* request has been completed successfully when the state "safe operating stop" has been reached. (Profisafe has an SS2_Active status, but it indicates the beginning of safe stopping, not its successful completion). Correspondingly, the safety output of SF_SafeStop2 is called S_Standstill instead of S_Stopped. Hence the substitute for SF_SafeStop2 looks as follows:

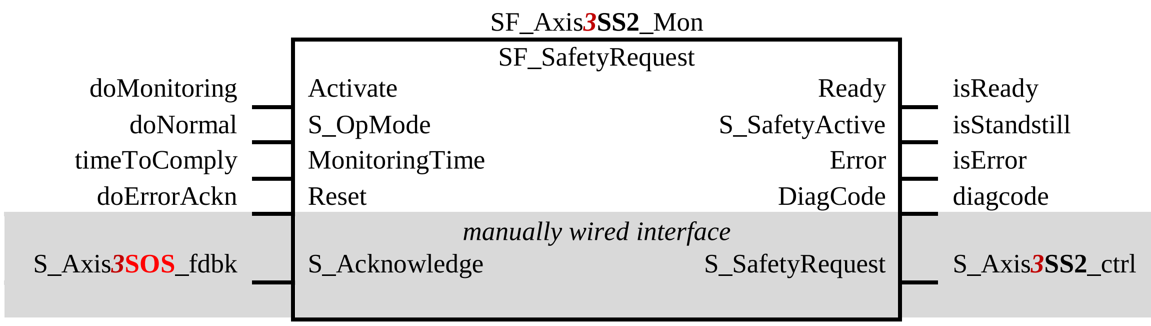
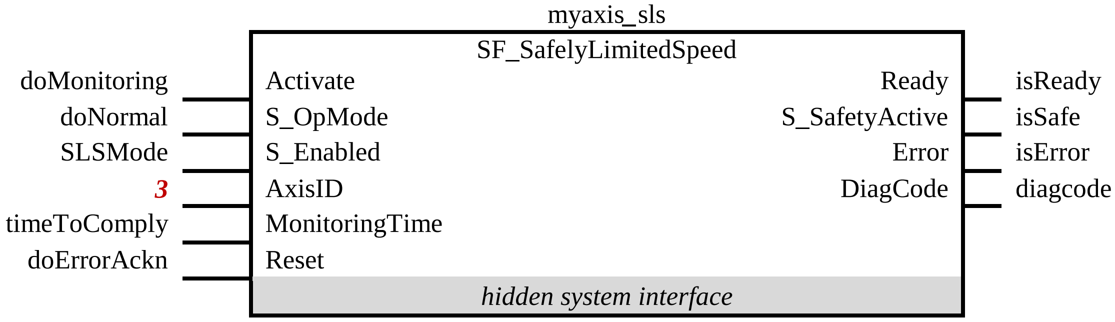


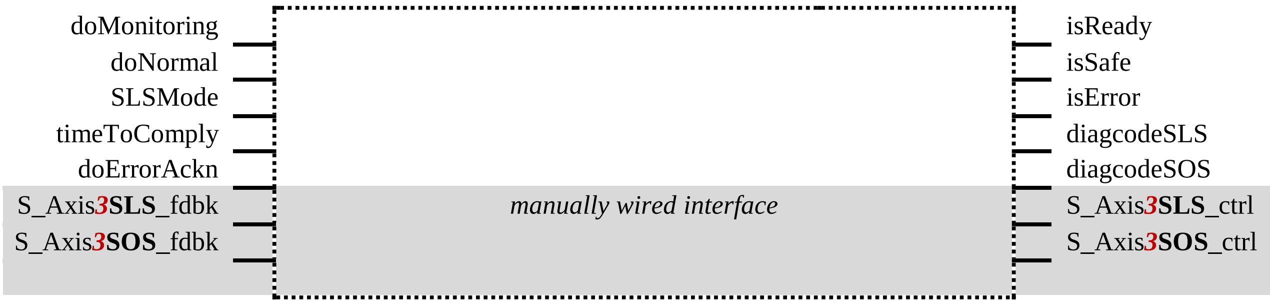
Figure 15: Mapping of SF_SafeStop2

4.5.3. Substituting SF_SafelyLimitedSpeed

SF_SafelyLimitedSpeed acts as the interface between the application program and the system environment. The supplier-specific details of the axis safety function are implemented on the system level and are hidden from the application programmer.



Substituting SF_SafelyLimitedSpeed is more complicated because it combines two drive functions in one function block: Normally it activates the *safely limited speed* function (S_Enabled = true). But if S_Enabled = false, it activates the *safe operating stop* function (effectively a speed limited of zero).
Hence the substitution of a SF_SafetyLimitedSpeed instance requires the manual wiring of *two* status and control bits: those for the drive's *safely limited speed* function and those for its *safe operating stop* function:



And implementing the functionality of SF_SafelyLimitedSpeed requires two instances of SF_SafetyRequest and their connection with the input and output signals.

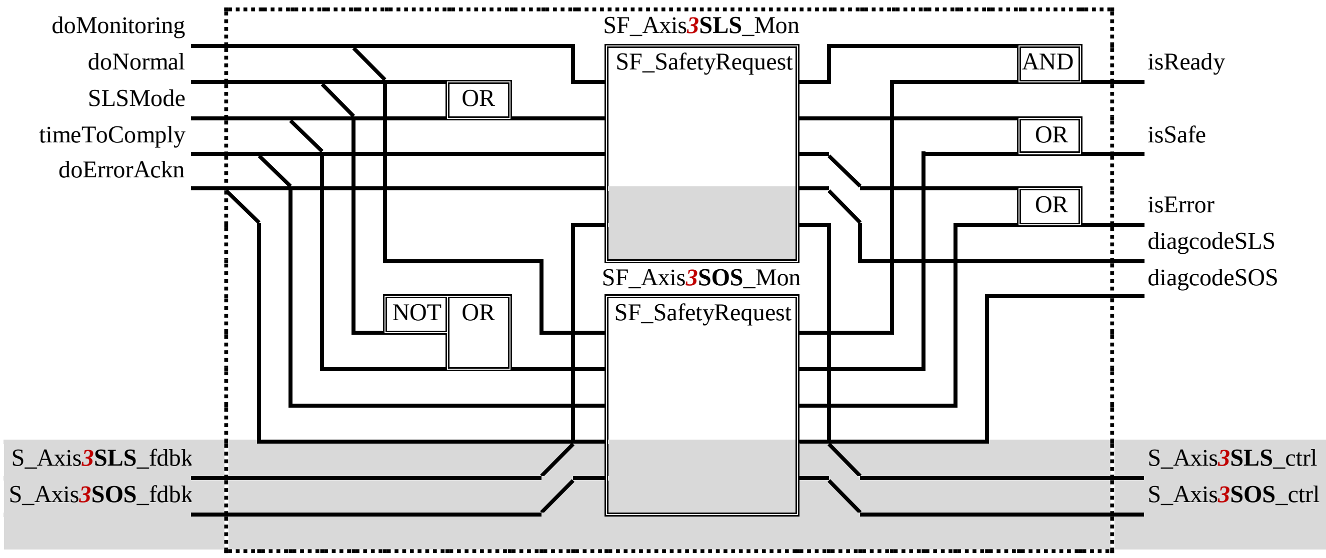


Figure 16: Mapping of SF_SafetyLimitedSpeed

4.6. Example of mapping on the OMAC specification

The naming convention for the symbolic I/O variables can be mapped to the different protocols via aliases. For example, if we look to the OMAC control and status bytes, this can be done in the following way for the example above:
Enabled AND OpMode in the SF_SafetyLimitedSpeed releases the safety functionality, so the output parameter S_Axis1SOS_ctrl bit is linked to the bit number 3 in the control byte. So there is a first drive called Drive1, which is mapped to the bit Drive1.SOS, or alias Axis1.SOS.

(Just to help here a copy of the OMAC control and status words:

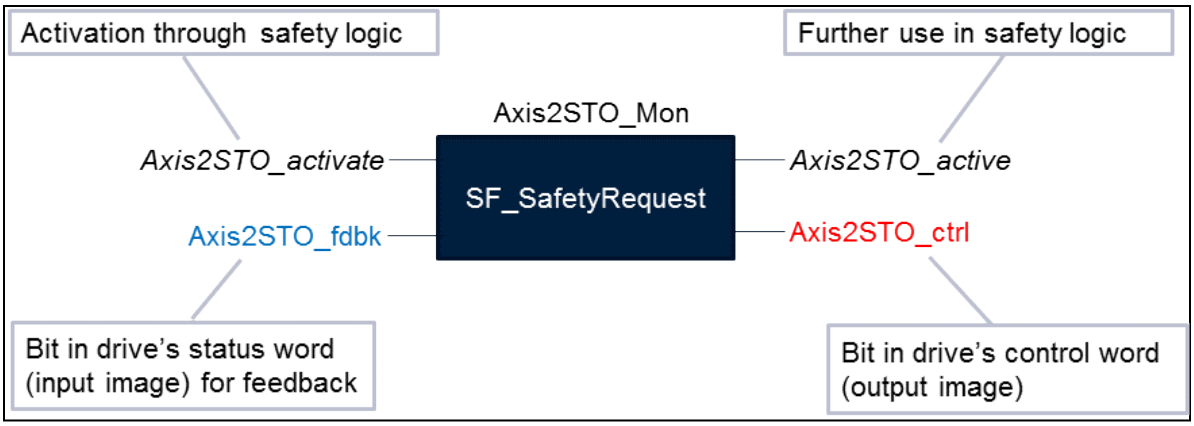
Safety Control Byte

Bit	Name	Description
0	STO	Safe Torque Off 0: activate 1: deactivate
1	SS1	Safe Stop 1 0: activate 1: deactivate
2	SS2	Safe Stop 2 0: activate 1: deactivate
3	SOS	Safe Operating Stop 0: activate 1: deactivate
4	SLS	Safely-limited Speed 0: activate 1: deactivate
5	Safety Function 1	Machine specific safety function e.g. SLA, SAR, SLT, SLP, SDI[2] 0: activate 1: deactivate
6	Safety Function 2	Machine specific safety function e.g. SLA, SAR, SLT, SLP, SDI 0: activate 1: deactivate
7	Error Ack	Error Acknowledge 0: no acknowledge 0→1: Acknowledge of an error in the drive

Safety Status Byte

Bit	Name	Description
0	STO active	Safe Torque off 0: is not active 1: is active
1	---	Note: Bit SS1 is not part of the Safety Statusbyte. During deceleration part of the function the machine is still in movement, the status SS1 can be part of standard status information. At the end the resulting stop function STO is reflected in Bit 0
2	---	Note: Bit SS2 is not part of the Safety Statusbyte. During deceleration part of the function the machine is still in movement, the status SS2 can be part of standard status information. At the end the resulting stop function SOS is reflected in Bit 3
3	SOS active	Safe Operating Stop 0: is not active 1: is active
4	SLS active	Safely-limited Speed 0: is not active 1: is active and speed of the drive is within the safe speed range
5	Safety Function 1 active	Machine specific safety function 1 active 0: is not active 1: is active
5	Safety Function 2 active	Machine specific safety function 2 active 0: is not active 1: is active
7	Error indication	Error indication 0: no Error 1: at least one safety-related error has occurred

If we map for example the Safe Torque Off functionality, it is the first bit in both the control and status byte. If we use Axis 2 for this example, the mapping of SF_SafetyRequest could look like:



Axis2STO_ctrl would be mapped to the first bit in safety control byte in the OMAC protocol.
In this way also the feedback is created via coupling to the first bit of the Safety Status Byte.

5 Safe Profiles

5.1. OMAC

OMAC The Organization for Machine Automation and Control	PackSafety *Machine Safety Interface*	Doc. ID: OMAC PackSafety Rev. 0.3 Date : 25.04.2013

Check www.OMAC.org

The organization OMAC considers complete production lines where the safety aspects are not just limited to one machine, but these are interconnected. This can be done with a centralized or de-centralized approach.

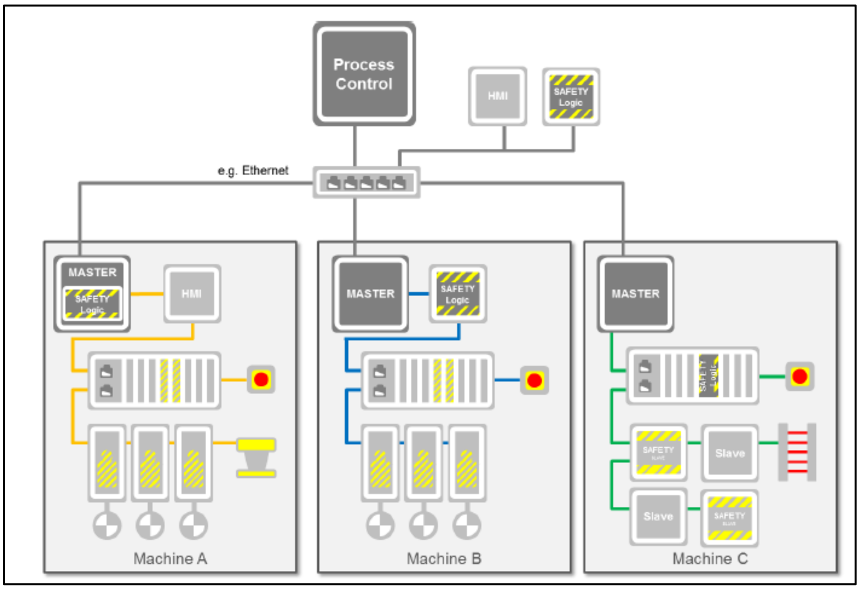


Figure 17: Plant wide safety machine architecture

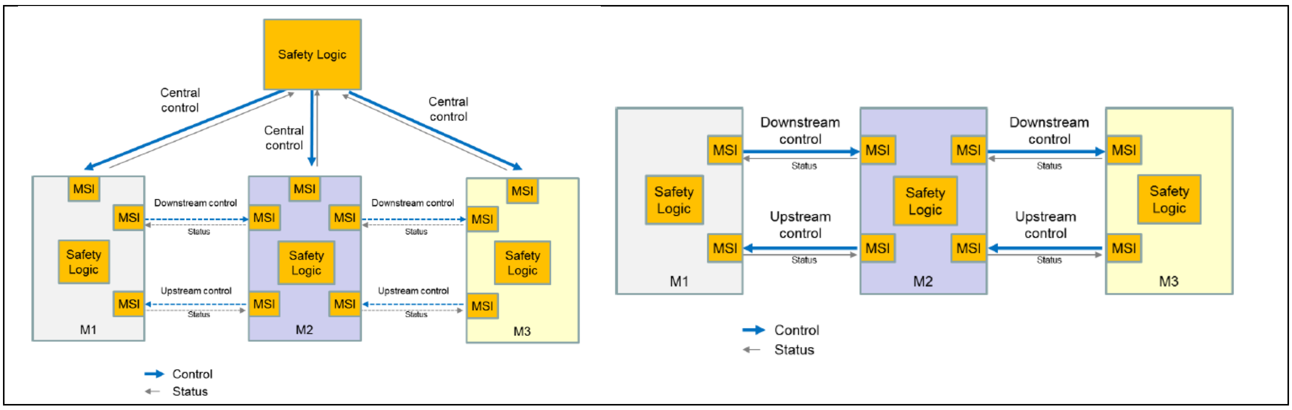


Figure 18: Centralized versus de-centralized safety solution

Safety Control Byte

Bit	Name	Description
0	STO	Safe Torque Off 0: activate 1: deactivate
1	SS1	Safe Stop 1 0: activate 1: deactivate
2	SS2	Safe Stop 2 0: activate 1: deactivate
3	SOS	Safe Operating Stop 0: activate 1: deactivate
4	SLS	Safely-limited Speed 0: activate 1: deactivate
5	Safety Function 1	Machine specific safety function e.g. SLA, SAR, SLT, SLP, SDI[2] 0: activate 1: deactivate
6	Safety Function 2	Machine specific safety function e.g. SLA, SAR, SLT, SLP, SDI 0: activate 1: deactivate
7	Error Ack	Error Acknowledge 0: no acknowledge 0→1: Acknowledge of an error in the drive

Safety Status Byte

Bit	Name	Description
0	STO active	Safe Torque off 0: is not active 1: is active
1	---	Note: Bit SS1 is not part of the Safety Statusbyte. During deceleration part of the function the machine is still in movement, the status SS1 can be part of standard status information. At the end the resulting stop function STO is reflected in Bit 0
2	---	Note: Bit SS2 is not part of the Safety Statusbyte. During deceleration part of the function the machine is still in movement, the status SS2 can be part of standard status information. At the end the resulting stop function SOS is reflected in Bit 3
3	SOS active	Safe Operating Stop 0: is not active 1: is active
4	SLS active	Safely-limited Speed 0: is not active 1: is active and speed of the drive is within the safe speed range
5	Safety Function 1 active	Machine specific safety function 1 active 0: is not active 1: is active
5	Safety Function 2 active	Machine specific safety function 2 active 0: is not active 1: is active
7	Error indication	Error indication 0: no Error 1: at least one safety-related error has occurred

5.2. *ProfiSafe*

(Note: the table references are to the original document)

5.2.1. Safety control word 1 (S_STW1)

Each autonomous safety drive function will be activated by a separate bit. A drive based safety function to be used in an application has to be explicitly enabled in the drive by use of parameterisation. The user data length of the signal S_STW1 is 2 bytes. The meaning of the bits 0 to 7 will be defined in this profile. The bits 8 to 15 may be used in a device specific way. In future versions, bits marked with "reserved", may be used for the activation of other autonomous safety functions of drives [7]. Reserved bits should be set to "0" as default.

The structure of the PROFIdrive Safety control word 1 is shown in Table 1.

Table 1: Overview on the assignment of the bits of Safety control word 1

Byte	Bit	Name	Significance
0	0	STO	No Safe Torque Off / Safe Torque Off (zero-active)
0	1	SS1	No Safe Stop 1 / Safe Stop 1 (zero-active)
0	2	SS2	No Safe Stop 2 / Safe Stop 2 (zero-active)
0	3	SOS	No Safe Operational Stop / Safe Operational Stop (zero-active)
0	4	SLS	No Safe Limited Speed / Safe Limited Speed (zero-active)
0	5	SLT	No Safe Limited Torque / Safe Limited Torque (zero-active)
0	6	SLP	No Safe Limited Position / Safe Limited Position (zero-active)
0	7	INTERNAL_EVENT_ACK	Safety Fault Buffer Fault Acknowledge (1 -> 0)
1	0 to 7	-	Device-specific

Explanation: The significance for bit value = 1 is to the left of the slash; bit value = 0 to the right.

5.2.2. Safety control word 2 (S_STW2)

Each autonomous safety drive function will be activated by a separate bit. A drive based safety function to be used in an application has to be explicitly enabled in the drive by use of parameterisation. The user data length of the signal S_STW2 is 4 bytes. The meaning of the bits 0 to 23 will be defined in this profile. The bits 24 to 31 may be used in a device specific way. In future versions, bits marked with "reserved", may be used for the activation of other autonomous safety functions of drives [7]. Reserved bits should be set to "0" as default.

The structure of the PROFIdrive Safety control word 2 is shown in Table 3..

Table 3: Overview on the assignment of the bits of Safety control word 2

Byte	Bit	Name	Significance
0	0	STO	No Safe Torque Off / Safe Torque Off (zero-active)
0	1	SS1	No Safe Stop 1 / Safe Stop 1 (zero-active)
0	2	SS2	No Safe Stop 2 / Safe Stop 2 (zero-active)
0	3	SOS	No Safe Operational Stop / Safe Operational Stop (zero-active)
0	4	SLS	No Safe Limited Speed / Safe Limited Speed (zero-active)
0	5	SLT	No Safe Limited Torque / Safe Limited Torque (zero-active)
0	6	SLP	No Safe Limited Position / Safe Limited Position (zero-active)

Byte	Bit	Name	Significance
0	7	INTERNAL_EVENT_ACK	Safety Fault Buffer Fault Acknowledge (1 -> 0)
1	0	-	Reserved for future use
1	1	SLS_LIMIT_BIT0	Bit 0 for selection of SLS speed limit value
1	2	SLS_LIMIT_BIT1	Bit 1 for selection of SLS speed limit value
1	3	-	Reserved for future use
1	4	SDI_P	Safe Direction Positive inactive / Safe Direction Positive active, only positive speed (zero-active)
1	5	SDI_N	Safe Direction Negative inactive / Safe Direction Negative active, only negative speed (zero-active)
1	6	-	Reserved for future use
1	7	-	Reserved for future use
2	0	SLT_LIMIT_BIT0	Bit 0 for selection of SLT torque limit value
2	1	SLT_LIMIT_BIT1	Bit 1 for selection of SLT torque limit value
2	2	-	Reserved for future use
2	3	SLP_LIMIT_BIT0	Bit 0 for selection of SLP Position limit value set
2	4	SLP_LIMIT_BIT1	Bit 1 for selection of SLP Position limit value set
2	5 to 7	-	Reserved for future use
3	0 to 7	-	Device-specific

Explanation: The significance for bit value = 1 is to the left of the slash; bit value = 0 to the right.

5.2.3. Safety status word 1 (S_ZSW1)

Different states of autonomous safety drive functions will be reported by different bits in the PROFIdrive Safety status word 1. An autonomous safety drive function to be used in an application has to be enabled explicitly in the drive by use of parameterization. The user data length of the „PROFIdrive Safety status word 1“ is 2 bytes. The meaning of the bits 0 to 7 will be defined in this profile. The bits 8 to 15 may be used in a manufacturer specific way. Bits which are marked 'reserved' or bits which are not in use have to be set to "0".

The structure of the PROFIdrive Safety status word 1 is shown in Table 5.

Table 5: Overview on the assignment of the bits of Safety status word 1

Byte	Bit	Name	Significance
0	0	POWER_REMOVED	Safe Torque Off active / Safe Torque Off inactive (one-active)
0	1	SS1_ACTIVE	Safe Stop 1 active / Safe Stop 1 inactive (one-active)
0	2	SS2_ACTIVE	Safe Stop 2 active / Safe Stop 2 inactive (one-active)
0	3	SOS_ACTIVE	Safe Operational Stop active / Safe Operational Stop inactive (one-active)
0	4	SLS_ACTIVE	Safe Limited Speed active / Safe Limited Speed inactive (one-active)
0	5	SLT_ACTIVE	Safe Limited Torque active / Safe Limited Torque inactive(one-active)
0	6	SLP_ACTIVE	Safe Limited Position active / Safe Limited Positon inactive(one-active)
0	7	INTERNAL_EVENT	Safety Fault present / No Safety Fault
1	0 to7	-	Device-specific

Explanation: The significance for bit value = 1 is to the left of the slash; bit value = 0 to the right.

5.2.4. Safety status word 2 (S_ZSW2)

Different states of autonomous safety drive functions will be reported by different bits in the PROFIdrive Safety status word 2. An autonomous safety drive function to be used in an application has to be enabled explicitly in the drive by use of parameterization. The user data length of the „PROFIdrive Safety status word 2“ is 4 bytes. The meaning of the bits 0 to 23 will be defined in this profile. The bits 24 to 31 may be used in a manufacturer specific way. Bits which are marked 'reserved' or bits which are not in use have to be set to "0".

The structure of the PROFIdrive Safety status word 2 is shown in Table 7.

Table 7: Overview on the assignment of the bits of Safety status word 2

Byte	Bit	Name	Significance
0	0	POWER_REMOVED	Safe Torque Off active / Safe Torque Off inactive (one-active)
0	1	SS1_ACTIVE	Safe Stop 1 active / Safe Stop 1 inactive (one-active)
0	2	SS2_ACTIVE	Safe Stop 2 active / Safe Stop 2 inactive (one-active)
0	3	SOS_ACTIVE	Safe Operational Stop active / Safe Operational Stop inactive (one-active)
0	4	SLS_ACTIVE	Safe Limited Speed active / Safe Limited Speed inactive (one-active)
0	5	SLT_ACTIVE	Safe Limited Torque active / Safe Limited Torque inactive (one-active)
0	6	SLP_ACTIVE	Safe Limited Position active / Safe Limited Positon inactive (one-active)
0	7	INTERNAL_EVENT	Safety Fault present / No Safety Fault
1	0	-	Reserved for future use
1	1	SLS_LIMIT_BIT0_ACTIVE	Bit 0 of selected SLS speed limit value
1	2	SLS_LIMIT_BIT1_ACTIVE	Bit 1 of selected SLS speed limit value
1	3	-	Reserved for future use
1	4	SDI_P_ACTIVE	Safe Direction Positive active / Safe Direction Positive inactive, only positive speed (on-active)
1	5	SDI_N_ACTIVE	Safe Direction Negative active / Safe Direction Negative inactive, only negative speed (on-active)
1	6	-	Reserved for future use
1	7	SSM	Speed below limit value / Speed equal or above limit value
2	0	SLT_LIMIT_BIT0_ACTIVE	Bit 0 of selected SLT torque limit value
2	1	SLT_LIMIT_BIT1_ACTIVE	Bit 1 of selected SLT torque limit value
2	2	-	Reserved for future use
2	3	SLP_LIMIT_BIT0_ACTIVE	Bit 0 of selected SLP Position limit value set
2	4	SLP_LIMIT_BIT1_ACTIVE	Bit 1 of selected SLP Position limit value set
2	5	-	Reserved for future use
2	6	SP_VALID	Safe Position in S_XISTx is valid / Safe Position in S_XISTx is not valid
2	7	SP_REF	Safe Position in S_XISTx is referenced / Safe Position in S_XISTx is not referenced

Byte	Bit	Name	Significance
3	0 to 7	-	Device-specific

Explanation: The significance for bit value = 1 is to the left of the slash; bit value = 0 to the right.

[Note from PLCopen: In Safety Status Word 1 & 2 the bit SS1_ACTIVE reflects that the activity safe stop is active, which does not have to mean that the drive reached the safe torque off. This is also valid for SS2_ACTIVE and safe stop reached. This means that the input for SF_SafetyRequest can be assembled form: S_Axis2SS1_fdbk := SS1_Active AND Power Removed]

5.3. Safety over EtherCAT

(Note: the table references are to the original document)

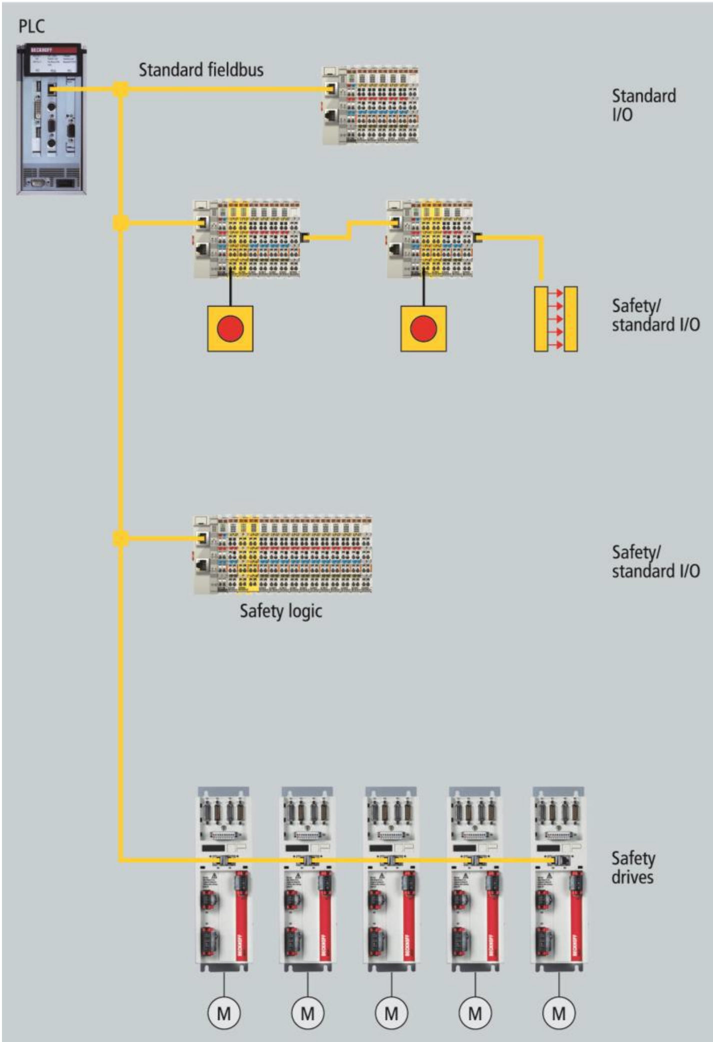


Table 3: Safety controlword: 1st byte

Bit	Name	Description
0	STO	Safe Torque Off 0: activate 1: deactivate
1	SS1_1	Safe Stop 1, Instance 1 0: activate 1: deactivate
2	SS2_1	Safe Stop 2, Instance 1 0: activate 1: deactivate
3	SOS_1	Safe Operating Stop, Instance 1 0: activate 1: deactivate
4	SSR_1	Safe Speed Range, Instance 1 0: activate 1: deactivate
5	SDIp	Safe Direction positive 0: disable positive direction 1: enable positive direction
6	SDIn	Safe Direction negative 0: disable negative direction 1: enable negative direction
7	Error Ack	Error Acknowledge 0: no acknowledge 0→1: Acknowledge of an error in the drive

Table 5: Safety statusword, 1st byte (fix meaning)

Bit	Name	Description
0	STO active	Safe Torque off 0: is not active 1: is active
1	SSM_1	Safe Speed Monitor 1, Instance 1 0: the drive is moving faster than the limit n_UL_SSM_1 1: the drive is moving slower or equal than the limit n_LL_SSM_1
2	SSM_2	Safe Speed Monitor 2, Instance 2 0: the drive is moving faster than the limit n_UL_SSM_2 1: the drive is moving slower or equal than the limit n_LL_SSM_2
3	SOS_1 active	Safe Operating Stop, Instance 1 0: is not active 1: is active
4	SSR_1 active	Safe Speed Range, Instance 1 0: is not active 1: is active and speed of the drive is within the safe speed range
5	SDIp active	Safe Direction positive 0: Drive does not move in positive direction 1: Drive moves in positive direction
6	SDIn active	Safe Direction negative 0: Drive does not move in negative direction 1: Drive moves in negative direction
7	Error	Error 0: no Error 1: at least one safety error has occurred

5.4. OpenSafety

Basic Set:

Control	Description	Status	Bit
Reset	Reset	Not error function	0
Activate	Activate	Operational	1
STO	Safe Torque Off	STO	2
SBC	Safe Brake Control	SBC	3
SS1	Safe Stop 1	SS1	4
Reserved		Reserved	5
Vendor specific		Vendor specific	6
Vendor specific		Vendor specific	7

Speed Extensions:

Control		Status	Bit
SOS	Safe Operating Stop	SOS	0
SS2	Safe Stop 2	SS2	1
SLA	Safely Limited Acceleration	SLA	2
SLS-1	Safely Limited Speed 1	SLS-1	3
SLS2	Safely Limited Speed 2	SLS-2	4
Reserved		Reserved	5
Vendor specific		Vendor specific	6
Vendor specific		Vendor specific	7

Advance Extension:

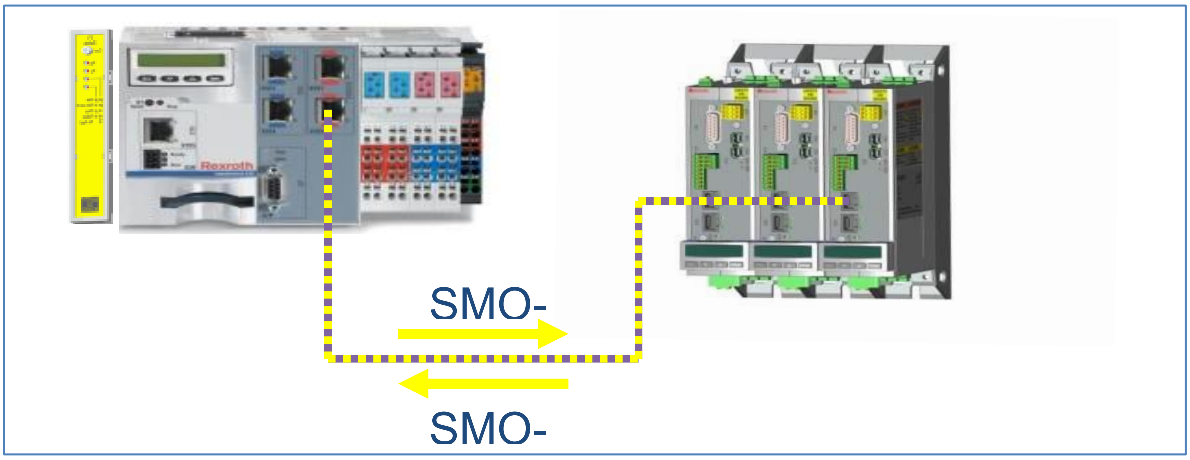
Control		Status	Bit
SDI-P	Safe Direction Positive	SDI-P	0
SDI-N	Safe Direction Negative	SDI-N	1
SLI	Safely Limited Increment	SLI	2
SBT	Safe Brake Test	SBT	3
Reserved		SBT is valid	4
SLT	Safely Limited Torque	SLT	5
Vendor specific		Vendor specific	6

Vendor specific		Vendor specific	7

Encoder Basic Set:

Control	Description	Status	Bit
Reserved	Encoder Status	Encoder Status	0
Homing	Homing	Homing	1
RefSwitch	Reference Switch	ReqHomingOK	2
SLP	Safely Limited Position	SLP	3
Reserved		Reserved	4
Reserved		Reserved	5
Vendor specific		Vendor specific	6
Vendor specific		Vendor specific	7

5.5. *CIP Safety over Sercos*



Definitions of Control-bits

Input combination	Value	SafeMotion-Description	Function
EmergencyStopSignal	0	SMES	Safe Motion Emergency Stop
EmergencyStopSignal Enabling Control ModeSelectionSignal	1 0 0	SMSS1 SMSS2	Safe Motion Safe Stop 1[1] Safe Motion Safe Stop 2[1]
EmergencyStopSignal Enabling Control ModeSelectionSignal	1 1 1	None	Safe Motion deaktiviert
EmergencyStopSignal Enabling Control ModeSelectionSignal SMMx	1 1 0 1	SMMx	Safe Motion Mode 1[2]

Possible Conventions for a SMM:

Abbreviation	SafeMotion Monitoring function
SLS	Safe Limited Speed
SDI	Safe Direction
SLI	Safely Limited Increment
SLT	Safely Limited Torque
SLP	Safely Limited Position
SSM	Safe Speed Monitor
SCA	Safe CAM

Assembly 778 / 782: 1 Word Binary-Coded SMM

Variable	Kanal	Datentyp	Einheit	Beschreibung
⊟	Output	STRUCT		1B-Control Bit-SMM
ModeSelectionSignal	bit0	SAFEBOOL		ModeSelectionSignal
EmergencyStopSignal	bit1	SAFEBOOL		EmergencyStopSignal
EnablingControl	bit2	SAFEBOOL		EnablingControl
SMM1Signal	bit3	SAFEBOOL		SMM1Signal
SMM2Signal	bit4	SAFEBOOL		SMM2Signal
SMM3Signal	bit5	SAFEBOOL		SMM3Signal
SMM4Signal	bit6	SAFEBOOL		SMM4Signal
SafeOutput_local	bit7	SAFEBOOL		SafeOutput_local

Control

Variable	Kanal	Datentyp	Einheit	Beschreibung
⊟	Input	STRUCT		1B-Status Bit-SMM
SMESStatus	bit0	SAFEBOOL		SMESStatus
SMSTStatus	bit1	SAFEBOOL		SMSTStatus
SMM1Status	bit2	SAFEBOOL		SMM1Status
SMM2Status	bit3	SAFEBOOL		SMM2Status
SMM3Status	bit4	SAFEBOOL		SMM3Status
SMM4Status	bit5	SAFEBOOL		SMM4Status
EncoderStandstill	bit6	SAFEBOOL		EncoderStandstill
SafetyStatus	bit7	SAFEBOOL		SafetyStatus

Status

Assembly 779 / 783 - 1 Word Binary-Coded SMM

Variable	Kanal	Datentyp	Einheit	Beschreibung
⊟	Output	STRUCT		1B-Control Bin-SMM
ModeSelectionSignal	bit0	SAFEBOOL		ModeSelectionSignal
EmergencyStopSignal	bit1	SAFEBOOL		EmergencyStopSignal
EnablingControl	bit2	SAFEBOOL		EnablingControl
SMM1Signal_coded	bit3	SAFEBOOL		SMM1Signal_coded
SMM2Signal_coded	bit4	SAFEBOOL		SMM2Signal_coded
SMM3Signal_coded	bit5	SAFEBOOL		SMM3Signal_coded
ReleaseBrake	bit6	SAFEBOOL		ReleaseBrake
SafeOutput_local	bit7	SAFEBOOL		SafeOutput_local

Control

Variable	Kanal	Datentyp	Einheit	Beschreibung
⊟	Input	STRUCT		1B-Status Bin-SMM
SMESStatus	bit0	SAFEBOOL		SMESStatus
SMSTStatus	bit1	SAFEBOOL		SMSTStatus
SMM1Status_coded	bit2	SAFEBOOL		SMM1Status_coded
SMM2Status_coded	bit3	SAFEBOOL		SMM2Status_coded
SMM3Status_coded	bit4	SAFEBOOL		SMM3Status_coded
SafetyError	bit5	SAFEBOOL		SafetyError
EncoderStandstill	bit6	SAFEBOOL		EncoderStandstill
SafetyStatus	bit7	SAFEBOOL		SafetyStatus

Status

Assembly 780 / 784 - 1 Word Bit-Coded SMM

Variable	Kanal	Datentyp	Einheit	Beschreibung
⊟	Output	STRUCT		2B-Control Bit-SMM
ModeSelectionSignal	bit0	SAFEBOOL		ModeSelectionSignal
EmergencyStopSignal	bit1	SAFEBOOL		EmergencyStopSignal
EnablingControl	bit2	SAFEBOOL		EnablingControl
SMM1Signal	bit3	SAFEBOOL		SMM1Signal
SMM2Signal	bit4	SAFEBOOL		SMM2Signal
SMM3Signal	bit5	SAFEBOOL		SMM3Signal
SMM4Signal	bit6	SAFEBOOL		SMM4Signal
SMM5Signal	bit7	SAFEBOOL		SMM5Signal
SMM6Signal	bit8	SAFEBOOL		SMM6Signal
SMM7Signal	bit9	SAFEBOOL		SMM7Signal
SMM8Signal	bit10	SAFEBOOL		SMM8Signal
SMM9Signal	bit11	SAFEBOOL		SMM9Signal
SMM10Signal	bit12	SAFEBOOL		SMM10Signal
SMM11Signal	bit13	SAFEBOOL		SMM11Signal
SMM12Signal	bit14	SAFEBOOL		SMM12Signal
SafeOutput_local	bit15	SAFEBOOL		SafeOutput_local

Control

Variable	Kanal	Datentyp	Einheit	Beschreibung
⊟	Input	STRUCT		2B-Status Bit-SMM
SMESStatus	bit0	SAFEBOOL		SMESStatus
SMSTStatus	bit1	SAFEBOOL		SMSTStatus
SMM1Status	bit2	SAFEBOOL		SMM1Status
SMM2Status	bit3	SAFEBOOL		SMM2Status
SMM3Status	bit4	SAFEBOOL		SMM3Status
SMM4Status	bit5	SAFEBOOL		SMM4Status
SMM5Status	bit6	SAFEBOOL		SMM5Status
SMM6Status	bit7	SAFEBOOL		SMM6Status
SMM7Status	bit8	SAFEBOOL		SMM7Status
SMM8Status	bit9	SAFEBOOL		SMM8Status
SMM9Status	bit10	SAFEBOOL		SMM9Status
SMM10Status	bit11	SAFEBOOL		SMM10Status
SMM11Status	bit12	SAFEBOOL		SMM11Status
SMM12Status	bit13	SAFEBOOL		SMM12Status
EncoderStandstill	bit14	SAFEBOOL		EncoderStandstill
SafetyStatus	bit15	SAFEBOOL		SafetyStatus

Status

Assembly 781 / 785 - 1 Word Binary-Coded SMM

Variable	Kanal	Datentyp	Einheit	Beschreibung
⊟	Output	STRUCT		2B-Control Bin-SMM
ModeSelectionSignal	bit0	SAFEBOOL		ModeSelectionSignal
EmergencyStopSignal	bit1	SAFEBOOL		EmergencyStopSignal
EnablingControl	bit2	SAFEBOOL		EnablingControl
SMM1Signal_coded	bit3	SAFEBOOL		SMM1Signal_coded
SMM2Signal_coded	bit4	SAFEBOOL		SMM2Signal_coded
SMM3Signal_coded	bit5	SAFEBOOL		SMM3Signal_coded
SMM4Signal_coded	bit6	SAFEBOOL		SMM4Signal_coded
	bit7	SAFEBOOL		
	bit8	SAFEBOOL		
	bit9	SAFEBOOL		
	bit10	SAFEBOOL		
	bit11	SAFEBOOL		
	bit12	SAFEBOOL		
	bit13	SAFEBOOL		
ReleaseBrake	bit14	SAFEBOOL		ReleaseBrake
SafeOutput_local	bit15	SAFEBOOL		SafeOutput_local

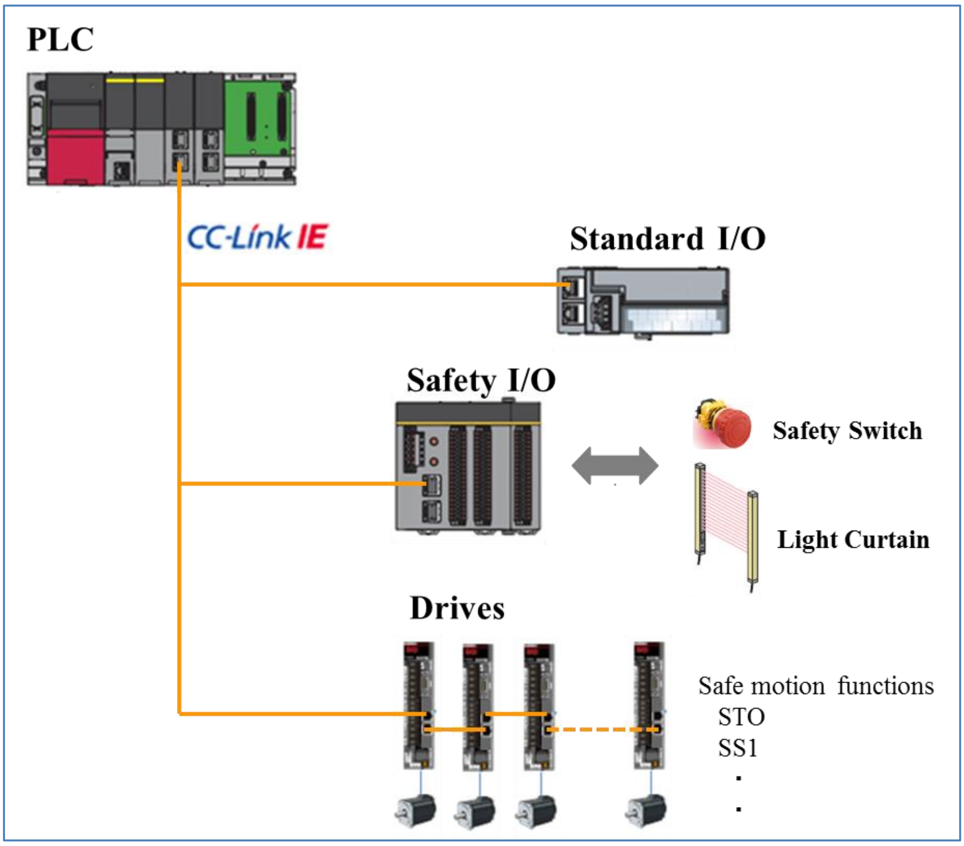
Control

Variable	Kanal	Datentyp	Einheit	Beschreibung
⊟	Input	STRUCT		2B-Status Bin-SMM
SMESStatus	bit0	SAFEBOOL		SMESStatus
SMSTStatus	bit1	SAFEBOOL		SMSTStatus
SMM1Status_coded	bit2	SAFEBOOL		SMM1Status_coded
SMM2Status_coded	bit3	SAFEBOOL		SMM2Status_coded
SMM3Status_coded	bit4	SAFEBOOL		SMM3Status_coded
SMM4Status_coded	bit5	SAFEBOOL		SMM4Status_coded
	bit6	SAFEBOOL		
	bit7	SAFEBOOL		
	bit8	SAFEBOOL		
	bit9	SAFEBOOL		
	bit10	SAFEBOOL		
	bit11	SAFEBOOL		
BrakeStatus	bit12	SAFEBOOL		BrakeStatus
SafetyError	bit13	SAFEBOOL		SafetyError
EncoderStandstill	bit14	SAFEBOOL		EncoderStandstill
SafetyStatus	bit15	SAFEBOOL		SafetyStatus

Status

5.6. CC-Link IE

Safety Communication



Safety PDU structure

The structure of the CC-Link IE Safety PDU (Protocol Data Unit) is shown in Figure 19: Safety PDU structure. S-Data in the PDU contains the safety input data or the safety output data. The size of S-Data area is 1..16 octet length.

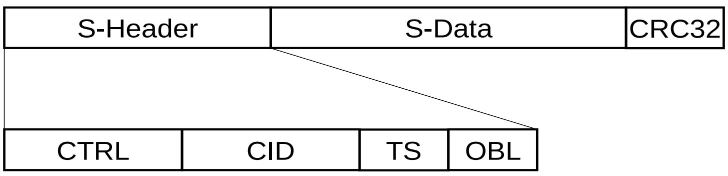


Figure 19: Safety PDU structure

Name, size, and contents of the elements contained in the safety PDU are shown in Table 1.

Table 1 Element of safety PDU

Name		Size	Description
S-Header		96 bits	Safety Protocol Information
	CTRL	32 bits	Command type, state etc.
	CID	32 bits	Connection ID
	TS	16 bits	Time stamp
	OBL	16 bits	Offset baseline
S-Data		1..16 octet	Safety data
CRC32		32 bits	32 bit CRC for safety communication

S-Data

S-Data is used to store safety data.
The structure of S-Data for safety refresh is shown in Figure 20: Structure of S-Data for safety refresh.

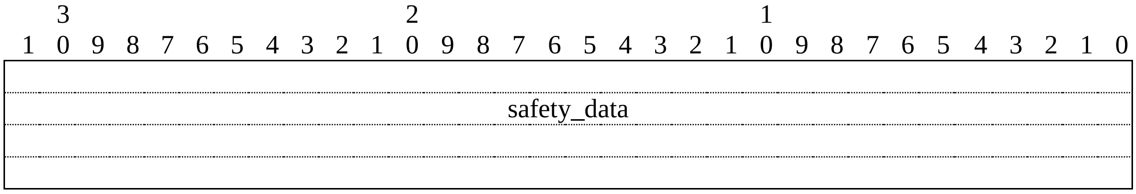


Figure 20: Structure of S-Data for safety refresh

Safety control word

The Safety data which are sent from Master to Slave are based on Safety control word shown below.

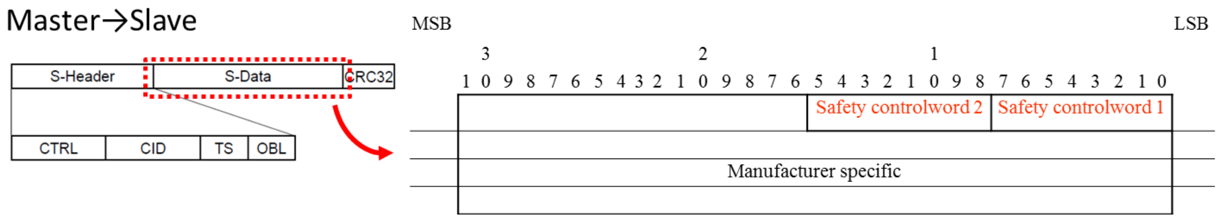


Table 2: Safety controlword 1

Bit	Name	Description	
0	STO	Safe Torque Off	0: activate 1: deactivate
1	SS1_1	Safe Stop 1, Instance 1	0: activate 1: deactivate
2	SS2_1	Safe Stop 2, Instance 1	0: activate 1: deactivate
3	SOS_1	Safe Operating Stop, Instance 1	0: activate 1: deactivate
4	SSR_1	Safe Speed Range, Instance 1	0: activate 1: deactivate
5	SDIp	Safe Direction positive	0: activate positive direction 1: deactivate positive direction
6	SDIn	Safe Direction negative	0: activate negative direction 1: deactivate negative direction
7	Error Ack	Error Acknowledge	0: no acknowledge 0→1: Acknowledge of an error in the drive

Table 3: Safety controlword 2

Bit	Name	Description
0	Manufacturer Specific	0: activate 1: deactivate
1		
2		
3		
4		
5		
6		
7		

Safety status word

The Safety data which are sent from Master to Slave are based on Safety status word shown below.

Slave→Master

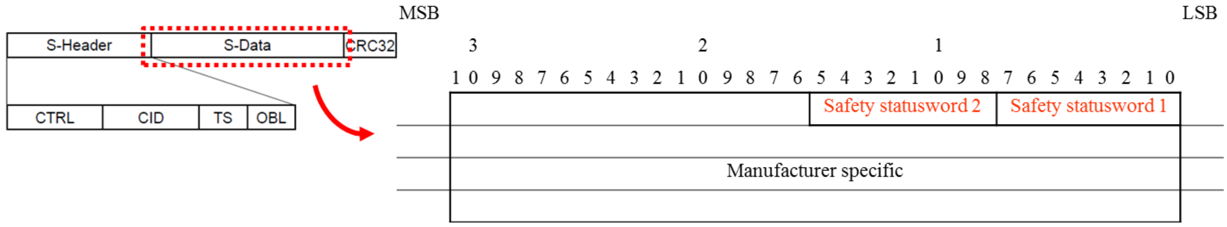


Table 4: Safety statusword 1

Bit	Name	Description	
0	STO active	Safe Torque Off	0: is not active 1: is active
1	SSM_1	Safe Speed Monitor, Instance 1	0: the drive is moving faster than the limit 1: the drive is moving slower or equal than the limit
2	SSM_2	Safe Speed Monitor, Instance 2	0: the drive is moving faster than the limit 1: the drive is moving slower or equal than the limit
3	SOS_1 active	Safe Operating Stop, Instance 1	0: is not active 1: is active
4	SSR_1 active	Safe Speed Range, Instance 1	0: is not active 1: is active
5	SDIp active	Safe Direction positive	0: Drive does not move in positive direction 1: Drive moves in positive direction
6	SDIn active	Safe Direction negative	0: Drive does not move in negative direction 1: Drive moves in negative direction
7	Error	Error	0: no error 1: at least one safety error has occurred

Table 5: Safety statusword 2

<table>
<tr><th>Bit</th><th>Name</th><th>Description</th></tr>
<tr><td>0</td><td rowspan="8">Manufacturer Specific</td><td rowspan="8">0: is not active
1: is active</td></tr>
<tr><td>1</td></tr>
<tr><td>2</td></tr>
<tr><td>3</td></tr>
<tr><td>4</td></tr>
<tr><td>5</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td></tr>
</table>

5.7. MECHATROLINK

MECHATROLINK Safety System

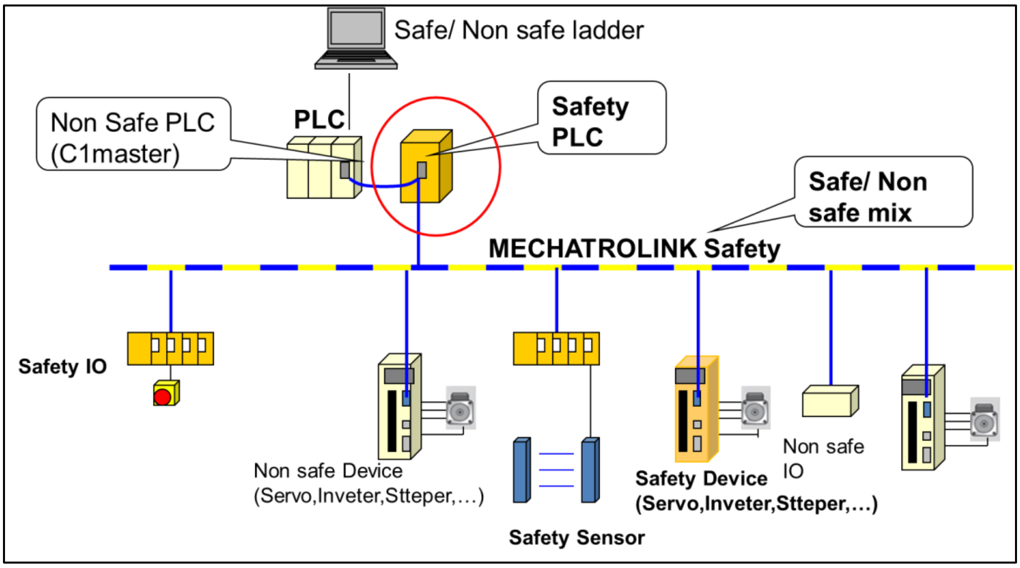


Figure 21: MECHATROLINK Safety System Structure

MECHATROLINK Safety System Structure

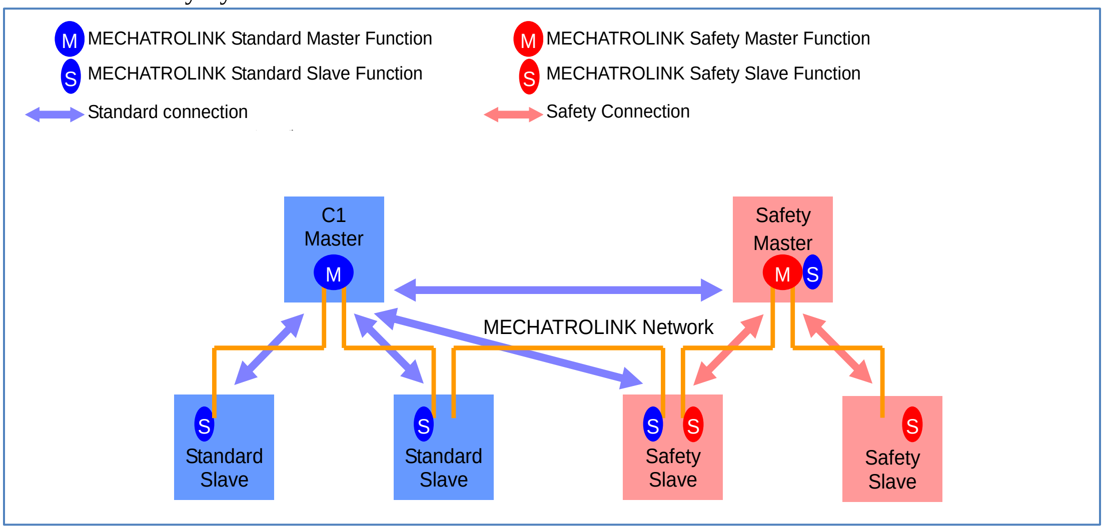


Figure 22: MECHATROLINK Safety System Structure Example

MECHATROLINK Safety PDU format

Figure 23: MECHATROLINK Safety System Safety PDU structure shows the safety PDU format. The Safe PDU is stored in an information section PDU part.

The PDU format is a structure that two or more blocks of 16 bytes or 32 bytes can be connected according to the communicated safety data size. The block of 50 pieces can be connected in the maximum.

Set the safety data size by "Request to set parameters" (S_PRM_SET) of the safety command.

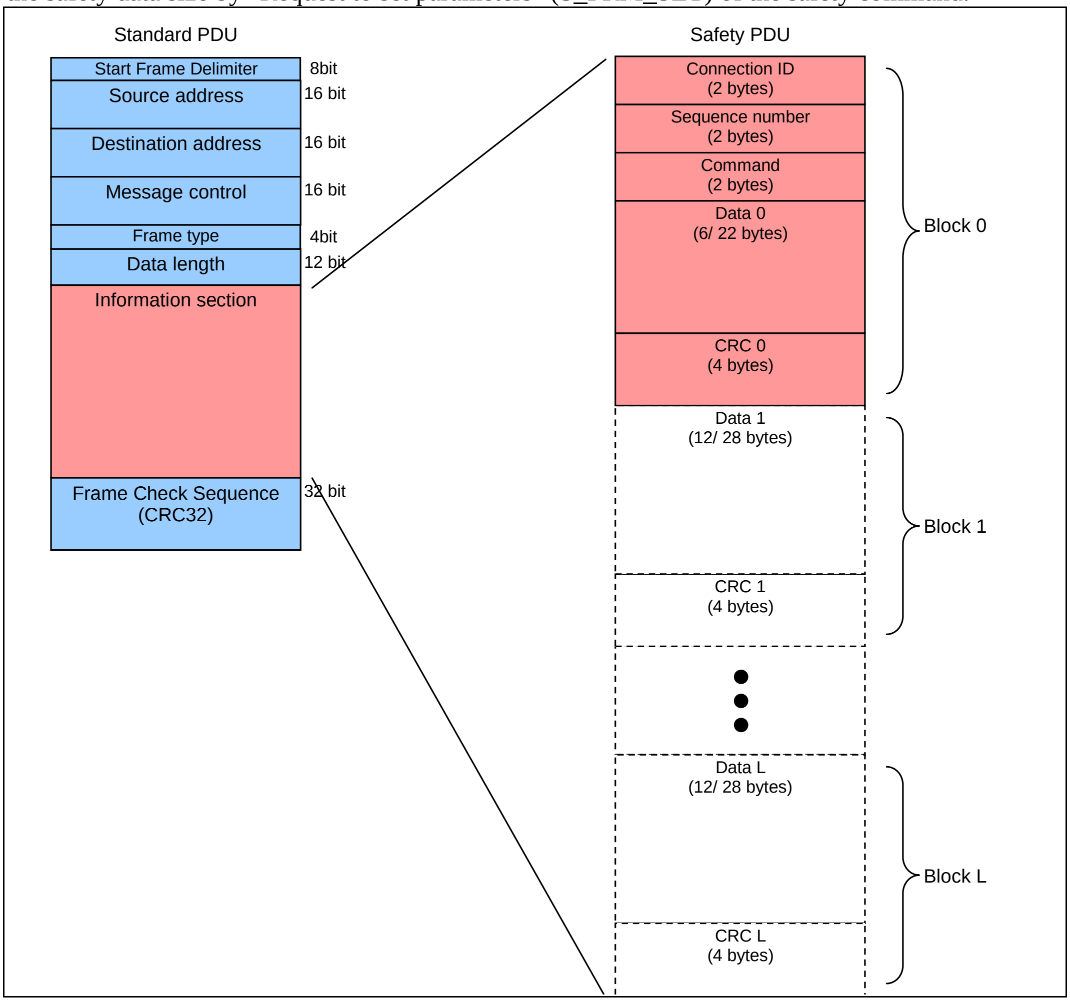
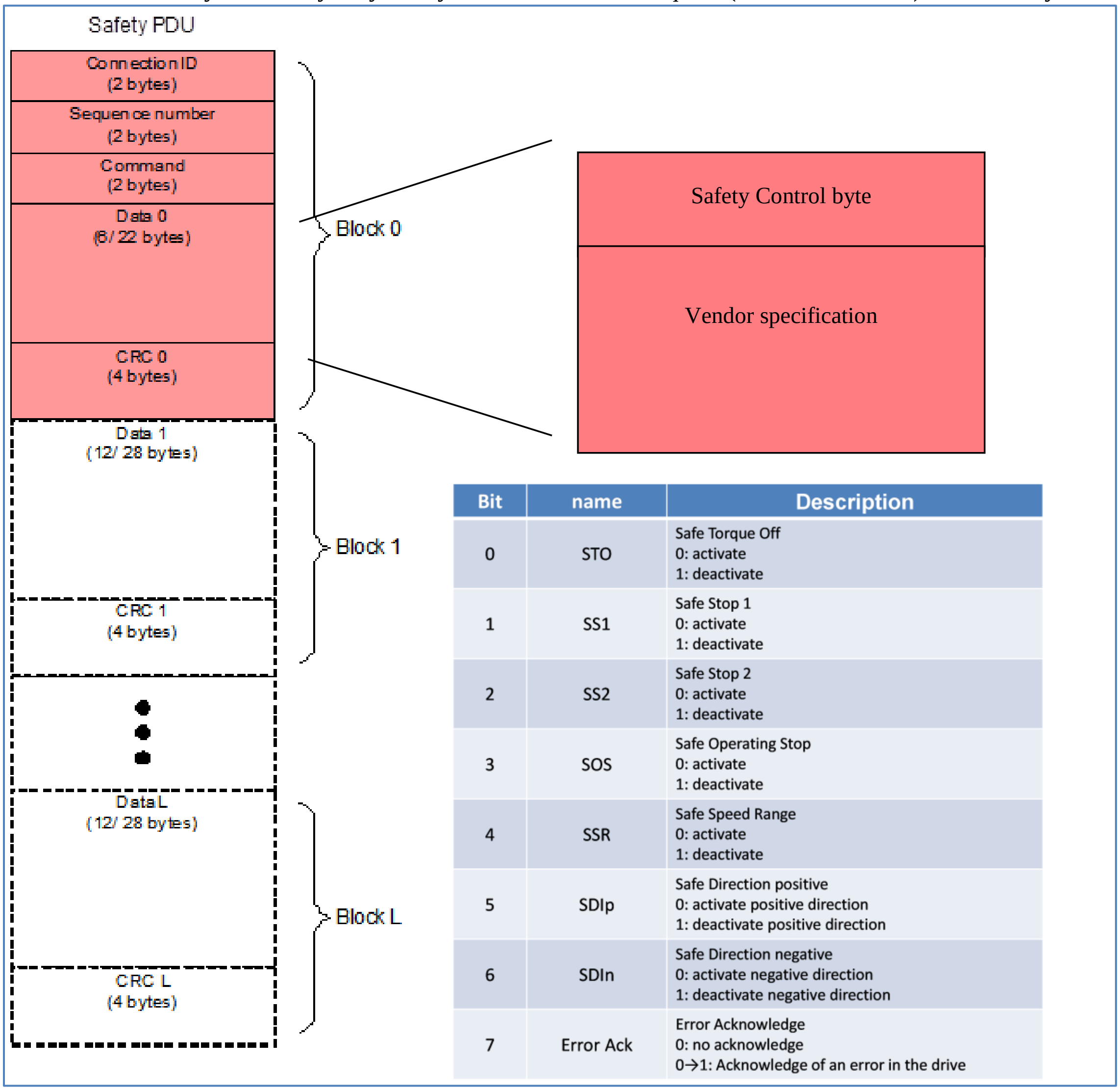


Figure 23: MECHATROLINK Safety System Safety PDU structure

MECHATROLINK Safety Control byte

Figure 24: MECHATROLINK Safety Control byte shows the structure of the MECHATROLINK Safety control byte. Send and receive the safety control byte by safety data transmission request (S_SAFE_DATA) of the safety command.

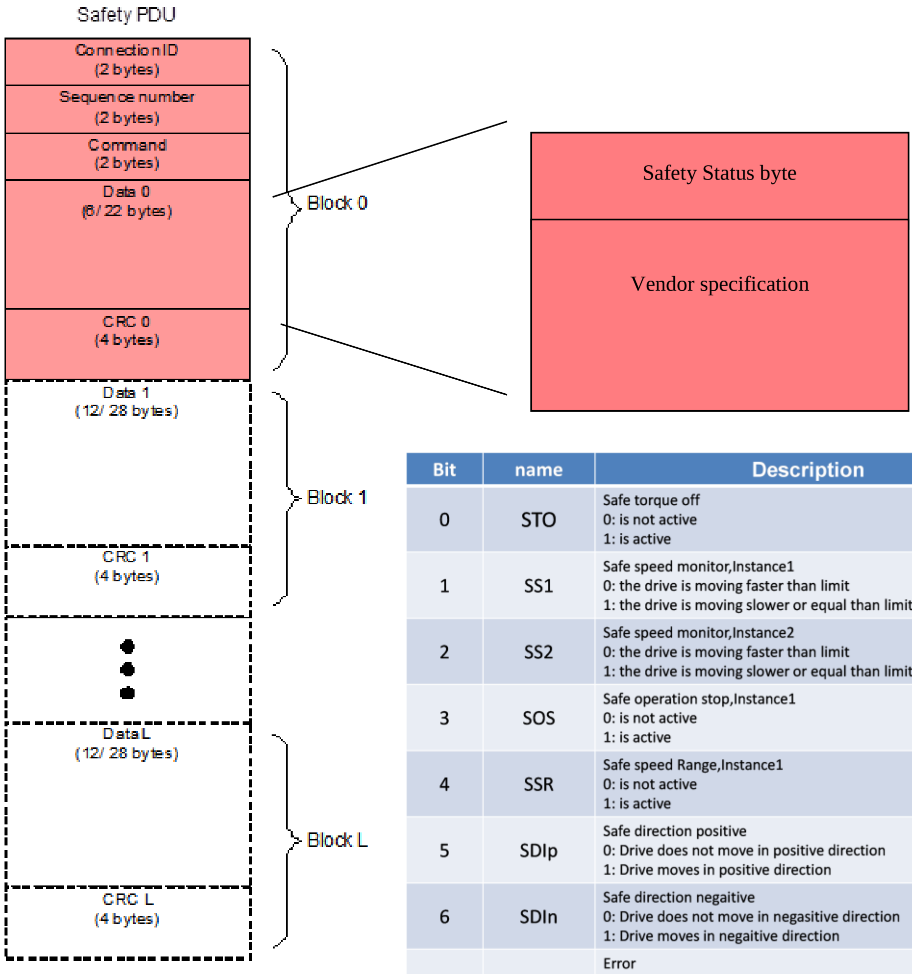


Bit	name	Description
0	STO	Safe Torque Off 0: activate 1: deactivate
1	SS1	Safe Stop 1 0: activate 1: deactivate
2	SS2	Safe Stop 2 0: activate 1: deactivate
3	SOS	Safe Operating Stop 0: activate 1: deactivate
4	SSR	Safe Speed Range 0: activate 1: deactivate
5	SDIp	Safe Direction positive 0: activate positive direction 1: deactivate positive direction
6	SDIn	Safe Direction negative 0: activate negative direction 1: deactivate negative direction
7	Error Ack	Error Acknowledge 0: no acknowledge 0→1: Acknowledge of an error in the drive

Figure 24: MECHATROLINK Safety Control byte

MECHATROLINK Safety Status byte

Figure 25: MECHATROLINK Safety Status byte shows the structure of the MECHATROLINK Safety status byte. Send and receive the safety status byte by safety data transmission request (S_SAFE_DATA) of the safety command.

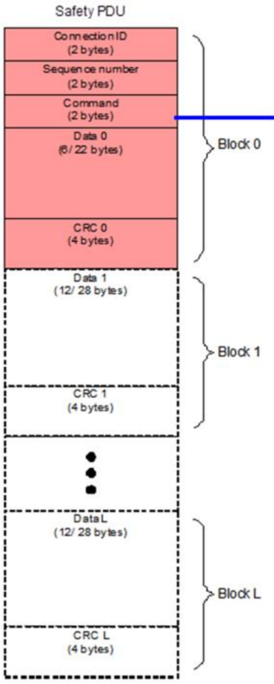


Bit	name	Description
0	STO	Safe torque off 0: is not active 1: is active
1	SS1	Safe speed monitor,Instance1 0: the drive is moving faster than limit 1: the drive is moving slower or equal than limit
2	SS2	Safe speed monitor,Instance2 0: the drive is moving faster than limit 1: the drive is moving slower or equal than limit
3	SOS	Safe operation stop,Instance1 0: is not active 1: is active
4	SSR	Safe speed Range,Instance1 0: is not active 1: is active
5	SDIp	Safe direction positive 0: Drive does not move in positive direction 1: Drive moves in positive direction
6	SDIn	Safe direction negaitive 0: Drive does not move in negasitive direction 1: Drive moves in negaitive direction
7	Error	Error 0: no error 1: at least one safety error das occurred

Figure 25: MECHATROLINK Safety Status byte

MECHATROLINK Safety Command

Figure 26: MECHATROLINK Safety Command shows the structure of the MECHATROLINK Safety command.



Code	Command	Description
0x0000	S_NOP	No operation.
0x0001	S_CONNECT_START	Request to start connection establishment. The safety master sends the master connection key to the safety slave. And the safety slave replies the slave connection key to the safety master. The safety master generates the connection ID by the master connection key and the slave connection key.
0x0002	S_CONNECT_CONF	Request to confirm connection establishment. If the connection ID the safety master generated is valid, the safety master sends this command to the safety slave and establishes connection.
0x0003	S_DISCONNECT	Request to disconnect connection. Disconnect the established connection.
0x0004	S_PRM_SET	Request to set parameters. The safety master set communication parameters and application parameters to the safety slave.
0x0005	S_PRM_APPLY	Request to apply parameters. Apply parameters sent by S_PRM_SET command to the safety slave.
0x0006	S_SAFE_DATA	Request to send safety data The safety master sends safety output data to the safety slave. And the safety slave that received it sends safety input data to the safety master.

Figure 26: MECHATROLINK Safety Command

6 Mapping SF_SafetyRequest to existing functionalities

The proposed mapping of SF_SafetyRequest to SafeMotion related functionalities is very much in line with the mapping of the other safety related functionalities like shown hereunder for SF_EmergencyStop, SF_EDM and SF_TwoHandControlTypeII and III. In Figure 27:Figure 27: Additional Safety Functionalities these functionalities are depicted in short form, while in Figure 28: these are represented with SF_SafetyRequest. Shown are several instances of SF_SafetyRequest for different drives (1 and 2) with a naming scheme for the drives and the safety functionalities.

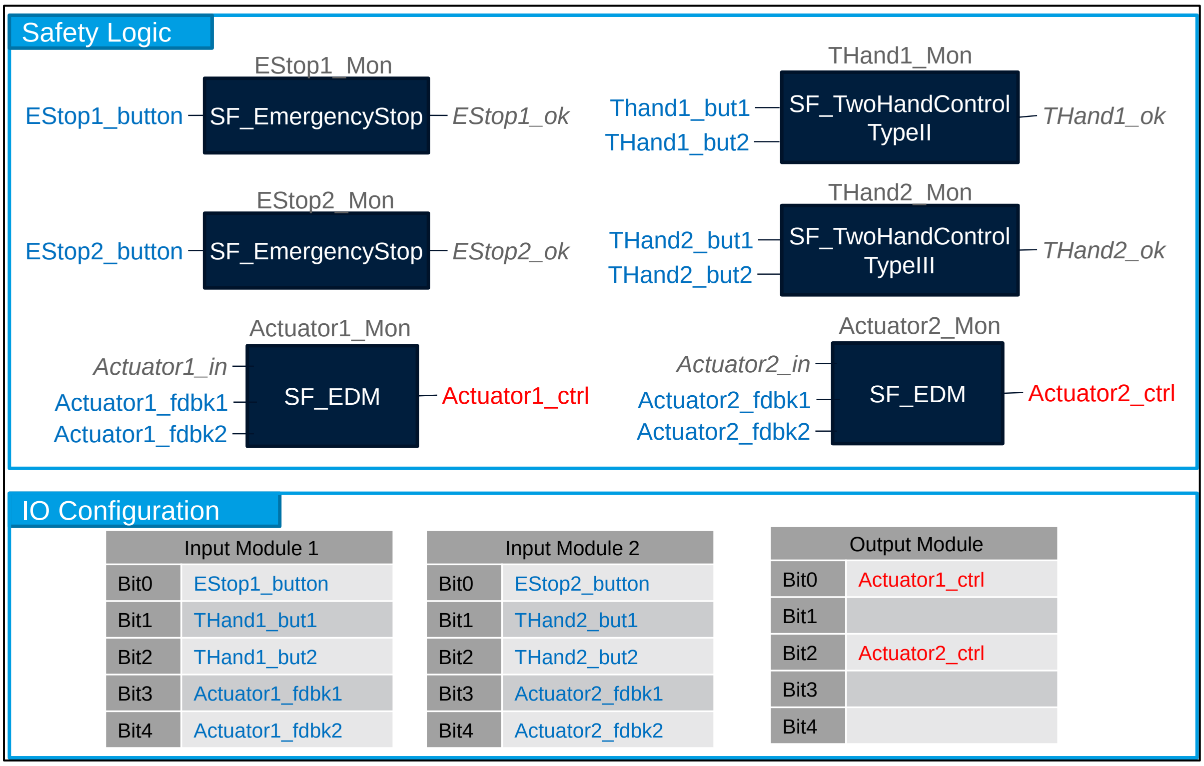


Figure 27: Additional Safety Functionalities

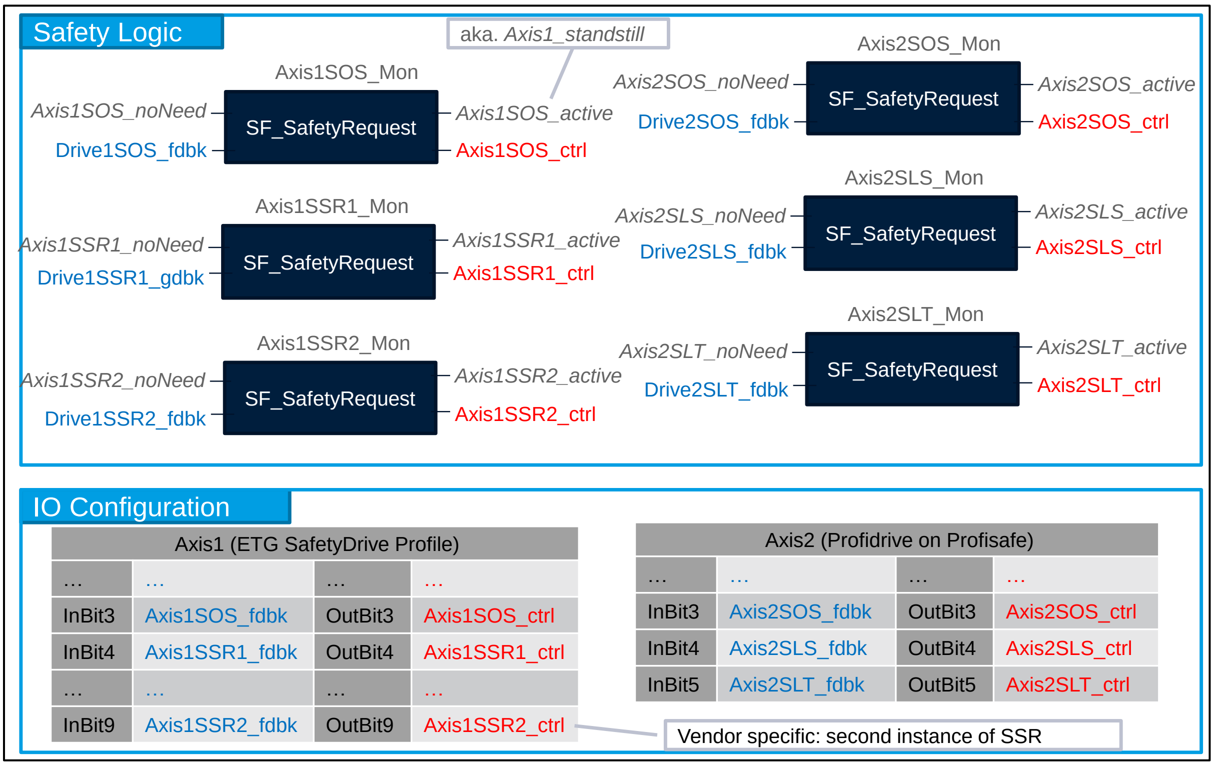


Figure 28: Mapping of these to SF_SafetyRequest