Safe electronic rotary cam arrangement

PILZ
THE SPIRIT OF SAFETY

Operating Manual-1003079-EN-02

1 Introduction

This operating manual describes how defined components from the Pilz range can be used to create a safe electronic rotary cam arrangement for the control of a mechanical press. The safe electronic rotary cam arrangement can be used in the automation system PSS 4000.

We assume that the following hardware and software products are used:

- Control system of the automation system PSS 4000 of performance class PSSu PLC (e.g. PSSu H PLC1 FS DP SN SD)
- PSSu E F ABS SSI
  (Electronic module for direct connection of an SSI absolute encoder for failsafe applications)
- Absolute encoder with SSI interface
  - PSEN enc m1 eCAM (multi-turn, hollow shaft version) or
  - PSEN enc m2 eCAM (multi-turn, solid shaft version) or
  - PSEN enc s1 eCAM (single-turn, hollow shaft version) or
  - PSEN enc s2 eCAM (single-turn, solid shaft version)
- Control block from the block library of the system software PAS4000

To implement the electronic cam arrangement, you must connect all the stated products with one another. This document is intended to be used as the operating instructions for connecting these products. These include information about configuring the products as well as programming guidelines for calling and linking control blocks.

The products listed (control system, modules, rotary encoder) are available from Pilz. The system software PAS4000 is available to download from the Pilz homepage.

You will need to be conversant with the general documentation on the automation system PSS 4000 (system description, safety manual, PSSuniversal installation manual, PAS4000 online help) in order to fully understand this manual. Please also refer to the following applicable documents:

- Operating manual for the relevant control system PSSu PLC
  (e.g. operating manual PSSu H PLC1 FS DP SN SD)
- Operating manual for the electronic module PSSu E F ABS SSI
- Operating manual for the absolute encoder:
  Operating manual PSEN enc m/s eCAM
- Descriptions of the relevant control blocks in PAS4000's online help

The operating manual is intended for experienced press specialists or persons with years of experience in the field of press upgrades.

This documentation is intended for instruction and should be retained for future reference.

1.1 Validity of documentation

This documentation is valid for the automation system PSS 4000. It is valid until new documentation is published.

This document often uses the term "PSS 4000" as an abbreviation of "Automation system PSS 4000".

1.2 Definition of symbols

Information that is particularly important is identified as follows:

DANGER!

This warning must be heeded! It warns of a hazardous situation that poses an immediate threat of serious injury and death and indicates preventive measures that can be taken.

WARNING!

This warning must be heeded! It warns of a hazardous situation that could lead to serious injury and death and indicates preventive measures that can be taken.

CAUTION!

This refers to a hazard that can lead to a less serious or minor injury plus material damage, and also provides information on preventive measures that can be taken.

NOTICE

This describes a situation in which the product or devices could be damaged and also provides information on preventive measures that can be taken. It also highlights areas within the text that are of particular importance.

INFORMATION

This gives advice on applications and provides information on special features.

2 Safety

2.1 Intended use

The safe electronic rotary cam arrangement is designed to replace electromechanical switches to control a mechanical press with an electronic solution. The requirements of DIN EN 692:2005+A1 clause 5.4.7 are equally met when the following prerequisites are also fulfilled:

- Use of all the documented products (see the chapter entitled Total Overview)
- Compliance with the documented configurations and settings
- Knowledge of and compliance with the general documentation on the automation system PSS 4000 and the applicable documents (see the chapter entitle Introduction)

NOTICE

- You must observe the safety regulations in this operating manual and in the applicable documents.
- All the laws and directives applicable for the country of sale must be followed. For Europe, these are the directive 2006/42/EC (Machinery Directive) for example, plus the relevant standards.

The safe electronic cam arrangement is designed for use in the automation system PSS 4000.

The safe electronic cam arrangement can be used in any situation where an angle resolution of 1/10 degrees is sufficient.

The documented blocks must be assigned to the FS resource of the relevant control system.

3 Total overview

The following block diagram provides an overview of the interaction between the hardware and software products you are using.

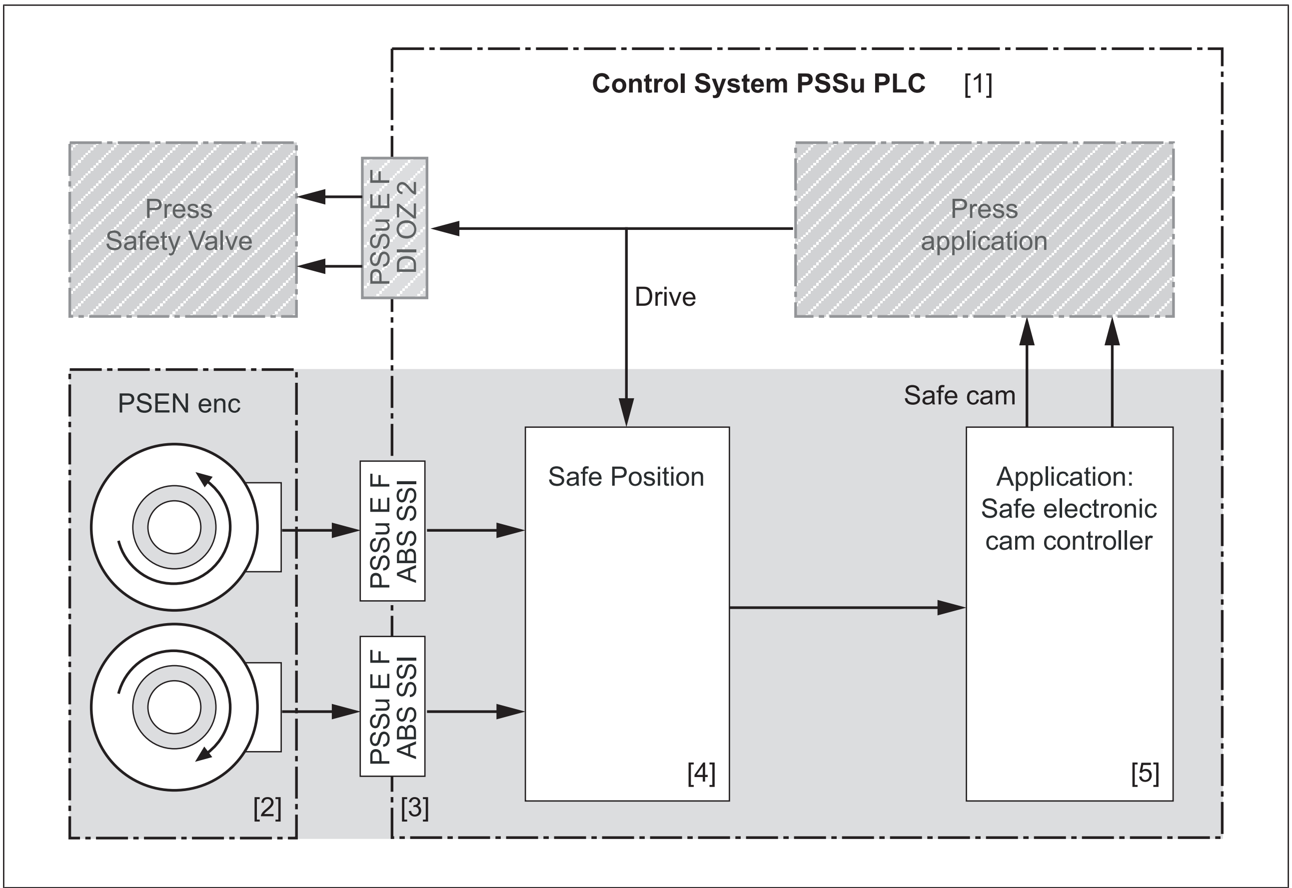


Fig.: Block circuit diagram containing all the components of the safe electronic rotary cam arrangement

[1] **Control system PSSu PLC**

This is a control system of the automation system PSS 4000 of performance class PSSu PLC with an FS resource (e.g. PSSu H PLC1 FS DP SN SD). On the one hand, the control system is used to actually control the press (press application); on the other hand, together with the absolute encoder [2] it forms the "Safe electronic rotary cam arrangement" (see [2] …[5]).

[2] **Absolute encoder**

The absolute encoder records the position of the shaft via two separate sensing units. The sensing units are set in opposite count directions. Each sensing unit is assigned an SSI interface on the absolute encoder to which it outputs its acquired position (see Absolute encoder [📖 10]).

[3] **Counter modules PSSu E F ABS SSI**

The SSI interfaces of the absolute encoder [2] are each connected to a counter module PSSu E F ABS SSI. This means that the two position values that come from the absolute encoder are read separately (see Counter Module [📖 13]). Both counter modules PSSu E F ABS SSI are part of the relevant control system.

[4] **Safe Position**

Determining the safe position value
The system checks the loaded position values and aggregates them to one safe position value (see Safe Position [📖 14]).

[5] **Application: Safe electronic cam controller**

The software part of the electronic rotary cam arrangement
The system determines from the safe position value a safe angle value, from which it determines the signal for safe shutdown of the press (see Electronic Camshaft (Application) [📖 25]).

4 Absolute encoder

Depending on the application, one of the following absolute encoders from Pilz can be used:

Design	Product type	Features
Multi-turn distance measuring system	PSEN enc m1 eCAM	Absolute encoder with ▸ hollow shaft ▸ SSI interface
	PSEN enc m2 eCAM	Absolute encoder with ▸ solid shaft ▸ SSI interface
Single-turn distance measuring system	PSEN enc s1 eCAM	Absolute encoder with ▸ hollow shaft ▸ SSI interface
	PSEN enc s2 eCAM	Absolute encoder with ▸ solid shaft ▸ SSI interface

Structure and function

The applicable absolute encoders are rotary measuring systems. The measuring systems are absolute multi-turn/single-turn distance measuring systems with Synchronous Serial Interface (SSI).

An absolute encoder consists of a redundant, dual-channel system, in which optical and magnetic sensing units are arranged on a drive shaft (hollow shaft/solid shaft). The two sensing units are used to detect the angle position. The optical sensing unit has greater accuracy and is used in this application as a master unit. The magnetic sensing unit is used for a feasibility test in this application.

Each sensing unit is assigned an SSI interface, to which the recorded angle position is issued.

4.1 Setting the count direction on the absolute encoder

The count direction can be set separately for each sensing unit (M1, M2). In the terminal compartment of the relevant absolute encoder there is a pair of DIP switches for each sensing unit.

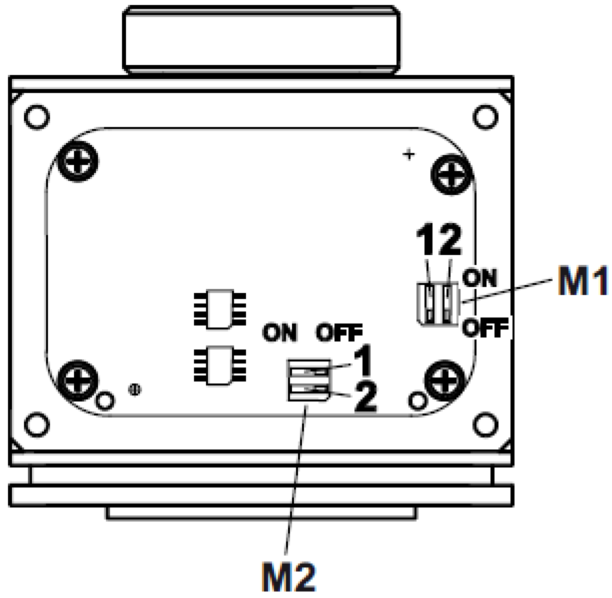
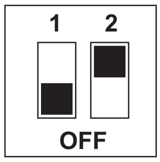
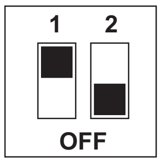


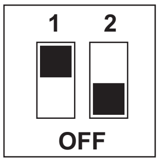
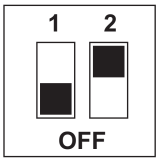
Fig.: Position of DIP switches

Set the sensing units to opposite count direction. The following combinations are possible:

- **DIP switch settings in combination 1**
  - Setting sensing unit M1: "Increasing in clockwise direction" and
  - Setting sensing unit M2: "Decreasing in clockwise direction"

Sensing unit	Count direction	DIP switch 1	DIP switch 2	View
M1	"Increasing in clockwise direction"	OFF	ON	
M2	"Decreasing in clockwise direction"	ON	OFF	

- **DIP switch settings in combination 2**
  - Setting sensing unit M1: "Decreasing in clockwise direction" and
  - Setting sensing unit M2: "Increasing in clockwise direction"

Sensing unit	Count direction	DIP switch 1	DIP switch 2	View
M1	"Decreasing in clock-wise direction"	ON	OFF	
M2	"Increasing in clock-wise direction"	OFF	ON	

NOTICE

– It is crucial to pay attention to the information and safety requirements in the operating manual PSEN enc m/s eCAM.

– Check the count direction when commissioning. In normal press operation the master unit needs to count forwards. Correct the switch settings of the DIP switch pair for M1 if necessary.

– When commissioning, check whether the sensing units are set in opposite count directions and correct the setting, if necessary.

5 Counter Module

Two counter modules PSSu E F ABS SSI are used. Each SSI interface on the employed absolute encoder is connected directly to the SSI interface on a PSSu E F ABS SSI. The counter modules PSSu E F ABS SSI are part of the employed control system.

Product type	Features
PSSu E F ABS SSI	Electronic module for direct connection of an SSI absolute encoder for failsafe applications

Mode of operation

As part of each cycle, a PSSu E F ABS SSI sends a pulse sequence at the test pulse output (CI) to the relevant SSI interface on the employed absolute encoder. Then the absolute encoder transmits its position data in this pulse to this SSi interface. The position data is read at the input of PSSu E F ABS SSI (D).
An instance of the control block FS_AbsoluteEncoder is required for each counter module. The control block FS_AbsoluteEncoder is a component of the PAS4000 library.

Configure properties

The properties of the two counter modules PSSu E F ABS SSI must be configured. This will be done in the hardware configuration of PAS4000. Configuration of the following properties is necessary:

Property	Value
Transmission rate	250 kHz
Gray code data format	FALSE
Fault detection in the data frame	FALSE
Input data length	32

NOTICE

It is crucial to pay attention to the information and safety requirements in the operating manual PSSu E F ABS SSI.

6 Safe Position

6.1 Overview

The following block diagram gives you an overview of the interaction between the software components that you are using.

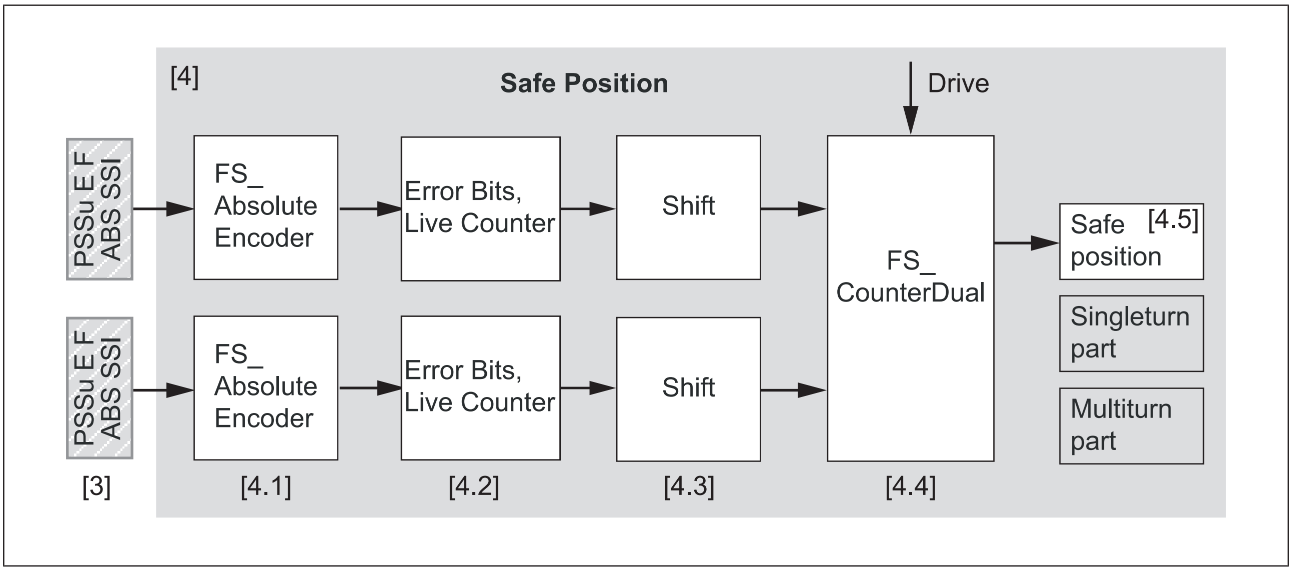


Fig.: “Safe position” block diagram

[3] **PSSu E F ABS SSI**

Loading of the position values of the absolute encoder using counter module PSSu E F ABS SSI (see Total overview [📖 8])

[4] **Safe Position**

Software components for determining the safe position(see Total overview [📖 8])

[4.1] **FS_AbsoluteEncoder**

Loading of the raw data of the counter modules using the FS_AbsoluteEncoder blocks

[4.2] **Error Bits, Live Counter**

Separation of the error bits and the live counter value form the absolute encoder’s raw data by masking;
subsequent checking of the raw data using the separated information

[4.3] **Shift**

Separation of the position values from the raw data using the FS_ShiftPosition-Range blocks

[4.4] **FS_CounterDual**

Plausibility checking of the position values using the FS_CounterDual block

[4.5] **Safe position**

Providing the safe position value for further processing in the user program
In the case of multi-turn applications, the safe position value can be extracted from the single-turn and multi-turn parts.

6.2 Loading Raw Data

Two instances of the FS_AbsoluteEncoder block are used to load the raw data.

Variable declaration

```
VAR
        ABS_Enc_Channel1                   : FS_AbsoluteEncoder;
        ABS_Enc_Channel2                   : FS_AbsoluteEncoder;
        Reset               AT %I*         : SAFEBOOL;
        ABS_Channel1                       : UDINT;
        ABS_Channel2                       : UDINT;
        Encoder_Channel1_Valid             : SAFEBOOL;
        Encoder_Channel2_Valid             : SAFEBOOL;
END_VAR
```

Program

The instantiated blocks are called in the user program. The I/O variables must be assigned to the I/O data of the counter modules.

```
CAL          ABS_Enc_Channel1(
AutoReset := FALSE,
MonitoredReset := TRUE,
Reset := Reset,
Reset_Valid := TRUE,
ValueValid => Encoder_Channel1_Valid,
Value => ABS_Channel1
)


CAL          ABS_Enc_Channel2(
AutoReset := FALSE,
MonitoredReset := TRUE,
Reset := Reset,
Reset_Valid := TRUE,
ValueValid => Encoder_Channel2_Valid,
Value => ABS_Channel2
)
```

Result

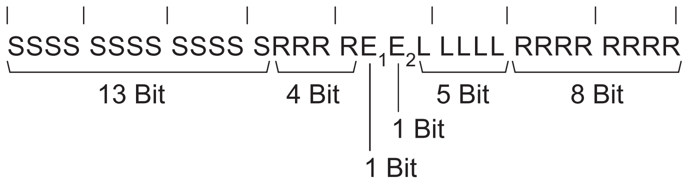
- ***ABS_Channel1*** and ***ABS_Channel2***
  The two O-variables of the raw data deliver the raw data of the two absolute encoders. Subsequently, the system checks the raw data and processes it.
- ***Encoder_Channel1_Valid*** and ***Encoder_Channel2_Valid***
  The two O-variables contain confirmation that the loaded raw data is valid.

6.3 Checking Raw Data

At each task cycle, a counter module PSSu E F ABS SSI loads a new position value from the sensing units of the two-channel absolute encoder. Raw data with a 32-bit data width is loaded via each channel. Data assignment is different with single-turn and multi-turn absolute encoders.

Raw data of a single-turn absolute encoder

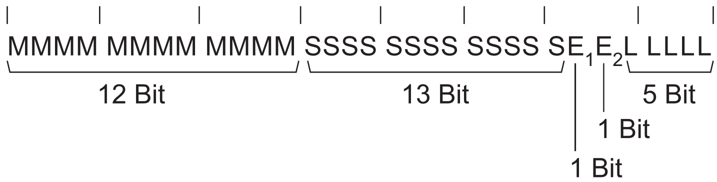
Structure and contents of the 32-bit datum:



S	Single-turn value
R	Reserved
E_1	Error bit of channel 1
E_2	Error bit of channel 2
L	Live counter

Raw data of a multi-turn absolute encoder

Structure and contents of the 32-bit datum:



M	Multi-turn value
S	Single-turn value
E_1	Error bit of channel 1
E_2	Error bit of channel 2
L	Live counter

Separate check information

The 32 bits of raw data contains 2 error bits and 5 bits for a live counter. The error bits and the live counter are separated by masking. In the go-state, the error bits are = 0 and the live counter changes in each task cycle. Additional variables are needed for masking.

Variables

```
VAR
	Encoder_Value_DWORD_1, LC_Channel1_akt_Cycle,
	LC_Channel1_last_Cycle                          : DWORD;
	Daten_Channel1_Valid                            : SAFEBOOL;

	Encoder_Value_DWORD_2, LC_Channel2_akt_Cycle,
	LC_Channel2_last_Cycle                          : DWORD;
	Daten_Channel2_Valid                            : SAFEBOOL;
END_VAR
```

Note:
If you want to separate the check information from the raw data, you need a different mask for the raw data of a single-turn absolute encoder to the one for the raw data of a multi-turn absolute encoder.

- **Masks for single-turn raw data**

```
VAR CONSTANT
	Mask_Error_Bit:                     DWORD :=
	DWORD#2#00000000000000000110000000000000;
	Mask_LiveCounter:                   DWORD :=
	DWORD#2#00000000000000000001111100000000;
	Mask_0000:                          DWORD :=
	DWORD#2#00000000000000000000000000000000;
END_VAR
```

- **Masks for multi-turn raw data**

```
VAR CONSTANT
	Mask_Error_Bit:                     DWORD :=
	DWORD#2#00000000000000000000000001100000;
	Mask_LiveCounter:                   DWORD :=
	DWORD#2#00000000000000000000000000011111;
	Mask_0000:                          DWORD :=
	DWORD#2#00000000000000000000000000000000;
END_VAR
```

Program

The user program must meet the following requirements:

- Conversion of the loaded raw data from data type UDINT to DWORD
- Error if the error bit is set
- Error if the live counter is unchanged since the last task cycle

```
UDINT_TO_DWORD(
        IN := ABS_Channel1
        )
        ST          Encoder_Value_DWORD_1
        //------ Errorbit Channel1 -----
        AND         Mask_Error_Bit
        NE          Mask_0000
        R           Daten_Channel1_Valid
        //------ Live counter -------------
        LD          LC_Channel1_akt_Cycle
        ST          LC_Channel1_last_Cycle
        LD          Encoder_Value_DWORD_1
        AND         Mask_LiveCounter
        ST          LC_Channel1_akt_Cycle
        EQ          LC_Channel1_last_Cycle
        R           Daten_Channel1_Valid
        //-----------------------------------
        LDN         Encoder_Channel1_Valid
        R           Daten_Channel1_Valid
        LD          Reset
        S           Daten_Channel1_Valid

UDINT_TO_DWORD(
        IN := ABS_Channel2
        )
        ST          Encoder_Value_DWORD_2
        //------ Errorbit Channel2 ----
        AND         Mask_Error_Bit
        NE          Mask_0000
        R           Daten_Channel2_Valid
        //------ Live counter --------------
        LD          LC_Channel2_akt_Cycle
        ST          LC_Channel2_last_Cycle
        LD          Encoder_Value_DWORD_2
        AND         Mask_LiveCounter
        ST          LC_Channel2_akt_Cycle
        EQ          LC_Channel2_last_Cycle
        R           Daten_Channel2_Valid
        //-----------------------------------
        LDN         Encoder_Channel2_Valid
        R           Daten_Channel2_Valid
        LD          Reset
        S           Daten_Channel2_Valid
```

Result

Daten_Channel1_Valid and ***Daten_Channel2_Valid*** confirms that

- the data was transferred correctly,
- there is no error message from the absolute encoders, and
- the message frames that were received last are current.

6.4 Separating Position Values

To be able to carry out plausibility checking of the position values of the two channels, each position value must be stored right-aligned in a variable. You do this using the FS_ShiftPositionRange block.

Masking in the same way as with the check information is not necessary.

Variables

Two instances of the FS_ShiftPositionRange block are needed as well as two variables in which the system stores the results of the shifting operations.

```
VAR
      SH_M              : FS_ShiftPositionRange;
      SH_S              : FS_ShiftPositionRange;
      Channel1_value    : UDINT;
      Channel2_value    : UDINT;
END_VAR
```

Note:
To get the right-justified position value, shifting must be by 19 bits with a single-turn absolute value encoder and by 7 bits with a multi-turn one.

```
VAR CONSTANT            // Constant for  (singleturn)
      ShiftBitNo:       USINT := USINT#19;
END_VAR
                        // oder

VAR CONSTANT            // Constant for  (multiturn)
      ShiftBitNo:       USINT := USINT#7;
END_VAR
```

Program

```
CAL         SH_M(
ValueIn := ABS_Channel1,
N := ShiftBitNo,
ValueOut => Channel1_value
)

CAL         SH_S(
ValueIn := ABS_Channel2,
N := ShiftBitNo,
ValueOut => Channel2_value
)
```

Result

Variables ***Channel1_value*** and ***Channel2_value*** contain the current and checked position value in each case.

6.5 Plausibility Checking of Position Values

By plausibility checking the two position values using the FS_CounterDual block, you get a safe position value.

Note:
Block FS_CounterDual supports a large number of applications. In the present application, however, the block is used exclusively for obtaining a safe position value by means of plausibility checking.

Variables

```
VAR
      Safe_Pos                 : FS CounterDual;
      Drive                    : SAFEBOOL;
      Safe_Position            : SAFEUDINT;
      SafePosition_Valid       : SAFEBOOL;
      Up, Down,
      ToleranceExceeded,Error_Standstill,
      Error_Shaft_broken,CD_ParameterError,
      CD_ReadyForReset,InputNotValid          : BOOL;
      DeltaValue,DeltaValueMax,
      Speed, LastStatStopTime                 : UDINT;
      Standstill               : SAFEBOOL;
END_VAR
VAR CONSTANT
      CounterRange      : USINT :=USINT#13;   //13 = Single-,
                                              //25 = Multiturn
      DriveTime         : TIME     := TIME#250ms;
END_VAR
```

Information About the Variables of the FS_CounterDual Block:

Variable	Comment
Safe_Pos	Instance of FS_CounterDual
Drive	A safe signal to monitor for possible shearpin breakage The signal indicates whether a movement is taking place.
Safe_Position	Safe position value (result of FS_CounterDual)
SafePosition_Valid	Confirmation of correctly plausibility checking
Up, Down, ToleranceExceeded, Error_Standstill, Error_Shaft_broken, CD_ParameterError, CD_ReadyForReset, InputNotValid DeltaValue, DeltaValueMax, Speed, LastStatStopTime Standstill	These variables are not evaluated in the present application. They are used exclusively to provide information in the dynamic program display or variable display

Variable	Comment
CounterRange	13 = single-turn 25 = multi-turn
DriveTime	Start delay On switching from ***Drive*** (see Total overview [8]), the time is started. After the ***DriveTime*** has expired, the absolute encoders must report a movement; otherwise, a broken shearpin is detected.

Program

The parameter setting that is documented below applies only to the present application. It is essential to apply all the constants as they are documented below:

```
CAL        Safe_Pos(
      Value1 := Channel1_value,
      Value1Valid := Daten_Channel1_Valid,
      Value2 := Channel2_value,
      Value2Valid := Daten_Channel2_Valid,
      CounterRange := CounterRange,
      SpeedMovingAverage := USINT#5,
      Reverse := TRUE,
      Tolerance := UINT#50,
      ToleranceTime := TIME#20ms,
      Drive := Drive,
      DriveValid := TRUE,
      ThresholdStandStill := UINT#20,
      ThresholdDrive := UINT#50,
      StandStillTime := TIME#500ms,
      DriveTime := DriveTime,
      ThresholdDirection := UINT#15,
      Reset := Reset,
      Enable => SafePosition_Valid,
      Value => Safe_Position,
      Speed => Speed,
      Standstill => Standstill,
      CountingUp => Up,
      CountingDown => Down,
      DeltaValue => DeltaValue,
      DeltaValueMax => DeltaValueMax,
      LastStartStopTime => LastStatStopTime,
      DiagToleranceExceeded => ToleranceExceeded,
      DiagStandstillError => Error_Standstill,
      DiagDriveError => Error_Shaft_broken,
      DiagParameterError => CD_ParameterError,
      DiagReadyForReset => CD_ReadyForReset,
      DiagInputNotValid => InputNotValid
      )
```

Result

Safe_Position contains the checked, plausibility-checked and current position value.

NOTICE

You must adjust the settings on block FS_CounterDual that refer to the properties of the press (e.g. ***ThresholdDrive***) to the actual conditions of the press. In this connection, please refer to the online help for block FS_CounterDual.

6.6 Separating the Safe Position Value Into The Single-Turn and Multi-Turn Parts

The description below is only relevant if you use the multi-turn version of an absolute encoder. It describes how you can separate the 25-bit wide position value into its multi-turn and single-turn parts.

Variables

```
VAR
      Single_DW, Multi_DW    : SAFEDWORD;
      Multi_DW_SH            : SAFEUDINT;
      Safe_Position_DW       : SAFEDWORD;
      Safe_Single_Position   : SAFEUDINT;
      Safe_Multi_Position    : SAFEUDINT;
      SH_Muilti_Turn         : FS_ShiftPositionRange;
END_VAR
VAR CONSTANT
 Singlepart: SAFEDWORD:=WORD#2#00000000000000000001111111111111;
 Multipart:  SAFEDWORD:=WORD#2#00000001111111111110000000000000;
END_VAR
```

Program

```
// ---------- seperate singleturn part --------------------
UDINT_TO_DWORD(
IN := Safe_Position
)
ST          Safe_Position_DW
AND         Singlepart   //00000000000000000001111111111111
ST          Single_DW
DWORD_TO_UDINT(
IN := Single_DW
)
ST          Safe_Single_Position

// ---------- seperate multiturn --------------------
LD          Safe_Position_DW
AND         Multipart    //00000001111111111110000000000000
ST          Multi_DW

DWORD_TO_UDINT(
IN := Multi_DW
)
ST          Multi_DW_SH

CAL         SH_Muilti_Turn( // shift 13 bit
ValueIn := Multi_DW_SH,
N := USINT#13,
ValueOut => Safe_Multi_Position
)
```

Result

Safe_Single_Position contains the single-turn value of the safe position value and ***Safe_Multi_Position*** contains its multi-turn value.

7 Electronic Camshaft (Application)

7.1 Overview

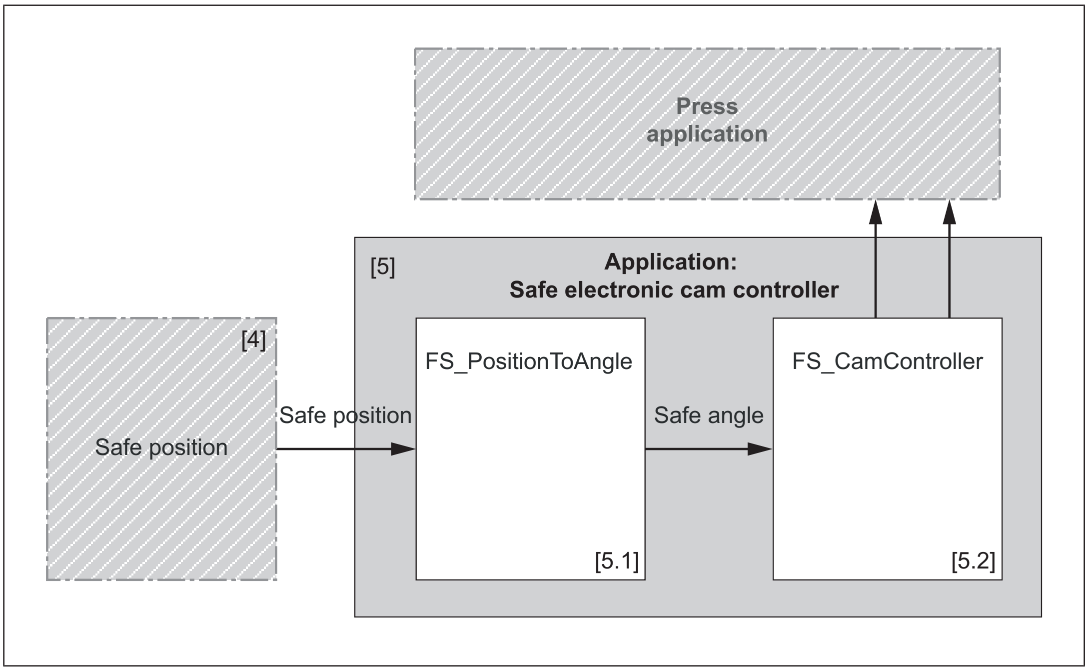


Fig.: Safe electronic rotary cam arrangement block diagram

[4] **Safe position** (see Total overview [8])

[5] **Application: Safe electronic cam controller** (see Total overview [8])

[5.1] **FS_PositionToAngle**

Determining the safe angle value from the safe position value using the FS_PositionToAngle block

[5.2] **FS_CamController**

Determining the signal for safe shutdown of the press from the safe angle values of block FS_CamController

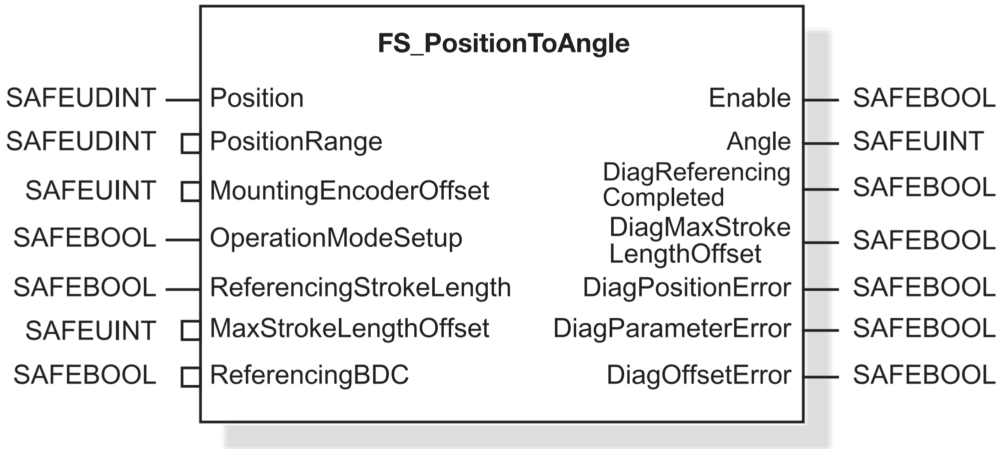
7.2 Block FS_PositionToAngle

7.2.1 Overview

This block uses the safe position value to calculate a safe angle value. To obtain a safe position value from a safe angle, the block scales the position value to an angle in the range of 0 … 359.9 degrees. The resolution is 1/10 degrees.

Block FS_PositionToAngle must be instantiated and the user must call it.

7.2.1.1 Schematic representation



Key:

Icon	Multi programming	IEC 61131 programming	Generic term in the description
☐	Parameter point (assignment in the block's properties view)	I-variable	I-variable
—	I-connection point, O-connection point	I-variable, O-variable	I-variable, O-variable
◖	I-PI point, O-PI point (mapping in the Multi Editor or I/O Mapping Editor)	I-PI variable, O-PI variable (mapping in the I/O Mapping Editor)	I-PI variable, O-PI variable

7.2.1.2 Notes on assigning the block interfaces

Input interface

I-variable	Meaning and assignment	Procedure
Position	safe position value; ***Position*** must be assigned with ***Safe_Position***	- - -
PositionRange	Value range for ***Position***; The value range must correspond to the resolution of the relevant absolute encoder in steps/revolution.	Always assign with the value 8192
MountingEncoderOffset	Encoder mounting offset in 0.1°	see Set encoder mounting offset [📖 30]
OperationModeSetup	Press in "setup" operating mode; ***OperatingModeSetup*** is used as the enable for a referencing process. This is permitted only in connection with a key switch.	Assign ***OperatingModeSetup*** to the signal "Setup operating mode"; e.g. use the appropriate signal from the operating mode selector switch see Set stroke length adjustment offset [📖 31]
ReferencingStrokeLength	Trigger referencing process due to stroke length adjustment; ***ReferencingStrokeLength*** controls the referencing process for determining the stroke length adjustment offset. The reset pushbutton or the two-hand pushbutton can be used as an operator element.	Assign ***Referencing-StrokeLength*** to the signal from the selected operator element see Set stroke length adjustment offset [📖 31]
MaxStrokeLengthOffset	Define maximum permitted stroke length adjustment offset in 0.1°; The maximum offset is defined by the manufacturer in compliance with the mechanical conditions of the press and the applicable standards	see Set stroke length adjustment offset [📖 31]
ReferencingBDC	Basis for referencing processes; When determining the encoder mounting offset (zero determination) or the stroke length adjustment offset (stroke length adjustment), the press must be in a defined mechanical position. In a referencing process, the press must either be in the position of 0° (TDC) or 180° (BDC).	Select one of the following settings: ▶ FALSE: All referencing processes are based on TDC (= default setting) ▶ TRUE: All referencing processes are based on BDC

Output interface

O-variable	Meaning	Value range
Enable	Enable See Angle and enable [📖 33]	▶ FALSE: No enable, there are errors present ▶ TRUE: Enable, the function is error-free
Angle	Safe position value (see I-variable ***Position***) that has been scaled to an angle in the range of 0 … 359.9 degrees See Angle and enable [📖 33]	UINT#0 ... 3599 corresponds to 0.0° … 359.9°
DiagReferencingCompleded	Status of the referencing process to determine the stroke length adjustment offset (diagnostic message); Following a referencing process to determine the stroke length adjustment offset, normal press operation may only be resumed if the signal at the O-variable ***DiagReferencingCompleted*** has changed from FALSE to TRUE. See Set stroke length adjustment offset [📖 31] and Diagnostics [📖 35]	▶ FALSE: No referencing process ▶ TRUE: Referencing process to determine the stroke length adjustment offset is complete
DiagMaxStrokeLengthOffset	Monitoring the maximum permitted stroke length adjustment offset See Diagnostics [📖 35]	▶ FALSE: Value not exceeded ▶ TRUE: Maximum permitted stroke length adjustment offset has been exceeded
DiagPositionError	Monitoring the position value See Diagnostics [📖 35]	▶ FALSE: No error ▶ TRUE: ***Position*** is outside the programmed range
DiagParameterError	Monitoring I-variables See Diagnostics [📖 35]	▶ FALSE: No monitoring error ▶ TRUE: I-variables are assigned invalid values or the referencing process to determine the encoder mounting offset is running
DiagOffsetError	Monitoring the stroke length adjustment offset See Diagnostics [📖 35]	▶ FALSE: Stroke length adjustment offset valid ▶ TRUE: Stroke length adjustment offset is invalid or not yet performed

7.2.2 Set encoder mounting offset

For mechanical reasons, once the absolute encoder has been installed there will always be a difference between the mechanical angle of the shaft and the measured angle. The difference is a fixed value.
Determine this difference value and assign the I-variable ***MountingEncoderOffset*** from the instantiated FS_PositionToAngle to this difference value. The difference value is called "Encoder mounting offset".

Determine encoder mounting offset

1. Set stroke length
   - ⇨ On a press with adjustable stoke length, set the eccentrics to maximum stroke length.
2. Set press in base position
   - ⇨ Set the press mechanically in the position you defined with the I-variable ***ReferencingBDC*** (TDC = 0° or BDC = 180°).
3. Pre-assign ***MountingEncoderOffset***
   - ⇨ Assign the value 9999 to ***MountingEncoderOffset*** and then run a project download.
4. Determine the encoder mounting offset
   - ⇨ Watch the O-variable ***Angle*** in PAS4000's dynamic program display.
5. Assigning ***MountingEncoderOffset***
   - ⇨ Assign ***MountingEncoderOffset*** with the value read out at ***Angle*** and then perform the project download again.
6. Carry out test
   - ⇨ Watch the O-variable ***Angle*** in PAS4000's dynamic program display and check whether the value that is output at ***Angle*** corresponds to the mechanical angle of the shaft.

NOTICE

Note that the encoder mounting offset has to be determined when first installed and each time the absolute encoder is exchanged. It is essential in this case to either comply with the procedure described above, or the procedure documented in the PAS4000 Library for FS_PositionToAngle.
Please also note that the system software PAS4000 and the current press project are required when exchanging the absolute encoder, as the offset has to be changed and saved in the project.

7.2.3 Set stroke length adjustment offset

If the stroke length is mechanically adjusted by turning the eccentric, the mechanical zero position of the shaft will also move. For this reason, each time the stroke length is adjusted, the electrical angle of the rotary cam arrangement must be tailored to the changed mechanical angel of the shaft; i.e. the displacement must be compensated via a referencing process. The displacement angle for the stroke length adjustment (stroke length adjustment offset) must be taught in during the referencing process and saved in non-volatile memory.

Referencing process to determine the stroke length adjustment offset after a stroke length adjustment

1. Set eccentrics
   ⇨ Use the eccentrics to set the required stroke length.
2. Set press in base position
   ⇨ Set the press mechanically in the position you defined with the I-variable ***ReferencingBDC*** (TDC = 0° or BDC = 180°).
3. Referencing step 1
   ⇨ Start referencing by pressing the pushbutton connected to ***ReferencingStrokeLength*** and keeping it pressed (see Fig. "Referencing process").
4. Referencing step 2
   ⇨ Set the operating mode selector switch to "Setup" mode within 3 s (see ***OperatingModeSetup*** in Fig. "Referencing process").
5. Referencing step 3
   ⇨ Release the pushbutton at ***ReferencingStrokeLength*** for 2 … 5 s (see Fig. "Referencing process").
6. Referencing step 4
   ⇨ Press the pushbutton at ***ReferencingStrokeLength*** for 2 … 5 s (see Fig. "Referencing process").
7. Complete referencing
   ⇨ Evaluate the O-variables ***DiagReferencingCompleted*** to determine whether the referencing process has been successfully completed. To do this, note the following descriptions.

Referencing process has been completed successfully

The determined stroke length adjustment offset is stored in non-volatile memory on the control system. Successful referencing is shown by the O-variable ***DiagReferencingCompleted***. On ***DiagReferencingCompleted*** the signal changes from FALSE to TRUE (see Fig. "Referencing process"). When you exit "Setup" operating mode (I-variable ***OperatingModeSetup***), ***DiagReferencingCompleted*** is reset.

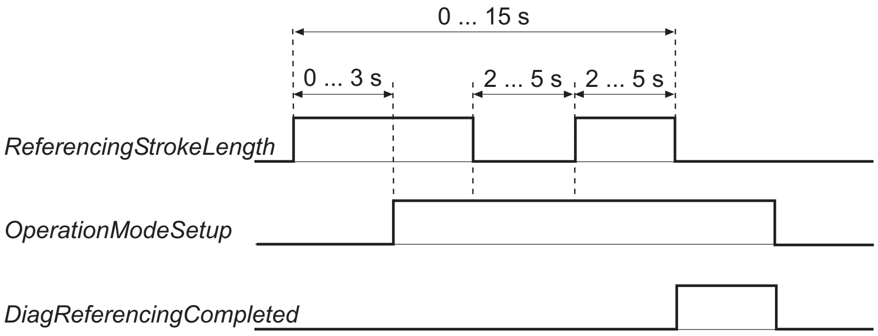


Fig.: Referencing process

Referencing process has not been completed successfully

The previous stroke length adjustment offset remains unchanged. The O-variable ***DiagReferencingCompleted*** remains FALSE, i.e. the signal does not change from FALSE to TRUE.

NOTICE

Following a referencing process to determine the stroke length adjustment offset, make sure that normal press operation does not resume until the signal at ***DiagReferencingCompeted*** changes from FALSE to TRUE.

7.2.4 Angle and enable

When functioning correctly, the block indicates the position value that is present at ***Position*** as an angle value at ***Angle***; i.e. the shaft angle is issued at ***Angle***.

<table>
<tr><th>O-variable</th><th colspan="2">Function</th></tr>
<tr><td>Angle</td><td colspan="2">The angle issued by the block FS_PostionToAngle is an addition of the following components:
Angle (Position)
Angle that is derived from the position supplied by the absolute encoder
MountingEncoderOffset
Offset between the zero point of the absolute encoders and the zero point of the shaft
StrokeLengthOffset (internal value)
Stroke length adjustment offset that is determined by the stroke length adjustment and taught in during the referencing process; The value is stored in non-volatile memory on the control system.
See the figure below "Offset by encoder mounting offset and stroke length adjustment offset"</td></tr>
<tr><td rowspan="3">Enable</td><td colspan="2">Enable</td></tr>
<tr><td>TRUE</td><td>The value that has been output at Angle is valid.
The press must be operated only when Enable = TRUE!</td></tr>
<tr><td>FALSE</td><td>The angle that has been output at Angle is not valid.
Resetting an enable must lead to the press safety valve being shut down.
The enable is not just reset if an error occurs, but also if a referencing process to determine the encoder mounting offset is carried out while "Setup" mode is being exited.</td></tr>
</table>

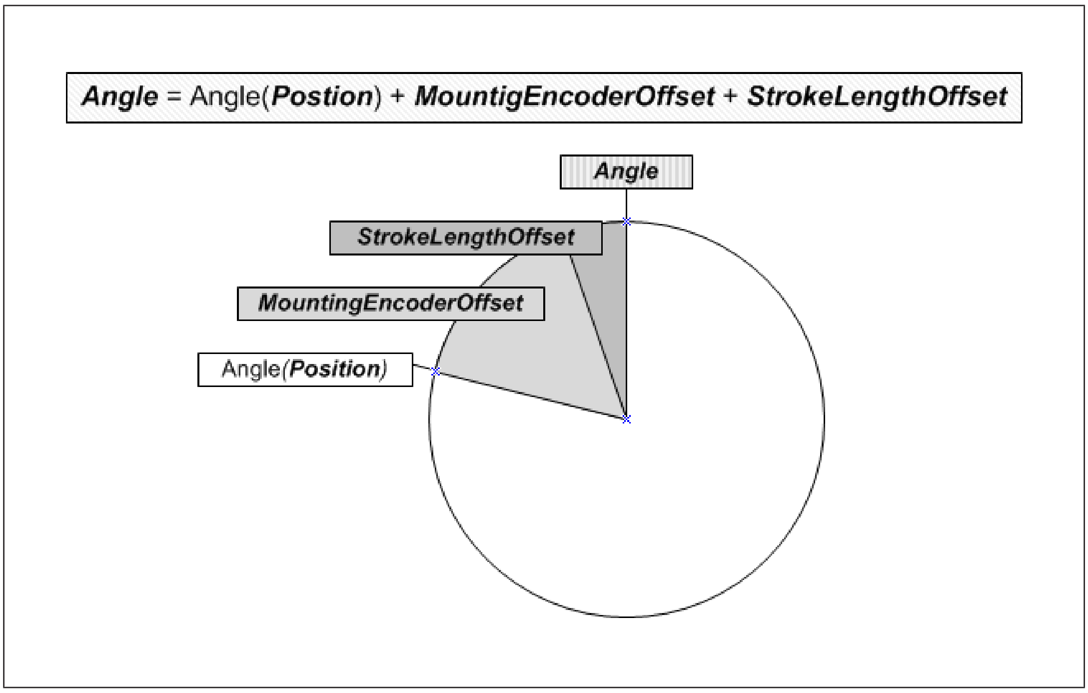


Fig.: Offset by encoder mounting offset and stroke length adjustment offset

7.2.5 Diagnostics

The block reacts to a detected error as follows:

- The O-variable ***Diag…*** is set.
- Enable is reset (O-variable ***Enable*** = FALSE)
- A message is displayed in the diagnostic list and diagnostic log via the process diagnostics

If an enable has been reset, the reason may be seen from the relevant O-variable.

<table>
<tr><th>O-variable</th><th colspan="2">Meaning</th></tr>
<tr><td rowspan="6">DiagReferencingCompleted</td><td colspan="2">Status of the referencing process to determine the stroke length adjustment offset (see Set stroke length adjustment offset [31])</td></tr>
<tr><td>FALSE</td><td>No referencing process</td></tr>
<tr><td>TRUE</td><td>Referencing process to determine the stroke length adjustment offset is complete</td></tr>
<tr><td colspan="2">The signal at the O-variable will change under the various conditions:</td></tr>
<tr><td>FALSE -> TRUE</td><td>Referencing process is completed successfully, the calculated stroke length adjustment offset has been adopted and is stored in non-volatile memory on the control system
Following a referencing process, make sure that normal press operation does not resume until the signal at DiagReferencingCompleted changes from FALSE to TRUE!</td></tr>
<tr><td>TRUE -> FALSE</td><td>"Setup" mode has been exited following a successful referencing process.</td></tr>
<tr><td rowspan="2">DiagMaxStrokeLengthOffset</td><td colspan="2">Monitoring the maximum permitted stroke length adjustment offset
A test is carried out to check whether the maximum permitted stroke length adjustment offset that has been set at the I-variable MaxStrokeLengthOffset) has been exceeded during the stroke length adjustment offset (see Set stroke length adjustment offset [31]).
Processes when the value has been exceeded:
- The referencing process is cancelled
- The stroke length offset remains unchanged and DiagMaxStrokeLengthOffset becomes TRUE
DiagMaxStrokeLengthOffset is reset as soon as a new referencing process is started.</td></tr>
<tr><td>TRUE</td><td>Maximum permitted stroke length adjustment offset has been exceeded</td></tr>
<tr><td rowspan="2">DiagPositionError</td><td colspan="2">Monitoring the position value
A test is carried out to check whether the value at Position is within the value range that has been defined via PositionRange.</td></tr>
<tr><td>TRUE</td><td>Position is outside the programmed range</td></tr>
</table>

O-variable	Meaning	
DiagParameterError	Monitoring I-variables ▸ ***PositionRange*** ▸ ***MountingEncoderOffset*** ▸ ***MaxStrokeLengthOffset*** ▸ ***ReferencingBDC***	
	TRUE	▸ The value at min. one of the I-variables ***PositionRange***, ***MountingEncoderOffset*** or ***MaxStrokeLengthOffset*** has been assigned an invalid value or ▸ The value at min. one of the I-variables ***MountingEncoderOffset*** or ***ReferencingBDC*** has changed during runtime or ▸ A referencing process must be carried out to determine the encoder mounting offset (see Set encoder mounting offset [30]).
DiagOffsetError	Monitoring the stroke length adjustment offset After a referencing process to determine the stroke length adjustment offset, the value is stored in non-volatile memory on the control system (see Set stroke length adjustment offset [31]). The stroke length adjustment offset becomes invalid if it is deleted on the control system, e.g. through a cold reset/cold restart, or if the project name is changed in PAS4000.	
	TRUE	Stroke length adjustment offset invalid

7.3 Block FS_CamController

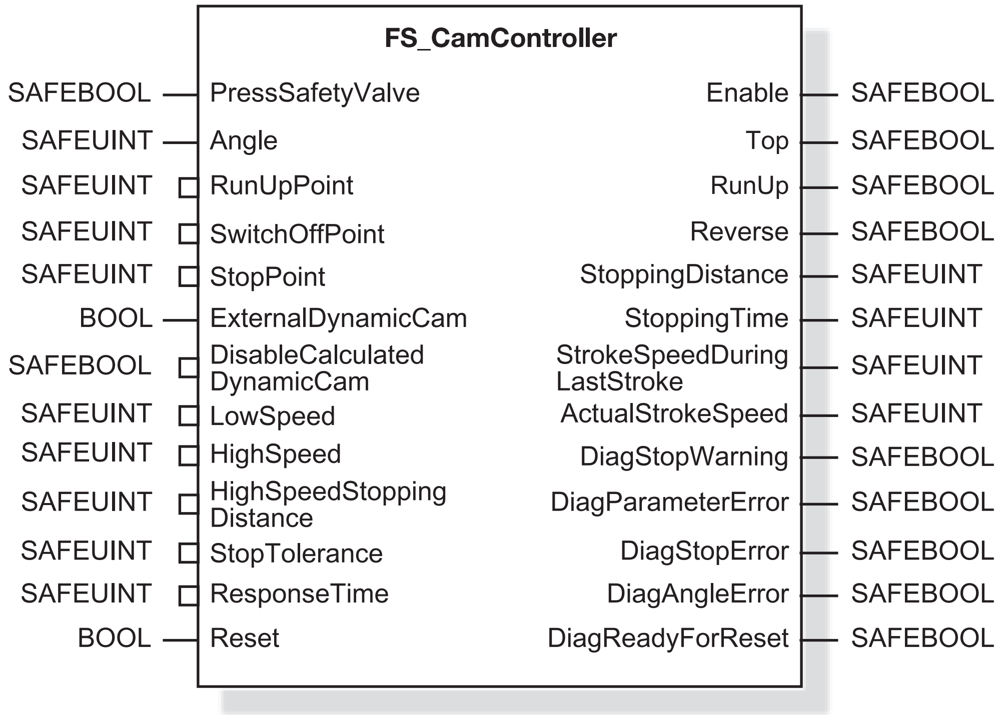
7.3.1 Overview

The block provides the position signals for a press control. These will be determined from the angle values of the block FS_PositionToAngle. The block implements the following:

- Signal for the stopping range (overrun cam)
- Signal for the run-up range (run-up cam)
- Signal for dynamic switch-off on presses with a variable number of strokes
- Overrun monitoring on presses with a fixed number of strokes
- Dynamic monitoring of stopping distance on presses with a variable number of strokes
- Detecting and displaying the direction of rotation
- Displaying number of strokes
- Displaying stopping distance
- Displaying stopping time

The block FS_CamController has to be instantiated and called up by the user.

7.3.1.1 Schematic representation



Key:

Icon	Multi programming	IEC 61131 programming	Generic term in the description
□	Parameter point (assignment in the block's properties view)	I-variable	I-variable
—	I-connection point, O-connection point	I-variable, O-variable	I-variable, O-variable
◖	I-PI point, O-PI point (mapping in the Multi Editor or I/O Mapping Editor)	I-PI variable, O-PI variable (mapping in the I/O Mapping Editor)	I-PI variable, O-PI variable

7.3.1.2 Terminology

Switch-off point
Angular position at which the press is normally switched off towards the end of a press stroke in "single-stroke" mode.
The press must come to a standstill at the stopping point (see "Stopping point"). The switch-off point can be advanced via a "Dynamic switch-off".

Stopping point
Angular position at which the press should come to a standstill after being shutdown by reaching the switch-off point.
The stop point is generally top dead centre (TDC = 0°). This may also be a different angle, depending on the construction of the press.

Stopping range
Angular range, which starts at the switch-off point and ends at a point determined by the application:

Fixed stroke speed
The end of the stopping range is defined by the user. With a fixed stroke speed, the end of the stopping range is also the end of overrun monitoring

Variable stroke speed (dynamic switch-off)
The end of the stopping range is fixed at 10° behind the stopping point

Run-up point
Angular position at which the hazardous press movement is at an end and from which the control command can therefore be accepted.

Run-up range
Angular range that starts with the run-up point and ends with the start of the stopping range.
The acceptance of the control command is permitted within the run-up range

Stopping distance
Distance (angular range) that the press travels between switch-off and standstill.
The stopping distance starts with the angular position in which the switch-off command is triggered and ends with the angular position in which the press comes to a standstill.

Overrun
Distance (angular range) that the press travels beyond a defined stopping point (e.g. top dead centre TDC) after switch-off.
The overrun distance starts with the defined stopping point and ends with the angular position in which the press comes to a standstill.

Safety time
Time that elapses between the point at which the safety device is triggered and the end of the hazardous movement of the press.
The safety time is used to calculate the minimum safety distance to the tool.

7.3.1.3 Notes on assigning the block interfaces

Input interface

I-variable	Meaning		Assignment
PressSafetyValve	Driving the press safety valve; The signal to activate the press safety valve comes from the press controller.		Link the ***PressSafetyValve*** directly to the O-variable that is used to activate the safety valve. ▶ FALSE: Press safety valve is switched off ▶ TRUE: Press safety valve is switched on
Angle	Current angle value of the shaft in 0.1°		Use the value of the O-variable ***Angle*** from the block FS_PositionToAngle
RunUpPoint	Bottom dead centre point in 0.1°		see Define run-up point (RunUpPoint) [📖 44]
SwitchOffPoint	Switch-off point in 0.1°; ▶ The meaning of the I-variable differs on presses with fixed/variable stroke speed ▶ Selection is made via ***DisableCalculateDynamicCam***		
	Fixed stroke speed		see Define switch-off point (SwitchOffPoint) for presses with fixed stroke speed [📖 47]
	Variable stroke speed		see Define switch-off point (SwitchOffPoint) for presses with variable stroke speed [📖 51]
StopPoint	Stopping point in 0.1°		see Define stop point (StopPoint) [📖 45]
DisableCalculateDynamicCam	Dynamic switch-off through speed and overrun measurement; This I-variable informs the electronic cam arrangement whether the press is operated with a fixed or a variable stroke speed.		▶ FALSE: Dynamic switch-off active ▶ TRUE: Dynamic switch-off inactive
LowSpeed	Lowest stroke speed for dynamic switch-off in strokes/min; (***DisableCalculateDynamicCam*** = FALSE)		see Define lowest stroke speed (LowSpeed) [📖 50]
HighSpeed	Highest stroke speed for dynamic switch-off in strokes/min; (***DisableCalculateDynamicCam*** = FALSE)		see Define highest stroke speed (HighSpeed) [📖 50]

I-variable	Meaning	Assignment
HighSpeedStoppingDistance	Stopping distance for dynamic switch-off at maximum stroke speed 0.1°; (***DisableCalculateDynamicCam*** = FALSE)	see Determine stopping distance (HighSpeedStoppingDistance) [53]
StopTolerance	Max. tolerance when stopping 0.1° ▸ Meaning differs on presses with fixed/variable stroke speed ▸ Selection is made via ***DisableCalculateDynamicCam***	
	Fixed stroke speed	see Define maximum overrun distance (StopTolerance) [49]
	Variable stroke speed	see Define tolerance for monitoring of the stopping distance (StopTolerance) [54]
ResponseTime	Response time of the brake when switching off the press in ms	see Determine response time of the brake (ResponseTime) [54]
Reset	Reset errors via reset button	see Diagnostics [58]

Output interface

O-variable	Meaning	Value range
Enable	Enable; Resetting an enable must lead to the press safety valve being shut down. **The press may only be operated when Enable = TRUE!**	▸ FALSE: No enable, there are errors present ▸ TRUE: Enable, the function is error-free
Top	Current angle within the stopping range; The press must be stopped in single stroke operating mode by the downstream blocks with the rising edge from ***Top***. The signal is also formed with a detected reverse movement.	▸ FALSE: Current angle outside the stopping range ▸ TRUE: Current angle inside the stopping range
RunUp	Current angle inside the run-up range; ▸ Forward mode: In the run-up range, the control command can be accepted. ▸ Reverse mode: The signal is not formed.	▸ FALSE: Current angle outside the run-up range ▸ TRUE: Current angle inside the run-up range (in forward mode only)

O-variable	Meaning	Value range
Reverse	Press in (physical) reverse mode; i.e. the shaft turns anti-clockwise.	▸ FALSE: Press in forward mode or at standstill ▸ TRUE: Press in reverse mode
StoppingDistance	Stopping distance from distance measurement in 0.1°; ***StoppingDistance*** shows the stopping distance that was determined the last time switch-off was via the switch-off point (***SwitchOffPoint***). The value is pending until the start of a new measurement. You can use ***StoppingDistance*** as a display or for individual monitoring of a stopping distance.	UINT
StoppingTime	Stopping distance from time measurement in ms; ***StoppingTime*** shows the stopping time that was determined the last time switch-off was via the switch-off point (***SwitchOffPoint***). The value is pending until the start of a new measurement. We recommend that you only use ***StoppingTime*** for display purposes. You **must** ensure that under no circumstances is ***StoppingTime*** used to determine or monitor the minimum safety distances of the press. Only appropriate overrun measuring devices should be used to determine or regularly monitor the stopping time.	UINT
StrokeSpeedDuringLastStroke	Stroke speed of last stroke in stroke/min; Prerequisite: Forward mode	UINT
ActualStrokeSpeed	Current stroke speed in strokes/min; Prerequisite: Forward mode	UINT
DiagStopWarning	Diagnostic message for switching the stroke speed; see Diagnostics [📖 58]	▸ FALSE: No warning ▸ TRUE: The max. stopping time can be exceeded if the machine is switched to a higher stroke speed in future.

O-variable	Meaning	Value range
DiagParameterError	Monitoring I-variables; see Diagnostics [📖 58]	▸ FALSE: No error ▸ TRUE: I-variables are assigned invalid values
DiagStopError	Overrun monitoring or dynamic monitoring of stopping distance; see Diagnostics [📖 58]	▸ FALSE: No error ▸ TRUE: The max. overrun distance (fixed stroke speed) or the max. stopping distance (variable stroke speed) was exceeded when switching off
DiagAngleError	Monitoring of angle value; see Diagnostics [📖 58]	▸ FALSE: No error ▸ TRUE: Angle is outside the permitted range or stroke speed is too high
DiagReadyForReset	Diagnostic information on reset; see Diagnostics [📖 58]	▸ FALSE: No reset required/ possible ▸ TRUE: Rotary cam arrangement is ready for reset All errors have been rectified, block restarts when the reset button is released

7.3.2 Run-up point and stopping point

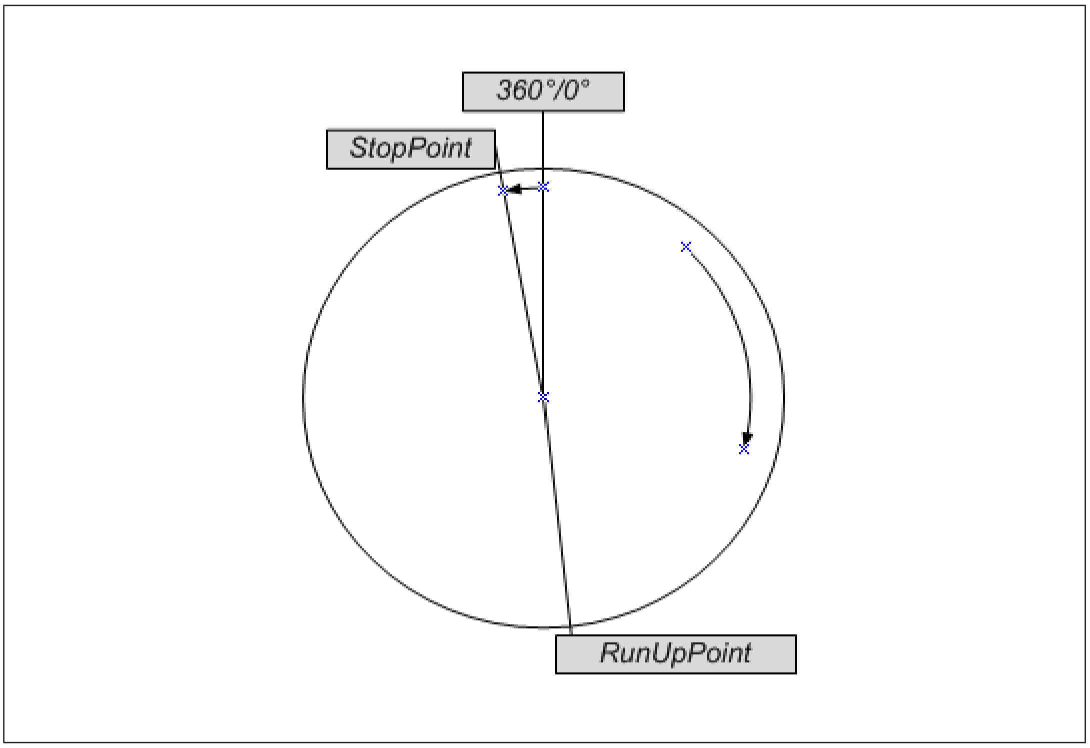


Fig.: Run-up point (***RunUpPoint***) and stop point (***StopPoint***)

7.3.3 Define run-up point (RunUpPoint)

RunUpPoint determines the **run-up point** of the press.

The run-up point is the angular position at which the hazardous press movement is at an end and from which the control command can therefore be accepted. The run-up point is also the start of the run-up cam.

Rule for assigning the *RunUpPoint*

The run-up point with presses is normally just before or just after 180°. Hazards can occur if the angle selected for the run-up is too small (see Run-up point and stopping point [44]).

7.3.4 Define stop point (StopPoint)

StopPoint determines the **stop point** of the press.

The stop point is the angular position at which the press should come to a standstill after being shutdown via the switch-off point (***SwitchOffPoint***).
The stop point is generally top dead centre (TDC = 0°). This may also be a different angle, depending on the construction of the press. For certain press types, the stop point can be advanced to another angle using ***StopPoint*** (= displacement of the angle anti-clockwise).

Rule for assigning the *StopPoint*

Angles greater than ***SwitchOffPoint*** up to and including 0° are possible for the stop point. It is not possible to place the stop point behind top dead centre; i.e. angles after 0° in clockwise direction are not possible (see Run-up point and stopping point [44]).

7.3.5 Press with fixed stroke speed

When using the block for presses with a fixed stroke speed, the following I-variables are relevant in addition to the I-variable ***RunUpPoint*** (run-up point) and ***StopPoint*** (stop point):

- ***DisableCalculateDynamicCam***
  TRUE: Press with fixed stroke speed
- ***SwitchOffPoint***
- ***StopTolerance***

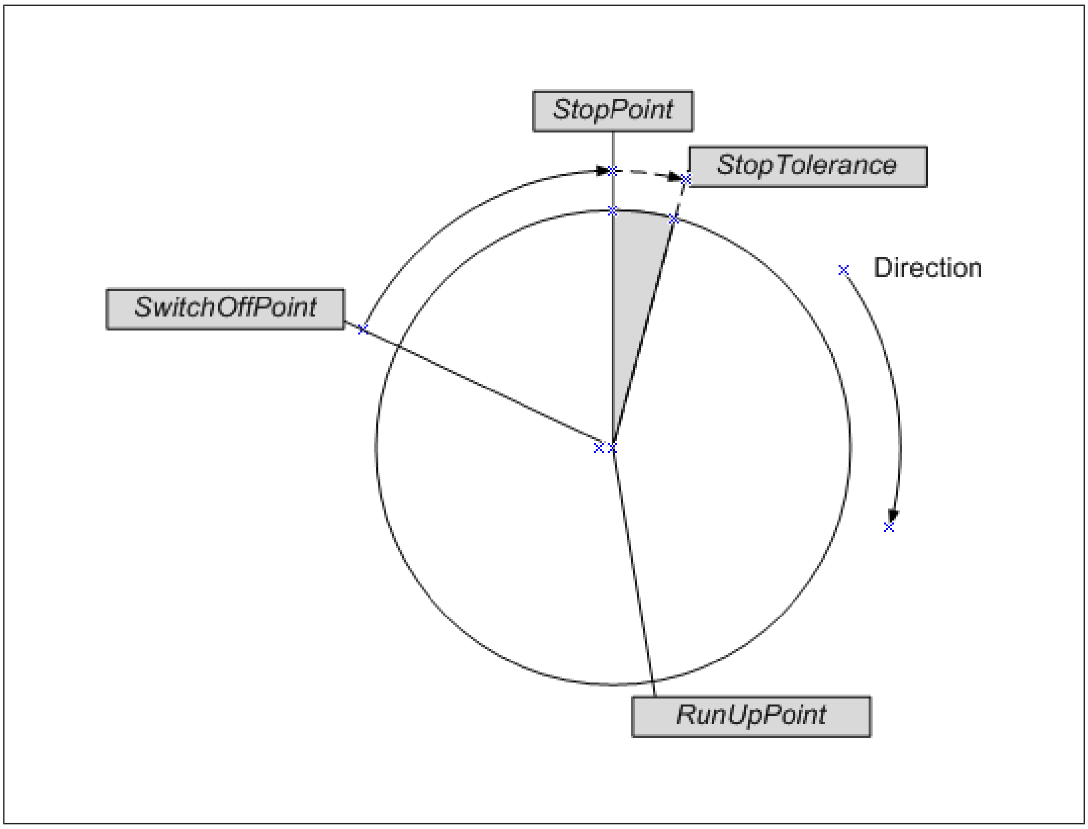


Fig.: Relevant I-variables on presses with a fixed stroke speed

Further information on the relevant I-variables can be found in the following chapters.

7.3.5.1 Define switch-off point (SwitchOffPoint) for presses with fixed stroke speed

SwitchOffPoint determines the **switch-off point** of the press.

The switch-off point is the angle position at which the press safety valve must be switched off in order to come to a standstill precisely at the stop point, taking the stopping distance into account (see Define stop point (StopPoint) [45]).

Prerequisites

SwitchOffPoint may only be set under the following conditions:

- The press is being commissioned for the first time
- A new brake in good working order is being used for initial commissioning.
- The most unfavourable mechanical conditions for a short braking distance are present (e.g. maximum tool weight, maximum pressure in the brake system and coupling system).

Rule for assigning the *SwitchOffPoint*

SwitchOffPoint must be less than the ***StopPoint***.
The exact value of ***SwitchOffPoint*** must be determined using test strokes with the press. The procedure described below deviates from the procedure that has been documented in the PAS4000 Library on FS_CamController. The two documented procedures are to be regarded as equal! To determine the ***SwitchOffPoint*** one of the documented procedures must be followed.

Procedure to determine the *SwitchOffPoint*

1. Set the press for fixed stroke speed
   ➪ Assign ***DisableCalculateDynamicCam*** with TRUE.
2. Assign ***SwitchOffPoint*** with test value
   ➪ Assign ***SwitchOffPoint*** with an angle value that is feasible for the test's switch-off point and then perform a project download.
3. Determine switch-off point (***SwitchOffPoint***)
   ➪ Perform a test stroke and display the actual stopping distance at the O-variable ***StoppingDistance*** in the dynamic program display of PAS4000.
   SwitchOffPoint = ***StopPoint*** – ***StoppingDistance***
   for ***StopPoint*** = 0° the following applies:
   SwitchOffPoint = 3600 (360°) – ***StoppingDistance***
4. Set determined switch-off point
   ➪ Assign ***DisableCalculateDynamicCam*** with TRUE and ***SwitchOffPoint*** with the determined angle. Then perform the project download again.
5. Check that the press stops at the stopping point (***StopPoint***)
   ➪ Use repeated test strokes to check whether the press stops in the angle that has been set by ***StopPoint***.

NOTICE

When performing test strokes to determine the switch-off point, make sure you heed the rules defined in the standards, under which the press overrun measurement must be made.

Also heed the information and requirements on this subject as described for this block in the PAS4000 online help (see online help - Library elements in the PAS4000 Library).

7.3.5.2 Define maximum overrun distance (StopTolerance)

During initial commissioning of the press the switch-off point ***SwitchOffPoint*** of the press is determined with new brake and clutch. Ageing of the brake can increase the stopping distance of the press.
For this reason the maximum permitted overrun distance is defined at ***StopTolerance*** for presses with a fixed stoke speed. The overrun distance must be defined by the user (press manufacturer, press upgrader).

Rule for assigning the *StopTolerance*

With knowledge of the press design characteristics and the standards situation, the maximum permitted overrun distance at ***StopTolerance*** must be defined by the user (press manufacturer, press upgrader).

It is essential to note:
The minimum safety distance must be maintained within the range of the maximum permitted overrun measurement. Make sure that ***StopTolerance*** is assigned in such a way that it is only possible to access the danger zone on the press when the press is at standstill.

7.3.6 Press with variable stroke speed

When using the block for presses with a variable stroke speed, the following I-variables are relevant in addition to the I-variable ***RunUpPoint*** (run-up point) and ***StopPoint*** (stop point):

- ***DisableCalculateDynamicCam***
  FALSE: Press with variable stroke speed
- ***LowSpeed***
- ***HighSpeed***
- ***SwitchOffPoint***
- ***HighSpeedStoppingDistance***
- ***StopTolerance***
- ***ResponseTime***

Further information on the relevant I-variables can be found in the following chapters. The following also contains information on the relationship between stopping distance and tolerance of the stopping distance and on the use of the warning threshold.

7.3.6.1 Define lowest stroke speed (LowSpeed)

LowSpeed defines the lowest stroke speed in "single stroke" operating mode for presses with variable stroke speed.

Rule for assigning *LowSpeed*

The ***LowSpeed*** value is determined by the electrical and mechanical design of the press. The value must be defined by the user (press manufacturer, press upgrader). Assign the press-dependent value specified by the user to ***LowSpeed.***

7.3.6.2 Define highest stroke speed (HighSpeed)

HighSpeed defines the highest stroke speed in "single stroke" operating mode for presses with variable stroke speed.

Rule for assigning *HighSpeed*

The ***HighSpeed*** value is determined by the electrical and mechanical design of the press. The value must be defined by the user (press manufacturer, press upgrader). Assign the press-dependent value specified by the user to ***HighSpeed.***

7.3.6.3 Define switch-off point (SwitchOffPoint) for presses with variable stroke speed

Dynamic switch-off is provided for presses with variable stroke speed. ***SwitchOffPoint*** determines the **switch-off point** of the press.

The switch-off point is the angle position at which the press safety valve must be switched off at the lowest speed (***LowSpeed***) in order to come to a standstill (see Define stop point (StopPoint) [📖 45]) precisely at the stopping point (***StopPoint***), taking the stopping distance into account (see Determine stopping distance (HighSpeedStoppingDistance) [📖 53]).

Prerequisites

SwitchOffPoint may only be set under the following conditions:

- The press is being commissioned for the first time
- A new brake in good working order is being used for initial commissioning.
- The most unfavourable mechanical conditions for a short braking distance are present (e.g. maximum tool weight, maximum pressure in the brake system and coupling system).

Rule for assigning the *SwitchOffPoint*

SwitchOffPoint must be less than the ***StopPoint***.
The exact value of ***SwitchOffPoint*** must be determined using test strokes with the press. The procedure described below deviates from the procedure that has been documented in the PAS4000 Library on FS_CamController. The two documented procedures are to be regarded as equal! To determine the ***SwitchOffPoint*** one of the documented procedures must be followed.

Procedure to determine the *SwitchOffPoint*

1. Set the press for fixed stroke speed

   ⇨ Assign ***DisableCalculateDynamicCam*** with TRUE!
   Please note: This assignment is only required to determine the ***SwitchOffPoint***.

2. Assign ***SwitchOffPoint*** with test value

   ⇨ Assign ***SwitchOffPoint*** with an angle value that is feasible for the test's switch-off and then perform a project download.

3. Determine switch-off point

   ⇨ Perform a test stroke with the **lowest** stoke speed and display the actual stopping distance at the O-variable ***StoppingDistance*** in the dynamic program display of PAS4000.
   SwitchOffPoint = ***StopPoint*** – ***StoppingDistance***
   for ***StopPoint*** = 0° the following applies:
   SwitchOffPoint = 3600 (360°) – ***StoppingDistance***

4. Set determined switch-off point

   ⇨ Assign ***DisableCalculateDynamicCam*** with TRUE and ***SwitchOffPoint*** with the determined angle. Then perform the project download again.

5. Check that the press stops at the stopping point (***StopPoint***)

   ⇨ Use repeated test strokes at the lowest stroke speed to check whether the press stops in the angle that has been set by ***StopPoint***.

NOTICE

When performing test strokes to determine the switch-off point, make sure you heed the rules defined in the standards, under which the press overrun measurement must be made.

Also heed the information and requirements on this subject as described for this block in the PAS4000 online help (see online help - Library elements in the PAS4000 Library).

7.3.6.4 Determine stopping distance (HighSpeedStoppingDistance)

HighSpeedStoppingDistance is used to define the **stopping distance** at the highest stroke speed for presses with variable stroke speed.

Prerequisites

HighSpeedStoppingDistance may only be set under the following conditions:

- The press is being commissioned for the first time.
- A new brake in good working order is being used for initial commissioning.
- The most unfavourable mechanical conditions for a short braking distance are present (e.g. maximum tool weight, maximum pressure in the brake system and coupling system).

Rule for assigning the *HighSpeedStoppingDistance*

The exact value of ***HighSpeedStoppingDistance*** must be determined using test strokes with the press. To do this, follow the procedure described below.

Procedure to determine the *HighSpeedStoppingDistance*

1. Set the press for variable stroke speed
   ➪ Assign ***DisableCalculateDynamicCam*** with FALSE.
2. Assign ***HighSpeedStoppingDistance*** with test value
   ➪ Assign ***HighSpeedStoppingDistance*** with an angle value that is feasible for the test's stopping point and then perform a project download.
3. Determine stopping distance
   ➪ Perform a test stroke with the **highest** stroke speed and display the actual stopping distance at the O-variable ***StoppingDistance*** in the dynamic program display of PAS4000.
   HighSpeedStoppingDistance = ***StoppingDistance***
4. Set determined stopping distance
   ➪ Assign ***DisableCalculateDynamicCam*** with FALSE and ***HighSpeedStoppingDistance*** with the determined angle. Then perform the project download again.
5. Check that the press stops at the stopping point (***StopPoint***)
   ➪ Use repeated test strokes at the highest stroke speed to check whether the press stops in the angle that has been set by ***StopPoint***.

> **NOTICE**
>
> When performing test strokes to determine the switch-off point, make sure you heed the rules defined in the standards, under which the press overrun measurement must be made.
>
> Also heed the information and requirements on this subject as described for this block in the PAS4000 online help (see online help - Library elements in the PAS4000 Library).

7.3.6.5 Define tolerance for monitoring of the stopping distance (StopTolerance)

For presses with variable stroke speed, ***StopTolerance*** is used to define the maximum tolerable overrun distance beyond the stopping distance (***HighSpeedStoppingDistance***) when the press is operated with the highest stroke speed. ***StopTolerance*** is therefore also a measure for the maximum tolerable wear on the brake.

Rule for assigning the *StopTolerance*

With knowledge of the press design characteristics and the standards situation, the maximum permitted overrun distance at ***StopTolerance*** that exceeds the stopping distance (***HighSpeedStoppingDistance***) must be defined by the user (press manufacturer, press upgrader).

Please note:

StopTolerance is defined for the highest stroke speed. This definition is made as an angle, however, in relation with the stroke speed a time specification can be derived from it. This time specification does not just apply to the highest stroke speed but also to lower stroke speeds. Therefore, the block for lower stroke speeds can automatically adjust the angle of ***StopTolerance*** downwards (see Stopping distance and tolerance of the stopping distance [📖 55]).

7.3.6.6 Determine response time of the brake (ResponseTime)

ResponseTime defines the **response time** when switching off the press.

The response time is the time between the point at which the shutdown is triggered and the point at which the actual physical braking process is initiated.

Rule for assigning the *ResponseTime*

The actual physical braking process is initiated at the moment at which the brake disk is applied and the braking power or braking torque is established. The response time includes the reaction time of the control system (on average 1.5 x task cycle time), press safety valve and brake. During this period, no braking effect is present, the speed is constant. The sum of the actual reaction times of the control system (1.5 x task cycle time), press safety valve and brake must be determined and configured at the I-variable ***ResponseTime***. If only some, or even none of these reaction times are known, we recommend that you enter the time value for 2 x task cycle times at this I-variable.

7.3.6.7 Stopping distance and tolerance of the stopping distance

The diagram below shows the angular speed in relation to time. The marked areas are therefore proportional to the travelled angles (= paths).

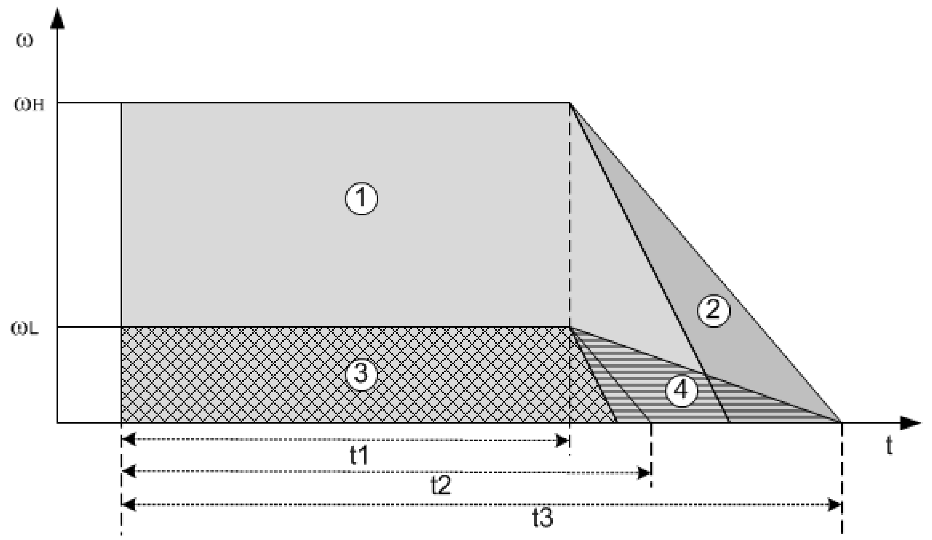


Fig.: Stopping distance (***HighSpeedStoppingDistance***) and tolerance of the stopping distance (***StopTolerance***)

Legend

ω	Angular speed
ωH	Angular speed, defined by ***HighSpeed***
ωL	Angular speed, defined by ***LowSpeed***
①	Stopping angle at the highest speed, defined by ***HighSpeedStoppingDistance***
②	Tolerance at the highest speed, defined by ***StopTolerance***
③	Stopping angle at the lowest stroke speed, defined by ***SwitchOffPoint***
④	Tolerance at the lowest stroke speed
t1	For all speeds constant time, defined by ***ResponseTime***
t2	Warning time; exceeding it will trigger the warning ***DiagStopWarning***
t3	Overall response time; exceeding it will trigger the error ***DiagStopError***

The tolerance defined by ***StopTolerance***② is defined as the angle for the highest angular speed ωH. The overall response time t3 of the press results from the stopping angle ① of the press at the highest stroke speed, and the tolerance ②.

The overall response time is important for calculating the safety distances on the press and it is therefore safety-relevant. Once the overall response time has been defined, it must be adhered to at all the possible speeds.

INFORMATION

Detailed information on this subject can be found in the block description for the FS_CamController (see "Fault detection through dynamic monitoring of stopping distance" in the application notes).

7.3.6.8 Warning threshold (DiagStoppingWarning)

During operation, the rotary cam arrangement constantly calculates a warning threshold automatically. This warning threshold is important when the press is operated at low stroke speed over a longer period. When the warning threshold is exceeded, it means that the press would exceed the monitoring limit after switching to a higher stroke speed.

The warning is displayed by setting the O-variable ***DiagStoppingWarning*** = TRUE.

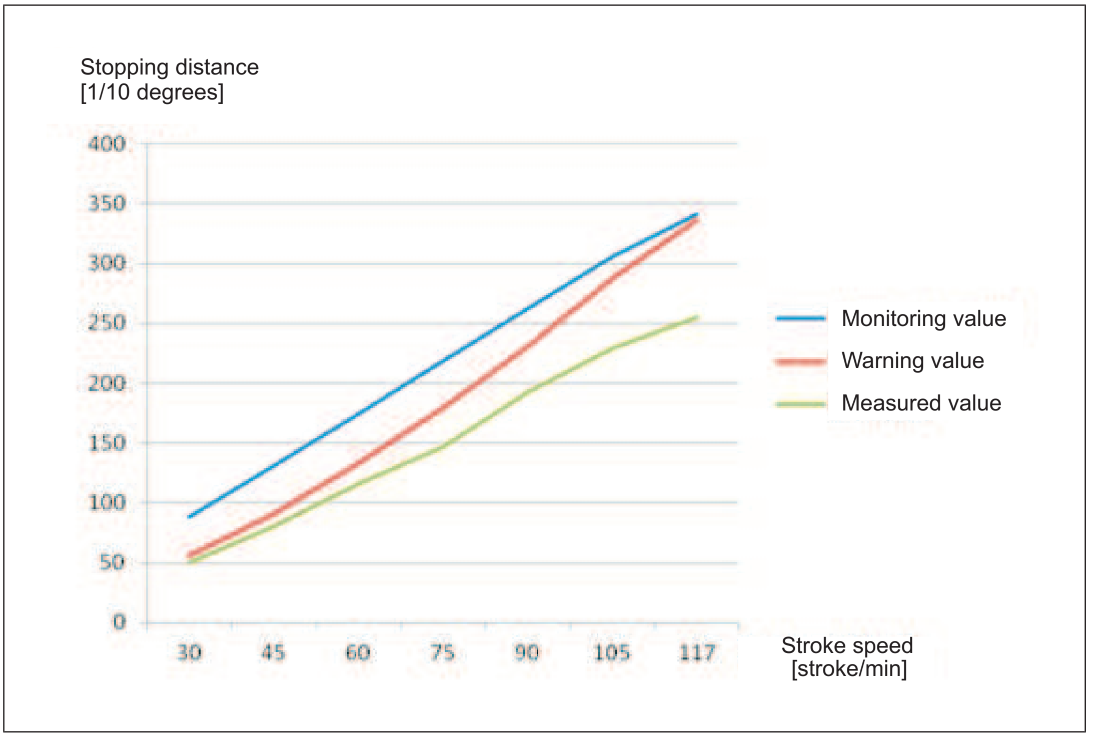


Fig.: Warning threshold on presses with variable stroke speed

7.3.7 Diagnostics

The block reacts to a detected error as follows:

- The O-variable Diag… is set
- Enable is reset (O-variable ***Enable*** = FALSE)
- A message is displayed in the diagnostic list and diagnostic log via the process diagnostics

If an enable has been reset, the reason may be seen from the relevant O-variable.

O-variable	Meaning	
DiagStopWarning	Diagnostic message for switching the stroke speed; The O-variable is set as soon as it is detected that the max. stopping time may be exceeded if the current stroke speed is switched to a higher stroke speed in future. Prerequisite: I-variable ***DisabelCalculatedDynamicCam*** = FALSE	
	TRUE	Warning: The max. stopping time can be exceeded if the machine is switched to a higher stroke speed in future.
DiagParameterError	Monitoring I-variables;	
	TRUE	Fault: • Value range of at least one of the I-variables is outside the permitted range or • At least one of the following I-variables has changed during runtime: – ***RunUpPoint*** – ***SwitchOffPoint*** – ***StopPoint*** – ***StopTolerance***
DiagStopError	Overrun monitoring or dynamic monitoring of stopping distance	
	TRUE	Fault: • Fixed stroke speed During shutdown via ***SwitchOffPoint***, the max. overrun distance was exceeded. • Variable stroke speed During dynamic switch-off via ***SwitchOffPoint***, the actual stopping distance exceeded the max. stopping distance that was calculated based on the speed.
DiagAngleError	Monitoring of angle value;	
	TRUE	Fault: • Value range for ***Angle*** exceeded or • Stroke speed too high Cam areas may be missed because the program is processed cyclically.

O-variable	Meaning	
DiagReadyForReset	Diagnostic information on reset;	
	FALSE	No reset required/possible
	TRUE	Rotary cam arrangement is ready for reset All errors have been rectified, block restarts when the reset button is released

Support

Technical support is available from Pilz round the clock.

Americas
Brazil
+55 11 97569-2804
Canada
+1 888-315-PILZ (315-7459)
Mexico
+52 55 5572 1300
USA (toll-free)
+1 877-PILZUSA (745-9872)

Asia
China
+86 21 60880878-216
Japan
+81 45 471-2281
South Korea
+82 31 450 0680

Australia
+61 3 95600621

Europe
Austria
+43 1 7986263-0
Belgium, Luxembourg
+32 9 3217575
France
+33 3 88104000
Germany
+49 711 3409-444
Ireland
+353 21 4804983
Italy, Malta
+39 0362 1826711
Scandinavia
+45 74436332
Spain
+34 938497433
Switzerland
+41 62 88979-30
The Netherlands
+31 347 320477
Turkey
+90 216 5775552
United Kingdom
+44 1536 462203

You can reach our international hotline on:
+49 711 3409-444
support@pilz.com

Pilz develops environmentally-friendly products using ecological materials and energy-saving technologies. Offices and production facilities are ecologically designed, environmentally-aware and energy-saving. So Pilz offers sustainability, plus the security of using energy-efficient products and environmentally-friendly solutions.

Energy saving by Pilz

BLUECOMPETENCE Alliance Member
Partner of the Engineering Industry Sustainability Initiative

pilz the spirit of safety
Partner of: The Best of German Engineering

CERTIFIED DIN EN ISO 9001 DQS QUALITY SYSTEM REG. NO. 2462

Pilz GmbH & Co. KG
Felix-Wankel-Straße 2
73760 Ostfildern, Germany
Tel.: +49 711 3409-0
Fax: +49 711 3409-133
info@pilz.com
www.pilz.com

PILZ
THE SPIRIT OF SAFETY

1003079-EN-02, 2016-12 Printed in Germany