# SIMATIC S5
# Standard Function Blocks
# S5-100U Closed-Loop Control

**Reference Manual**

**EWA 4NEB 812 6037-02**

## ⚠ WARNING

**Hazardous voltage.**

**Can cause death, severe personal injury, or substantial property damage.**

Restrict use to qualified personnel.
See safety instructions.

Only qualified personnel should install or maintain this equipment after becoming thoroughly familiar with all warnings, safety notices, and maintenance procedures contained in this manual. The successful and safe operation of this equipment is dependent upon proper handling, installation, operation, and maintenance.

The following are definitions of the terms “qualified person,” “danger,” “warning,” and “caution,” as applicable for this document.

**Qualified Person**
One who is familiar with the installation, construction, and operation of this equipment and the hazards involved. In addition, the person should have the following qualifications:

- Be trained and authorized to use and tag circuits and equipment in accordance with established safety practices
- Be trained in the proper care and use of protective equipment in accordance with established safety practices
- Be trained in rendering first aid

**DANGER**

Indicates loss of life, severe personal injury, or substantial property damage will result if proper precautions are not taken.

**WARNING**

Indicates loss of life, severe personal injury, or substantial property damage can result if proper precautions are not taken.

**CAUTION**

Indicates minor personal injury or property damage can result if proper precautions are not taken.


First Edition, March 1990
Printed in the Federal Republic of Germany

*EWA 4NEB 812 6037-02*

# Additional Information Regarding the Standard Function Blocks for S5-100U Closed-Loop Control (6ES5 840-4BC21)

- **Contents of the memory submodule containing the standard function blocks for S5-100U closed-loop control**

**Block list**

| Block Type | Number | Length | Library Number |
|---|---|---|---|
| FB | 201 | 249 | P71200 - S 1201 - A - 1 |
| FB | 202 | 415 | P71200 - S 1202 - A - 1 |
| FB | 203 | 487 | P71200 - S 1203 - A - 1 |
| FB | 204 | 143 | P71200 - S 1204 - A - 1 |
| FB | 205 | 221 | P71200 - S 1205 - A - 1 |
| FB | 206 | 200 | P71200 - S 1206 - A - 1 |
| FB | 207 | 183 | P71200 - S 1207 - A - 1 |
| FB | 208 | 116 | P71200 - S 1208 - A - 1 |
| FB | 209 | 128 | P71200 - S 1209 - A - 1 |
| FB | 210 | 64 | P71200 - S 1210 - A - 1 |
| FB | 211 | 115 | P71200 - S 1211 - A - 1 |
| FB | 212 | 77 | P71200 - S 1212 - A - 1 |
| FB | 213 | 120 | P71200 - S 1213 - A - 1 |
| FB | 214 | 133 | P71200 - S 1214 - A - 1 |
| FB | 215 | 64 | P71200 - S 1215 - A - 1 |
| FB | 216 | 67 | P71200 - S 1216 - A - 1 |
| FB | 217 | 72 | P71200 - S 1217 - A - 1 |
| FB | 218 | 72 | P71200 - S 1218 - A - 1 |
| FB | 219 | 104 | P71200 - S 1219 - A - 1 |
| DB | 130 | 28 | |

- **Handling**

  The "S5-100U Closed-Loop Control" memory submodule has the same design as the EPROM submodule with the machine-readable product designation 6ES5 375-0LA15.
  The submodule has the program number 5 on the PG 615 and 11 on all other programmers.

- **CPU 103 versions for S5-100U**

| Standard Function Blocks for S5-100U Closed-Loop Control Version | Release | |
|---|---|---|
| | CPU 103 6ES5 103-8MA01 | CPU 103 6ES5 103-8MA02 |
| V 1.0 | Z02 | Z02 |

The CPU software release (and that of the PGAS software) can be queried on a programmer equipped with the relevant STEP 5 software via a softkey ("Info", "Syspar").

The CPU software releases listed above are subject to the following constraints:

1) Rounding errors
   When a controller with a D component but no I component (PD controller) is used and supplied with an alternating system deviation of, for instance, (+5 - 5 +5 - 5) counts, the controller output drifts '1' LSB in the direction of the lower output limit on every second change from plus to minus and vice versa.
   Corrective measures: Minimize the amount of integral action (i.e. high integral-action time $T_N$).

2) The internal computing accuracy of OB 251 is + 15 bits plus a 16-bit remainder. Only the internal value range of the D component encompasses + 31 bits. A high amount of integral action in combination with large, step changes in the system deviation may result in an unwanted loss in dynamic performance, and the reduction in the manipulated variable is excessive after a number of sampling intervals.
   Corrective measures: Compensate setpoint changes over a ramp .

# Preface

The "S5-100U Closed-Loop Control" software package provides a set of function blocks for solving closed-loop control tasks with the S5-100U programmable controller (CPU 103). Each function block implements a specific closed-loop control function, e.g. setpoint adjuster, continuous-action controller, filter element, polygon generator, setpoint sequence, etc.

These digital loop controllers are based on the 100U programmable controller's PID algorithm, i.e. OB 251. The standard function blocks generate the environment for the required control structure.

The standard function blocks are linked via defined "marshalling points", i.e. data interfaces that are handled in much the same way as solder tags in hardwired systems.

The modular and highly adaptable "S5-100U Closed-Loop Control" software package provides many advantages over conventional hardwired controllers:

- Simple modification of the control structure without the need for hardware changes
- Easy matching of controller parameters to the needs of the process
- Multiloop control (several control loops driven by a single processor)
- A high degree of flexibility (control loops can be added to an existing SIMATIC control system without additional hardware overhead)
- Full expansion capability (because the loop controllers are linked into the CPU program, you can perform any necessary data conversions in STEP 5).

The objective of this manual is to provide you with detailed information and simplify use of the standard function blocks in the "S5-100U Closed-Loop Control" software package. However, not all problems that might occur in the many and varied applications can be handled in detail in a single manual. If you encounter a problem that is not discussed in this manual, contact your nearest Siemens branch office or representative for help or advice.

# Introduction

The following pages contain information to help you familiarize yourself with the manual.

**Description of contents**

In order to understand the material presented in this manual, the reader must have a basic knowledge of the STEP 5 programming language and of control engineering concepts as a whole.
Since the software package can be used only in conjunction with the S5-100U programmable controller, it has been assumed that you, as user, have an S5-100U Manual.

- Basic Concepts of Control Engineering
- System Overview
- Software Start-Up
  This section includes a simple example and a complete STEP 5 program.
- Description of the Standard Function Blocks
- Sophisticated Applications
- Complex Example: Cascade Control

In the Appendix you will find a tabular overview of the function blocks.

Please use the forms at the back of this manual for any suggestions or corrections you may have and return them to us. This will help us make the necessary improvements in the next edition.

**Training courses**

- Course S 22
  Digital PID control with SIMATIC S5

For more information, please contact your nearest Siemens representative.

**Reference literature**

- Closed-Loop Control with SIMATIC S5
  Volume 1: Basic Concepts
  Siemens AG, Nuremberg, 1989
  Order No.: E80850-C331-X-A1-7600
- Volume 2: Sample Applications for Software-Based Closed-Loop Control
  Order No.: 6ZB5310-0AA01-0BA1 (available in German only)
- Introduction to Electronic Control Engineering
  Siemens AG, Berlin and Munich, 1976
  Order No.: ISBN 0-85501-290-0

**Conventions**

In order to improve the readability of the manual, a menu-style breakdown was used, i.e.:

- The individual chapters can be quickly located by means of a thumb register.
- There is an overview containing the headings of the individual chapters at the beginning of the manual.
- Each chapter is preceded by a breakdown of its subject matter.
  The individual chapters are subdivided into sections and subsections. **Boldface** type is used for further subdivisions.
- Figures and tables are numbered separately in each chapter. The page following the chapter breakdown contains a list of the figures and tables appearing in that particular chapter.

Certain conventions were observed when writing the manual. These are explained below.

- A number of abbreviations have been used.
  Example: Programmable controller (PLC)
- Footnotes are identified by a subscript consisting of a small digit (e.g. ”1”), or asterisk (”*”). The actual footnote is at the bottom of the page or beneath a table.
- Cross-references are shown as follows:
  ”( 7.3.2)” refers to subsection 7.3.2.
  No references are made to individual pages.

- All dimensions in drawings etc. are given in millimeters (mm).
- Values may appear as binary, decimal or hexadecimal numbers. The relevant number system is shown by a subscript, e.g.$F000_H$.
- Information of particular importance is framed in grey-colored bars. The top bar contains a heading indicating the type of information involved.

**Note:**

Additional information; emphasizes an exception or special feature.

**CAUTION:**

Provides information which you must note with particular care in order to prevent damage to the hardware or software.

Manuals can only describe the current version of the programmable controller. Should modifications or supplements become necessary in the course of time, a supplement will be prepared and included in the manual the next time it is revised. The relevant version or edition of the manual appears on the cover. The present manual is edition "1". In the event of a revision, the edition number will be incremented by "1".

- Remarks Form

  The Remarks Form is provided for your comments and suggestions. If you are in the United States, please use the postage-paid form.

## Conventions

The following conventions are used in this book and are listed for your reference.

| Convention | Definition | Example |
|---|---|---|
| | A box that indicates a type of hazard, describes its implications, and tells you how to avoid the hazard, is a safety notice. Some safety notices include a graphic symbol representing an electrical or radio-frequency hazard. All safety notices have one of the following levels of caution:<br><br>• Indicates that loss of life, severe personal injury, or substantial property damage will result if proper precautions are not taken.<br><br>• Indicates that loss of life, severe personal injury, or substantial property damage can result if proper precautions are not taken.<br><br>• Indicates that minor personal injury or property damage can result if proper precautions are not taken. | |

# 1 Basic Concepts of Control Engineering

## Figures



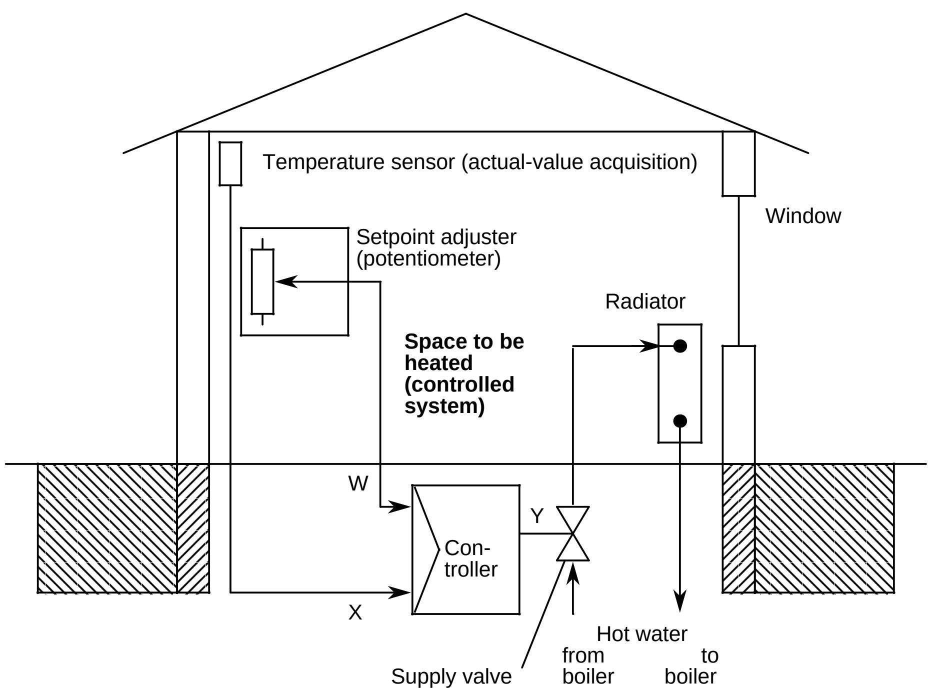
## Tables

# 1 Basic Concepts of Control Engineering

## 1.1 Principles

Closed-loop control is a procedure in which the variable to be controlled (actual value) is continuously measured and compared with another variable (setpoint) until the controlled variable corresponds as accurately as possible with the setpoint.

The principles of closed-loop control can best be explained on the basis of a simple example.




w Setpoint
x Actual value
y Manipulated variable

**Figure 1-1. Closed-Loop Control of a Space Heating System**

Figure 1-1. shows a simplified diagram of a space heating system. The boiler with its burner, and the oil tank, are not shown.

The control procedure is as follows:

- The desired room temperature, i.e. the setpoint (w), is set on the potentiometer. The potentiometer is referred to as the "setpoint adjuster", and supplies the reference variable (w).
- The temperature sensor measures the room temperature. The temperature sensor is the "converter" for the controlled variable, i.e. the room temperature, and supplies the actual value (x) processed in the "controlling system". The value of the controlled variable is returned to the controller's input over this path ("feedback").
- Next, a comparator must compare the reference variable (w) with the controlled variable (x) to ascertain the system deviation ($x_d$). The system deviation is obtained by subtracting the controlled variable (x) from the reference variable (w). The control algorithm then generates the manipulated variable (y).
- The supply valve controls the inflow of hot water to the radiator.

## 1.2 Controlled Systems

In Figure 1-1., the room to be heated represents the controlled system. A controlled system can be made up of a number of different parts. Each part of the system is identified by its response characteristics. It is essential to know and be able to assess these characteristics if the controlled variable produced by the system is to be handled optimally. Only then is it possible to select and tune a suitable controller.

The dynamic response of a controlled system can differ considerably:

| | |
|---|---|
| Furnace control system | Hours |
| Motor control system | Seconds |

* The response characteristics are based on the ability to store energy. The dynamic response depends on the size of the room, i.e. of the controlled system.

Table 1-1. gives an overview of the elementary transfer elements and the responses of these elements to step changes at the input. The combination of these elements forms the actual controlled system. The performance characteristics of these elements are described in more detail in subsequent subsections on the basis of examples.

A transfer element is normally represented as a rectangle with an arrow indicating the input. Another arrow pointing away from the rectangle identifies the output. It is also common to find a stylized transfer function drawn in the rectangle to make the function of the element clear.

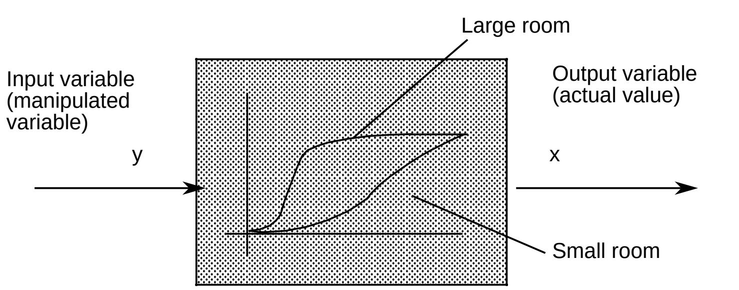
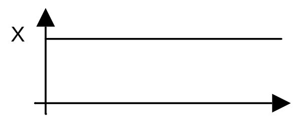
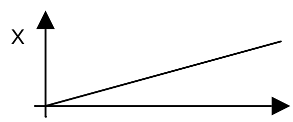
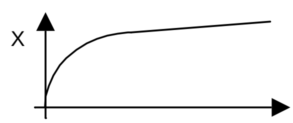
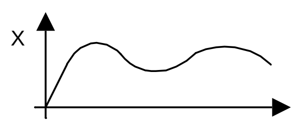
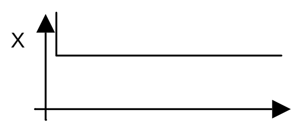
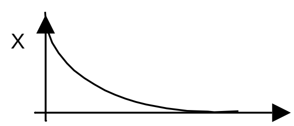
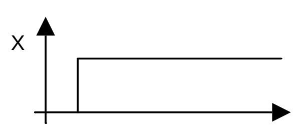



**Figure 1-2. Transfer Elements**

In order to determine the time-sensitive and amplitude-sensitive characteristics of a transfer element, easily reproducible signals are constantly applied to the input and the resulting signals at the output evaluated.

The response characteristics can best be evaluated from a step change in the input signal to the transfer element. This is called a unit step.

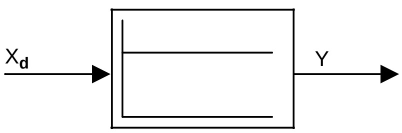
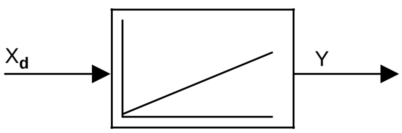
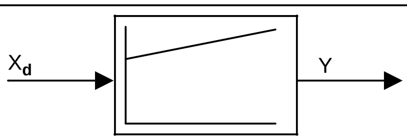
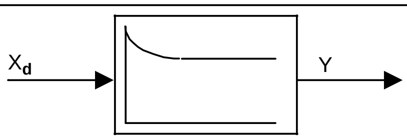
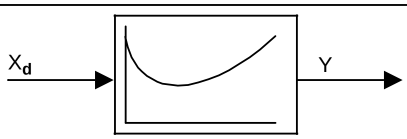
**Table 1-1. Elementary Transfer Elements and Their Step Responses**

| Transfer Element | | Step Response | Output Variable x ... |
|---|---|---|---|
| **Characteristic** | **Abbre-viation** |  | |
| Proportional | P |  | ... is instantaneously propor-tional to input variable y, e.g. flow rate |
| Integral | I |  | ... is the time integral of input variable y, e.g. level |
| 1st order lag | P-T1 |  | ... is proportional to input variable y, but delayed according to an exponential function, e.g. temperature |
| 2nd order lag | P-T2 |  | ... is proportional to input variable y, but delayed according to a transient function, e.g. iron |
| Derivative (rate) | PD |  | ... is proportional to input variable y; the differential of input variable y is also included |
| Delayed decay | D-T1 |  | ... becomes zero. A change in the input variable gives the differential of input variable y with delayed decay |
| Dead time | P-T1 |  | ... is proportional to input variable y, but delayed by a fixed time period, e.g. belt conveyor |

## 1.3 Controllers

The PID algorithm forms the basis for tuning the different variants and types of controllers.

**Table 1-2. Controller Variants**

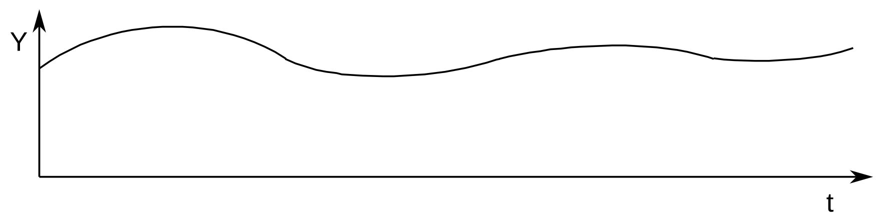
| | | | |
|---|---|---|---|
| P controller |  | Proportional action | Reacts quickly to a system deviation, but a steady-state system deviation or offset always remains |
| I controller |  | Integral action | No steady-state system deviation, but reacts more slowly to a system deviation than does a P controller |
| PI controller |  | Proportional-plus-integral action | No steady-state system deviation. Corrects a system deviation more quickly than an I controller. Suitable for all controlled systems. |
| PD controller |  | Proportional-plus-derivative action | Reacts very quickly to a system deviation, but a steady-state system deviation always remains |
| PID controller |  | Proportional-plus-integral-plus-derivative action | No steady-state system deviation. Highest attainable control quality, but difficult to tune. |

The standard function blocks in the "Closed-Loop Control with the S5-100U" software package contain three types of controllers. Which type you use depends entirely on the final control element that the controller drives.

- Continuous-action controller
- Step-action controller (hysteresis)
- Pulse controller

**Continuous-action controllers**

The characteristic feature of continuous-action controllers is that manipulated variable Y can assume any value within the total correcting range. A continuous-action controller's output value is an analog value. A continuous-action controller outputs a manipulated variable even when the system deviation has been completely eliminated ($X_d$ =0).
Continuous-action controllers are required when the final control element has to have an uninterrupted actuating signal (permanently applied voltage at a valve, for instance).




Y  Manipulated variable  t  Time

**Figure 1-3. Characteristic of the Manipulated Variable on a Continuous-Action (K) Controller**

**Step-action controllers**

A step-action controller provides three operating positions; e.g. FORWARD-STOP-REVERSE.
A step-action controller is used to drive a latching final control element, e.g. a screw with traveling nut. The motor provides the relevant pulse only when the actual value differs from the setpoint ($X_d$  0).

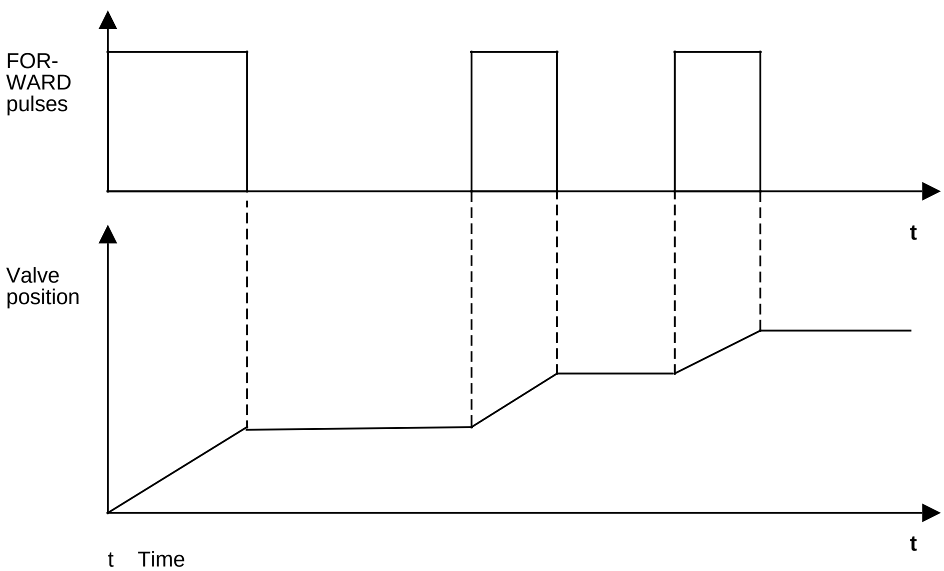
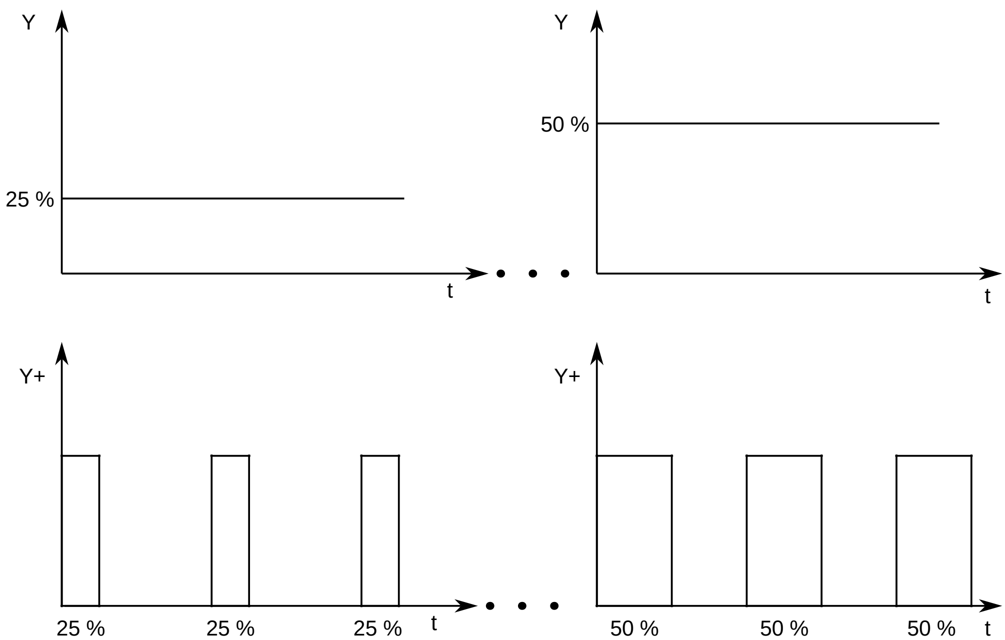



**Figure 1-4. Response Characteristics of a Step-Action (S) Controller with a Series-Connected I Element (Motor)**

**Pulse controllers**

When a pulse controller is used, the continuous output signal is converted into a PDM signal. The controller response is determined by the mark-to-space ratio.

When used, for example, for temperature control, the controller can function as either two-step (heating only) or three-step (heating and cooling) controller.




Y Manipulated variable
Y+ Actuating pulse
t Time

**Figure 1-5. Signals Output by the Pulse Controller**

## 1.4 Control Loops

Control loops are formed by the totality of all elements participating in the feedback control system. A control loop is divided into control equipment (controller) and controlled system. The simplest form of closed-loop control is the single control loop, in which the controller matches controlled variable x to reference variable w.

Let us use the closed-loop-controlled space heating system from Section 1.1 ("Principles"), which is illustrated in Figure 1-6. in the form of a signal-flow diagram, as an example.

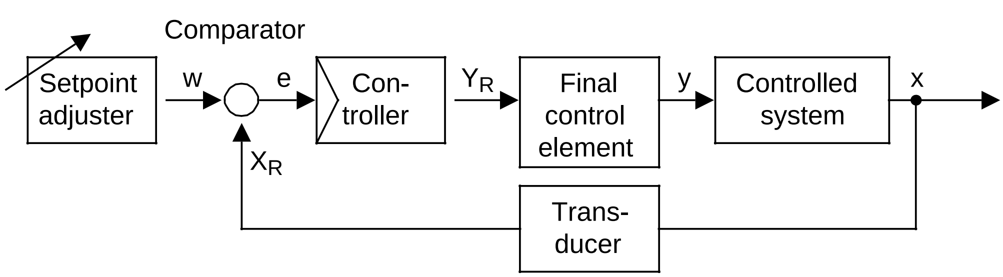
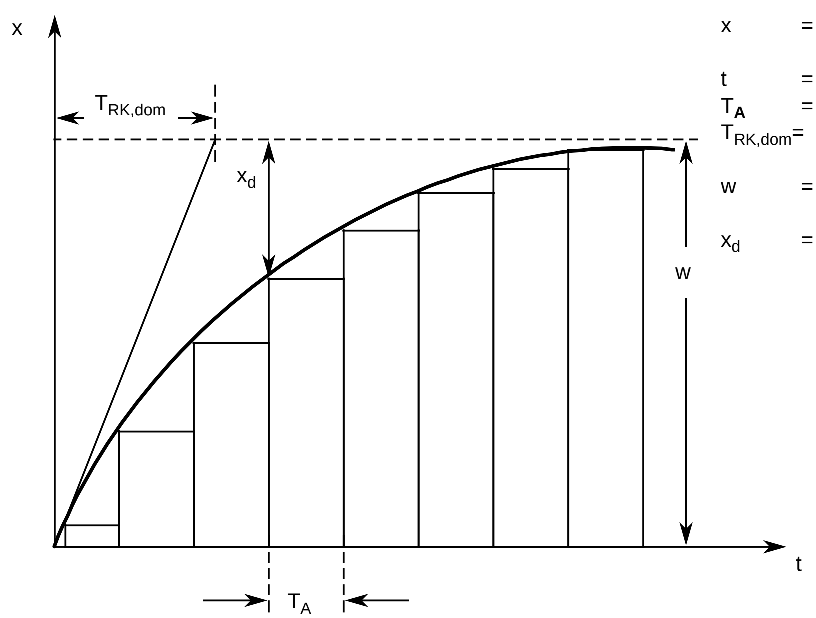



**Figure 1-6. Single Control Loop**

## 1.5 Digital Controllers

### 1.5.1 Sampling Interval

The time basis for processing of the control program is sampling interval $T_A$.
In this program, system deviation $X_d$ is computed at regular intervals (sampling interval $T_A$) from the current value of controlled variable x (actual value) and reference variable w (setpoint).
The sampling interval is a characteristic feature of digital control loops, and represents the amount of time between two control program passes .
The longer the sampling interval, the more the central processor is off-loaded, and thus the larger the number of control loops it can process successively within one sampling period.
The sampling interval may not be made infinitely short. Decisive for the duration of a sampling interval are program runtimes and the encoding times required by the ADCs and DACs.



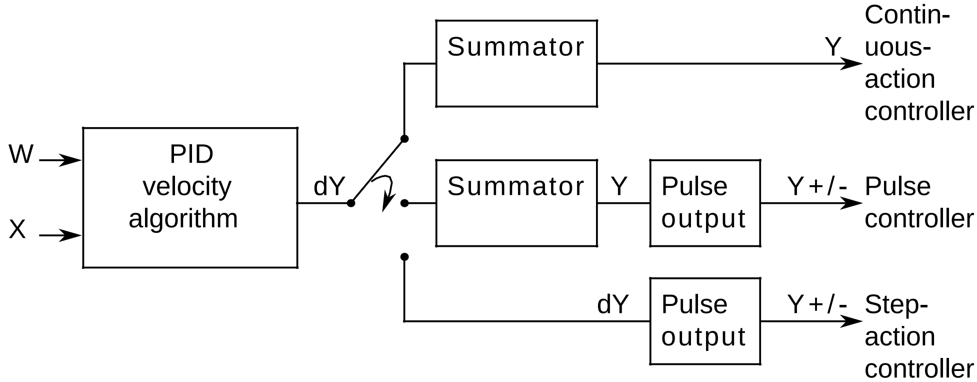

x = Controlled variable
t = Time
$T_A$ = Sampling interval
$T_{RK,dom}$ = Dominant system time constant
w = Reference variable/setpoint
$x_d$ = System deviation

**Figure 1-7. Sampling Principle**

**Note :**

The sampling interval should be about $\frac{1}{10}$ of the dominant system time constant.
The dominant system time constant can be ascertained from the control loop's step response.

## 1.5.2 Structure of Digital Controllers




| | | | |
|---|---|---|---|
| dY | Correction increment | W | Setpoint |
| Y | Manipulated variable | X | Actual value |
| Y+/- | Actuating pulses | | |

**Figure 1-8. Structure of Digital Controllers**

The following PID algorithm (OB 251) is used in the "S5-100U Closed-Loop Control" software package:

$$dY_k = \mathbf{K} \times (dPW_k \mathbf{P} + d\mathbf{I}_k + d\mathbf{D}_k)$$

P component — I component — D component

The PID algorithm computes correction increment dY from system deviation $X_d$.

- Continuous-action controller

  The correction increments are added by a summator and output as manipulated variable Y.

- Pulse controller

  The signal output by the summator (Y) is also forwarded to the pulse output, where manipulated variable Y is converted into actuating pulses Y+/-; these pulses are then output to the final control element.

- Step-action controller

  Correction increments dY are forwarded directly to the pulse output, where they are converted into actuating pulses Y+/-.

# 2 System Overview

**Figures**



**Tables**

# 2 System Overview

## 2.1 General Remarks

**The 55-1** OOU closed-loop control software package contains function blocks representing freely configurable closed-loop control system. Each function block is a self-contained unit. You can combine the function blocks as needed and interconnect them by assigning the necessary function parameters. The function parameters are initialized in data blocks allocated to the function blocks. You can thus generate the control structure that you need to suit your particular application.

Figure 2-1. shows an overview of the standard function blocks in the "s5-1OOU Closed-Loop Control" software package. The "switches" between the FBs are replaced by data words in the control program.

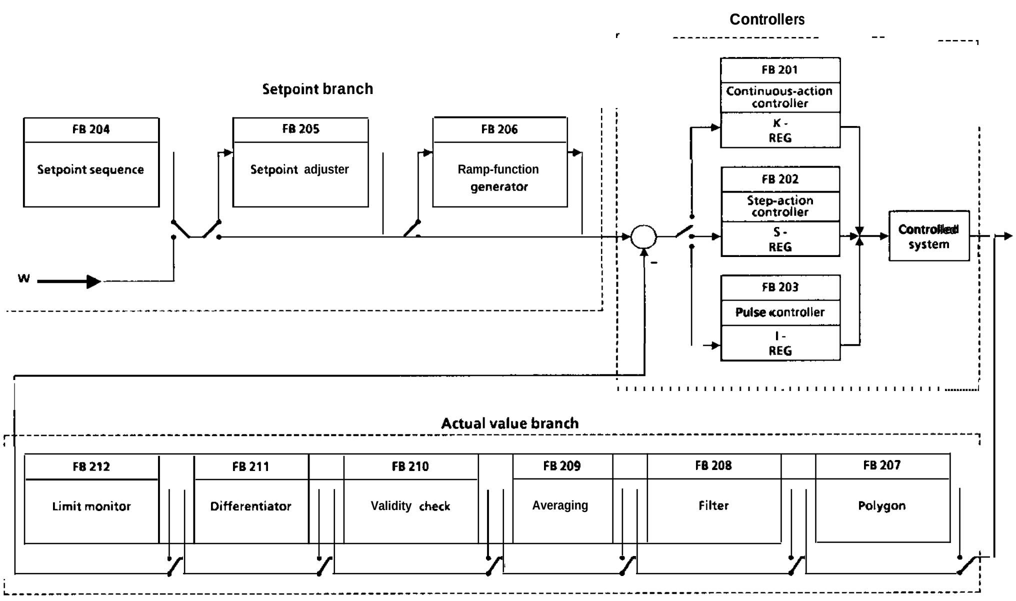



**Figure 2-1. Overview of the Standard Function Blocks in the "S5-100U Closed-Loop Control" Software Package**

**CAUTION:**

Most of the "S5-100U Closed-Loop Control" function blocks in which invalid input parameters would generate undefined output signals have a PAFE (parameter assignment error) parameter. When the values of input parameters are out of range, PAFE bits are set in the allocated data word; you can analyze these bits to locate the errors.

The "S5-100U Closed-Loop Control" function blocks that have no PAFE parameter or read out data from data blocks do not check the input parameters for validity. In this case, you must take particular care to make sure that the input parameters are within the permissible range.

## 2.2 Typical Applications

**Table 2-1. Typical Controller Applications**

| Type of Controller | Application |
|---|---|
| Continuous-action controller | Continuous-action controllers are suitable for most applications:<br>- Temperature control<br>- Pressure control<br>- Flow control<br>- Liquid level control<br>- Speed control<br>- Follow-up control |
| Step-action controller | Step-action controllers are used, for example, when electric motors are employed as actuators for driving a wide variety of final control elements, such as butterfly valves, slide valves, etc. These controllers require integral-action final control elements. |
| Pulse controller | A pulse controller can be used in a temperature control system with a self-adjusting final control element. This is the case, for example, when measuring electrical resistances. A specific mark-to-space ratio is set for each constant temperature value. |

## 2.3 Structure of the Software

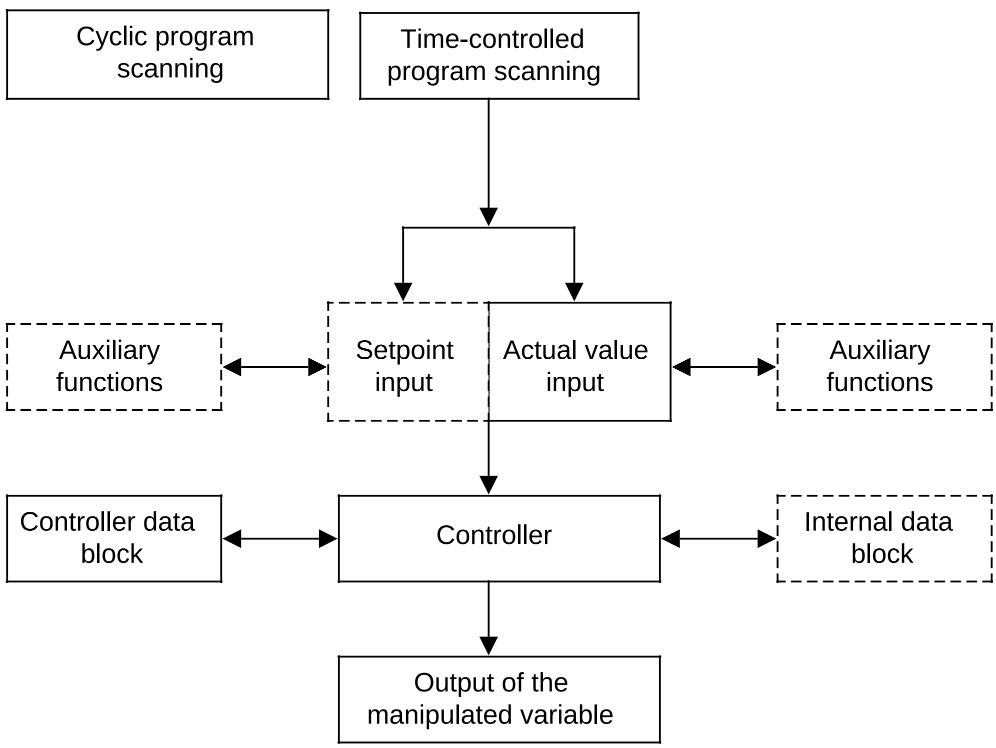



**Figure 2-2. Structure of the Software**

Figure 2-2. shows the structure of the S5-100U Closed-Loop Control software.

To help you understand this structure, the functions performed by the various blocks are described in detail in the following subsections.

**Cyclic program scanning**

To scan the application program cyclically, the system program invokes organization block OB 1. The first STEP 5 statement in organization block OB 1 is also the first statement in the application program, and thus the beginning of the program.

**Time-controlled program scanning**

Timed-controlled program scanning (setting the sampling interval) is made possible by programming OB 13. The program is time-controlled when a timing signal causes the processor to interrupt cyclic program scanning and execute a specific routine. After executing this routine, the processor resumes the cyclic program scan at the point of interruption.

**Reading/entering the setpoint**

Use standard function block FB 250 ( 3.1) if you want to enter the setpoint as an analog value.
FB 250 reads an analog value from an analog input module and makes a value (XA) available at the output; this value is scaled, and lies in the range specified.
The following FBs are provided as auxiliary functions:

- Setpoint sequence FB 204
- Setpoint adjuster FB 205
- Ramp-function generator FB 206

Input of the setpoint via the analog input module, and thus the FB 250 standard function block call, can be omitted if you write the setpoint directly into the data word of controller data block DBRE.

**Reading/entering the actual value**
Standard function block FB 250 also reads in the actual value from the analog input module.

The "S5-100U Closed-Loop Control" software package also provides powerful auxiliary functions for influencing the actual value, for instance. These functions can be invoked as required.

- Polygon FB 207
- Filter FB 208
- Deadtime element FB 209
- Averaging FB 210
- Validity check FB 211
- Differentiator FB 212
- Limit monitor FB 213

**Controllers**
Three types of controller can be used for process control:

- Continuous-action controller FB 201
- Step-action controller FB 202
- Pulse controller FB 203

**Internal data block DBOB**
DBOB is a controller-specific data block, and is used to buffer internal controller data.

**Controller data block DBRE**
DBRE serves as user interface to the controller, as buffer for controller data, and as interface between the controllers in a multicontroller configuration.

**Output of the manipulated variable**
Standard function block FB 251 ( 3.1), which is integrated in the S5-100U programmable controller, is used to output the manipulated variable. This function block outputs the computed analog value to an analog output module.

## 2.4 Ordering Data

| | | Order No. |
|---|---|---|
| "S5-100U Closed-Loop Control" standard function blocks | | |
| Plugin submodule with reference manual | | |
| English | | 6ES5 840-4BC21 |
| German | | 6ES5 840-4BC11 |
| Italian | | 6ES5 840-4BC51 |
| | | |
| CPU 103 | | |
| Without manual | | 6ES5 103-8MA02 |
| With manual | English | 6ES5 103-8MA22 |
| | German | 6ES5 103-8MA12 |
| | French | 6ES5 103-8MA32 |
| | Spanish | 6ES5 103-8MA42 |
| | Italian | 6ES5 103-8MA52 |
| | | |
| Manual for the S5-100U programmable controller | | |
| English | | 6ES5 998-0UB22 |
| German | | 6ES5 998-0UB12 |
| French | | 6ES5 998-0UB32 |
| Spanish | | 6ES5 998-0UB42 |
| Italian | | 6ES5 998-0UB52 |
| | | |
| Analog input modules | | |
| 4 x ± 50V | Galvanically isolated | 6ES5 464-8MA11 |
| 4 x 0 to 10V | | 6ES5 466-8MC11 |
| 4 x ± 1V | Galvanically isolated | 6ES5 464-8MB11 |
| 4 x ± 10V | Galvanically isolated | 6ES5 464-8MC11 |
| 4 x ± 20mA | Galvanically isolated | 6ES5 464-8MD11 |
| 4 x ±4 to 20mA | Galvanically isolated | 6ES5 464-8ME11 |
| 2 x PT 100/± 500mV | Galvanically isolated | 6ES5 464-8MF11 |
| | | |
| Analog output modules | | |
| 2 x ± 10V | Galvanically isolated | 6ES5 470-8MA11 |
| 2 x ± 20mA | Galvanically isolated | 6ES5 470-8MB11 |
| 2 x + 4 to 20mA | Galvanically isolated | 6ES5 470-8MC11 |
| 2 x + 1 to 5V | Galvanically isolated | 6ES5 470-8MD11 |

# 3 Software Start-Up

## Figures



## Tables

3

# 3 Software Start-Up

## 3.1 procedures

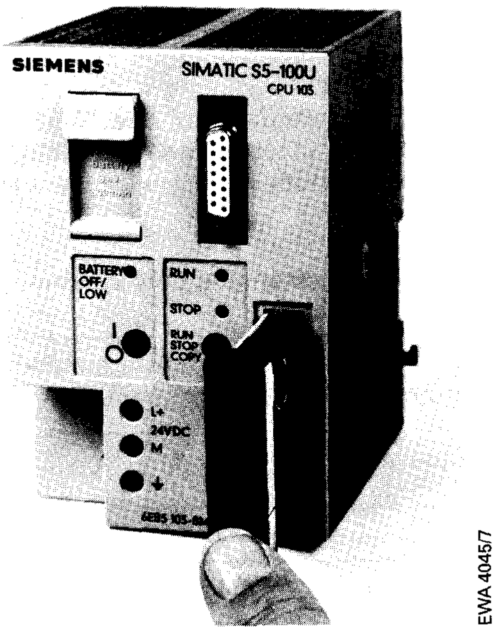



Figure 3-1. Memory Submodule in the Programmable Controller

The "S5-1 OOU Closed-Loop Control" software package is delivered in the form of a memory submodule, and can be put into service once the memory submodule has been plugged into the CPU 103 or the programmer/personal computer.

**Note :**

Please observe the information presented in the relevant manuals when plugging the memory submodule into the programmer or programmable controller.

The control loop must be carefully planned and defined before putting the software into operation.

- **Technical task definition**
  Definition of a controller-independent schematic in the form of a block diagram. The overview diagram in the Appendix will assist you in this task.

- **Solution proposal**
  Conversion of the block diagram into a solution proposal that takes into account the specific characteristics of the "S5-100U Closed-Loop Control" software package. When multiloop and multivariable control is involved, the proposed solution must include in particular the interconnecting structure of the various controllers.

The software should be started up on a step-by-step basis:

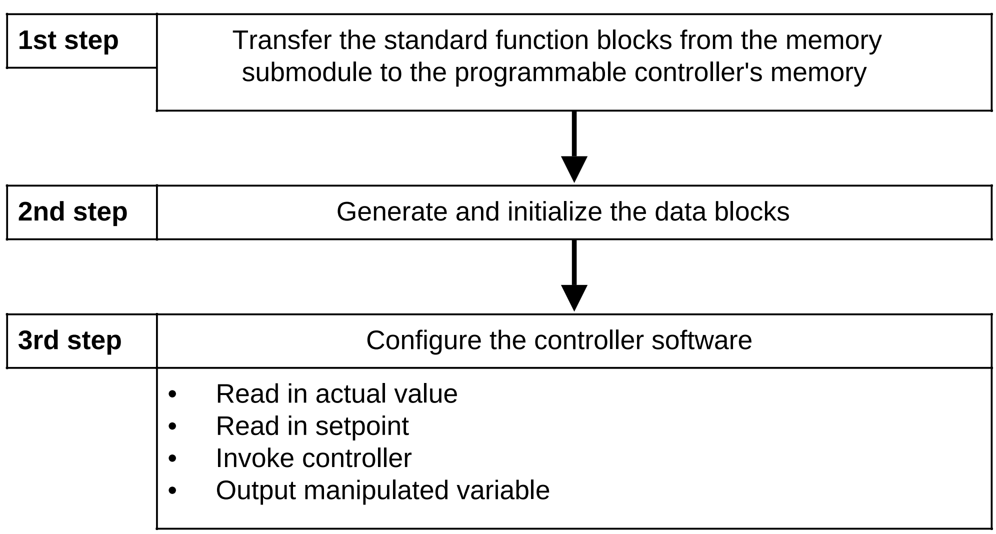



**Figure 3-2. Start-Up**

**1st step**

Memory submodule plugged into the programmer:
The function blocks are transferred from the memory submodule to the programmable controller's memory by invoking the programmer function "Transfer blocks, EPROM PLC".
For more detailed information, consult the programmer manual.

Memory submodule plugged into the programmable controller:
The function blocks are transferred to the programmable controller's memory by setting the mode selector on the PLC to "COPY". This procedure is described in detail in the S5-100U manual ( 4.6.2).
The FBs that are not needed should be deleted to save space in memory.

**2nd step**

A closed-loop control system requires one DBRE data block, which is shared by all control loops in that system, and one (internal) DBOB data block for each control loop. These data blocks are not part of the standard software package, and you must generate and initialize them yourself.

- Controller data block DBRE
  The DBRE data block is generated with the programmer's "Input" function. The number of data words that must be initialized depends on the control structure. In the basic structure (single control loop), for example, you must initialize nine data words and then transfer the data block to PLC memory.

- Internal data block DBOB
  This data block is also generated with the programmer's "Input" function, and must comprise at least 54 data words.

**Note:**

When implementing multicontroller systems, you must generate a 54-word DBOB data block for each controller.

In contrast to DBOB, DBRE is shared by all controllers, and is used to store the parameters for each individual controller.

**3rd step**

**Configuring the control system software**

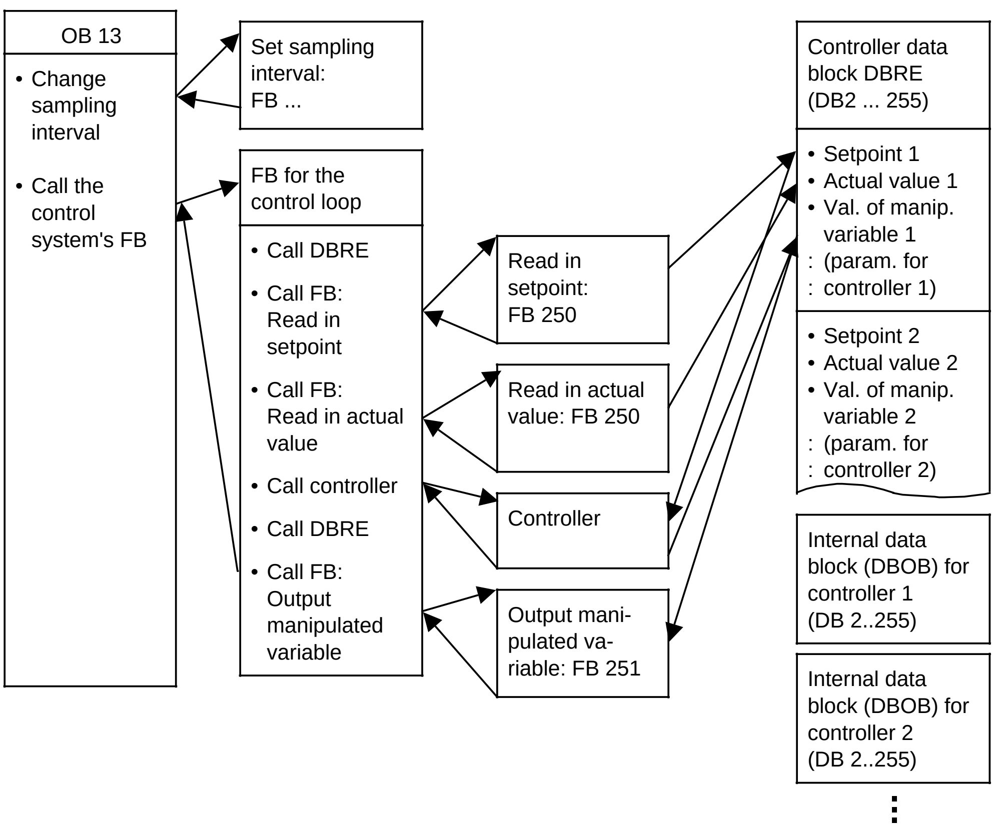



**Figure 3-3. Program Structure of the Closed-Loop Control System**

Figure 3-3. shows the program structure for a simple control loop. Configure your control system as follows:

- **Read in the required setpoint:**
  Use function block FB 250, which is integrated in the CPU 103, to read in the setpoint over an analog input module (refer to the manual for the S5-100U programmable controller ( 7.6.5). A data word in the DBRE must be allocated to the digitized setpoint (XA parameter in FB 250). The actual setpoint variable is then entered in this data word.

**Table 3-1. Calling and Initializing FB 250**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| BG | Slot number | DKF | 0 to 7 | : JU FB 250<br>NAME : RLG:AE<br>BG :<br>KNKT :<br>OGR :<br>UGR :<br>EINZ :<br>XA :<br>FB :<br>BU : |
| KNKT | Channel number<br>Channel type | DKY | KY= x,y<br>x=0 to 3<br>y=3 to 6<br>3: Absolute-value represent. (4 to 20 mA)<br>4: Unipolar represent.<br>5: Bipolar absol. number<br>6: Bipol.fixed-point num. | |
| OGR | Upper limit of output value | DKF | -32768 to +32767 * | |
| UGR | Lower limit of output value | DKF | -32768 to +32767 * | |
| EINZ | Irrelevant | | | |
| XA | Output value | QW | Scaled analog value is "0" on a wirebreak | |
| FB | Error bit | Q BI | Is "1" on a wirebreak or in the case of an invalid channel number, slot number or channel type | |
| BU | Range violation | Q BI | Is "1" when the nominal range is exceeded | |

* KF = -2047 to 2047 applies here, as the controller resolution is limited.

**Note:**

If you enter the control setpoint directly into a data word in data block DBRE, it is not necessary to input the setpoint over the analog input module and thus invoke function block FB 250. Which of these options you choose depends on the structure of the closed-loop control system.

- **Reading in the required actual value:**
  Function block FB 250 is also used to read in the actual value. Here, too, the digitized output value must be allocated to a data word in data block DBRE. For information on calling and initializing FB 250, see Table 3-1.

- **Calling the controller:**
  The "S5-100U Closed-Loop Control" software package provides the following types of controllers, depending on the structure of the control system and the demands made on it:
  - Continuous-action controller FB 201
  - Step-action controller FB 202
  - Pulse controller FB 203

- **Output of the required manipulated variable:**
  The manipulated variable is output via function block FB 251, which is integrated in the CPU 103. FB 250 reads the digitized value of the manipulated variable from the appropriate data word of data block DBRE.

**Table 3-2. Calling and Initializing FB 251**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| XE | Analog value to be output | I W | Input value (two's complement) in range UGR to OGR | : JU FB 251<br>NAME : RLG:AA<br>XE :<br>BG :<br>KNKT :<br>OGR :<br>UGR :<br>FEH :<br>BU : |
| BG | Slot number | D KF | 0 to 7 | |
| KNKT | Channel number Channel type | D KY | KY= x,y<br>x=0;1<br>y=0;1<br>0: Unipolar representation<br>1: Bipolar fixed-point number | |
| OGR | Upper limit of output value | D KF | -32768 to +32767 * | |
| UGR | Lower limit of output value | D KF | -32768 to +32767 * | |
| FEH | Limit value specification error | Q BI | Is "1" when UGR=OGR or when the channel number, slot number or channel type is invalid | |
| BU | Input value exceeds upper (OGR) or lower (UGR) limit | Q BI | Is "1" when XE exceeds limit (UGR; OGR)<br>XE assumes the limiting value | |

* KF = -2047 to 2047 applies here, as the controller resolution is limited.

The controller reads the digitized actual value and setpoint from data block DBRE and outputs the computed value of the manipulated variable to a data word in that same data block.

- **Generating the function block for the control loop:**
  The function block is freely selectable (FB 0 to 200), and may contain the following:
  - Call DBRE
  - Call FB: Read setpoint
  - Call FB: Read actual value
  - Call controller
  - Call DBRE
  - Call FB: Output manipulated variable

- **Generating OB 13:**
  Time-controlled scanning is made possible through the inclusion of the control loop in OB 13. The control routine is then invoked automatically at specific intervals (the default interval is 100 ms).
  The following should be programmed in OB 13:
  - Change sampling interval
  - Call function block for the control loop (FB 0 to 200)

## 3.2 Controller Start-Up

### 3.2.1 Resetting the Controllers

All of the controllers require internal data; this data is stored in data block DBOB. Internal data words in DBOB can be reset by invoking function block FB 215, thus preventing the controller from being continually on-loaded by data that it no longer needs.
You can reset up to four DBOBs, i.e. four controllers, with a single call. FB 215 should be invoked in restart OB 21 only.

**Table 3-3. Calling and Initializing FB 215**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| DB01 | DBOBs (data blocks) to be reset | D KY | KY=x,y<br>2 to 255, 2 to 255 | : JU FB 215<br>NAME : ANLAUF<br>DB01 :<br>DB02 : |
| DB02 | | D KY | KY=x,y<br>2 to 255, 2 to 255 | |

**Note:**
If you want to reset one DBOB only, set parameter y (second controller) to zero, e.g. 2,0.

### 3.2.2 Estimating the Sampling Interval

The controlled variable is read in via analog input modules (6ES5 464..., 6ES5 466...). The analog values are converted into the corresponding digital values. Because analog input modules require a certain amount of time for conversion, the digital value is not available to the controller until the so-called conversion, or encoding time, has elapsed.
It is therefore often necessary to estimate the sampling interval, as it depends on the amount of time the analog input module needs for decoding and on the execution time of the time-controlled program:

**Table 3-4. Execution Times of the Function Blocks**

| Name | FB Number | Execution Time |
|---|---|---|
| Continuous-action controller | FB 201 | 12.0 ms |
| Step-action controller | FB 202 | 12.0 ms |
| Pulse controller | FB 203 | 16.0 ms |
| Setpoint adjuster | FB 204 | 2.5 ms |
| Setpoint sequence | FB 205 | 8.0 ms |
| Ramp-function generator | FB 206 | 2.5 ms |
| Polygon generator | FB 207 | 5.0 ms |
| Filter | FB 208 | 5.0 ms |
| Deadtime element | FB 209 | 10.0 ms |
| Averaging | FB 210 | 2.0 ms |
| Validity check | FB 211 | 4.0 ms |
| Differentiator | FB 212 | 8.0 ms |
| Limit monitor | FB 213 | 1.5 ms |
| Hysteresis | FB 214 | 3.0 ms |
| Reset | FB 215 | 10.0 ms |
| Time-slice distributor | FB 216 | 2.0 ms |
| Save scratch flags | FB 217 | 3.0 ms |
| Load scratch flags | FB 218 | 4.0 ms |
| Actual value/setpoint converter | FB 219 | 4.0 ms |
| Read in and scale analog value | FB 250 | 3.0 ms |
| Output and scale analog value | FB 251 | 6.0 ms |

**Table 3-5. Encoding Times of the Analog Input Modules**

| Name | Encoding Time |
|---|---|
| 6ES5 464...module | 400 ms |
| 6ES5 466...module | 20 ms |

The rule-of-thumb is:

Sampling interval > Execution time of the control program and
> Encoding time of the analog input module

The sampling interval for a single loop is determined below as an example:

- Execution time of the control program

**Table 3-6. Execution Time of the Control Program**

| Function | FB number | Execution time |
|---|---|---|
| Read in setpoint | FB 250 | 3 ms |
| Read in actual value | FB 250 | 3 ms |
| Continuous-action controller | FB 201 | 12 ms |
| Output manipulated variable | FB 251 | 6 ms |
| | Total | 24 ms |

- Encoding time of the analog input module

| Module | Encoding Time |
|---|---|
| 6ES5 464 ...analog input module | 400 ms |
| 6ES5 466 ...analog input module | 20 ms |

The minimum sampling intervals are thus as follows:

| | | |
|---|---|---|
| Sampling interval | > Execution time of the control program: | 24 ms |
| | > Encoding time of the 6ES5 464 ... analog input module: | 400 ms |
| | Sampling interval to be set: **400 ms** | |
| Sampling interval | > Execution time of the control program: | 24 ms |
| | > Encoding time of the 6ES5 466 ... analog input module: | 20 ms |
| | Sampling interval to be set: 24 ms **30 ms** | |

**Setting the sampling interval**

The sampling interval is preset to 100 ms (KF=10), and can be modified in increments of 10 ms. The time factor is in system data byte RS 97. The sampling interval can be changed in one of the following organization blocks:

OB1, OB13, OB21, OB22

As an example, OB 13 is to be invoked at 500 ms intervals. To do this, invoke function block FB 180, which you must program yourself, in an organization block.

```
FB 180

L KF + 50    Load interval into accumulator 1 (500 ms)
T RS 97      Transfer to RS 97
BE
```

**Note:**

- The sampling interval should be equal to or less than 1/10 of the dominant system time constant.
- The sampling interval can be modified in FBs only.
- RS 97 is a system command: "System commands YES" must be specified on the programmer.
- Using OB13, you can generate a time base, which you can adapt to the relevant closed-loop control task.

### 3.2.3 Saving Scratch Flags

In many application programs, the cyclic program uses flags located in the scratch flag area (F200-F254).

The standard function blocks in the "S5-100U Closed-Loop Control" software package also use flags located in this area for buffering internal data. In such cases, you must save the scratch flag area by transferring it to a data block prior to invoking the time-controlled program and reload the flags from the data block when that program has terminated.

The "S5-100U Closed-Loop Control" software provides two FBs for this purpose:

- Save scratch flags : FB 217 (MERET)
- Load scratch flags : FB 218 (MELAD)

These FBs should be invoked in OB13. Before doing so, you must first generate and call a 38-word data block. The scratch flags are then transferred to this DB.

**Example:**

```
OB 13

C DB 10         Call relevant DB (e.g. DB 10)
JU FB 217       Save flags
NAME: MERET
JU FB 4         Call control program (e.g. FB 4)
NAME: REGLER
C DB 10         Call relevant DB (e.g. DB 10)
JU FB 218       Load flags
NAME: MELAD
BE
```

## 3.2.4 Time-Slice Distributor

Complex closed-loop control systems (cascade control systems, multiloop control systems) often comprise several controllers which affect the controlled system either in combination or separately.

Through use of function block FB 216 (ZEITVER), the "S5-100U Closed-Loop Control" software allows you to distribute time slices to the controllers so that not all of them are processed during a time interval.

Time-slice distribution is implemented by setting and resetting bits in flag or data words. The flag or data words are allocated via parameters S3S2 / S5S4. You can access the time-slice distribution for 2, 3, 4 or 5 controllers over these flag or data words. Time-slice distribution is dependent on the sampling interval, which you can set via the TA parameter. Time-slice distributor FB 216 must be invoked at the beginning of OB 13.

**Table 3-7. Time-Slice Distribution with FB 216 (ZEITVER)**

| No. of Con-trol-lers | Period in Sec. | Assigned Flag/Data Word for Parameter: | Time-Slice Distribution: | | | | |
|---|---|---|---|---|---|---|---|
| | | | 1st Con-troller | 2nd Con-troller | 3rd Con-troller | 4th Con-troller | 5th Con-troller |
| 2 | TA [s]·2 | S3S2 | Bit 1 | Bit 0 | | | |
| 3 | TA [s]·3 | | Bit 10 | Bit 9 | Bit 8 | | |
| 4 | TA [s]·4 | S5S4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| 5 | TA [s]·5 | | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 |

**Table 3-8. Calling and Initializing FB 216**

| Parameter | Description | Param. / Data | Assignment | STL |
|---|---|---|---|---|
| S3S2 | Time-slice distribution for 2 or 3 controllers | IW | DW 0 to 255;<br>FW 0 to 200:<br>KM = ”0” or ”1” | `: JU FB 216`<br>`NAME : START`<br>`S3S2 :`<br>`S5S4 :`<br>`TA :` |
| S5S4 | Time-slice distribution for 4 or 5 controllers | IW | DW 0 to 255;<br>FW 0 to 200:<br>KM = ”0” or ”1” | |
| TA | Sampling interval | DKF | KF = 10 to 1000 in increments of 10 (KF = 100 = 1s) | |

**Note:**

If you allocate data words to parameters S3S2 / S5S4, you must call the relevant data block (C DB ...) before invoking the time-slice distributor.

**Example:**

- Time-slice distribution for 3 controllers (FB 114, 115, 116)
- Sampling interval 1s   KF=100

```
FB 120

        NAME :ZEITVER

        000A      :C DB 2        CALL DATA BLOCK DB1
        000C      :JU FB 216     CALL TIME-SLICE DISTRIBUTOR
        000E NAME :START
        0010 S3S2 :DW  2
        0012 S5S4 :DW  4
        0014 TA   :KF + 100
        0016      :
        0018      :TB D 2.10     TEST BIT 10 IN DW 2 FOR "1"
        001A      :JC FB 104     CALL CONTROL FB 104
        001C NAME :REG1
        001E      :
        0020      :TB D 2.9      TEST BIT 9 IN DW 2 FOR "1"
        0022      :JC FB 105     CALL CONTROL FB 105
        0024 NAME :REG2
        0026      :
        0028      :TB D 2.8      TEST BIT 8 IN DW 2 FOR "1"
        002A      :JC FB 106     CALL CONTROL FB 106
        002C NAME :REG3
        002E      :BE
```

## 3.3 Example of a Start-Up Procedure

This section provides a simple STEP 5 start-up example, which includes a problem definition for an automatic control task and a complete solution to the problem described.

### 3.3.1 Problem Definition

- Single control loop
- Continuous controller (FB 201) with PI action
- PT 1 element (FB 208)
- Setpoint adjuster: For purposes of simplification, the setpoint is written to a data word in data block DBRE, thus making reading in of the setpoint via the analog input module and calling of standard function block FB 250 unnecessary.

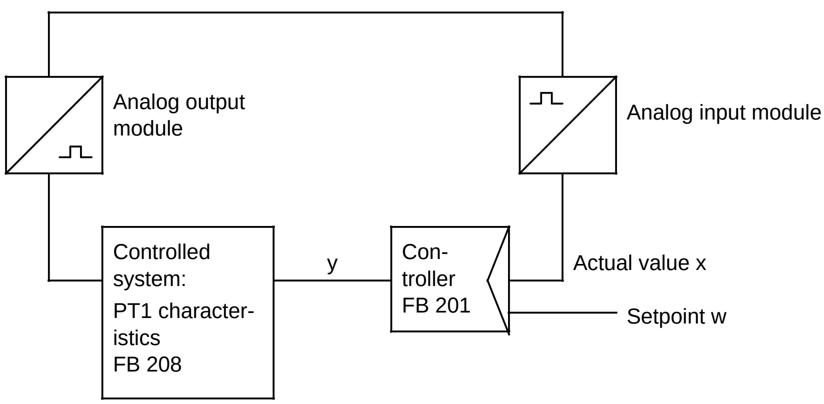



**Figure 3-4. Problem Definition: Single Control Loop**

The problem definition is such that the solution requires the following hardware:
- CPU 103 (6ES5 103-8MA02)
- Analog output module 2 x ± 10V (6ES5 470-8MA11)
- Analog input module 4 x ± 10V (6ES5 464-8MC11)
- PT1 element: Emulated by function block FB 208 from the software package

## 3.3.2 Solution

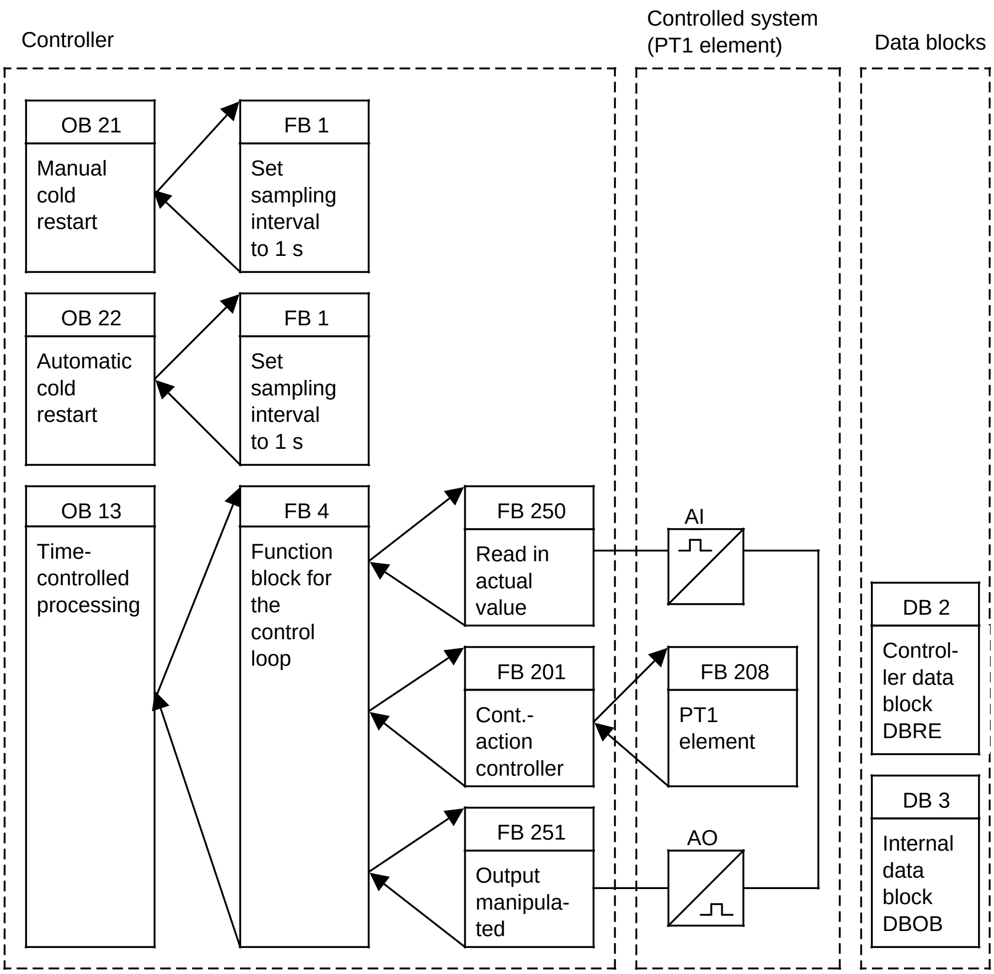



**Figure 3-5. Program Structure of the Control System**

The hardware prerequisites must be met before you can put the software into operation:

- Module slot assignments

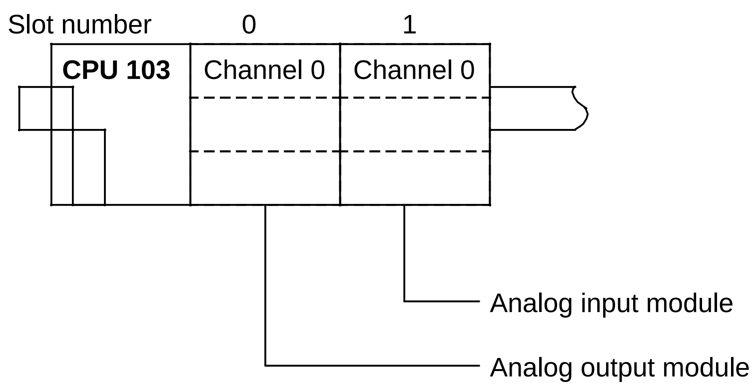
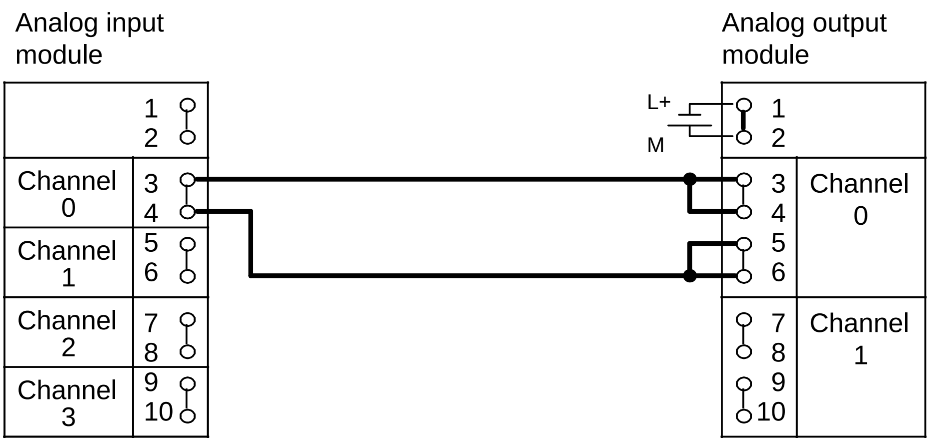
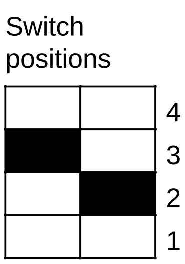



**Figure 3-6. Module Slot Assignments**

- Interfacing the PT1 element to analog modules




The analog input module is set for "1-channel" operation

**Figure 3-7. Interfacing the PT1 Element to Analog Modules**

The hardware prerequisites are now fulfilled, and the software can be put into operation:

- Transfer of the standard function blocks from the plugin submodule to PLC memory

- Generation and initialization of data blocks in PLC memory

  - Controller data block DBRE (DB 2)

```
DB 2

 0:    KF = +00000:              ACTUAL VALUE
 1:    KF = +00600:              SETPOINT
 2:    KF = +00000:
 3:    KF = +00000:              MANIPULATED VARIABLE
                                 (CONTROLLER OUTPUT)
 4:    KF = +00000:              SYSTEM DEVIATION
 5:    KM = 00000000 00000001:   CONTROL WORD (AUTO MODE)
 6:    KF = +01000:              GAIN (V=1)
 7:    KF = +02000:              PROPORTIONAL GAIN (P=2)
 8:    KF = +00350:              INTEGRATION TIME (TI=0.35S)
 9:    KF = +00000:              DERIVATIVE-ACTION TIME (TV=OS)
10:    KF = +00000:
11:    KF = +00000:
12:    KM = 00000000 00000000:   PAFE FOR CONTROLLER
13:    KF = +00000:
14:    KF = +00000:
15:    KF = +00000:              OUTPUT VALUE FROM PT1 ELEMENT
16:    KF = +00010:              T1=10 SECONDS
17:    KF = +00000:              INTERNAL DATA WORD FOR PT1
                                 ELEMENT
18:    KM = 00000000 00000000:
19:    KF = +00000:              PAFE FOR PT1 ELEMENT
20:
```

- Internal data block DBOB (DB 3)

```
DB 3

          0:     KF = +00000;
          1:     KF = +00000;
           :
           :
         53:     KF = +00000;
         54:
```

• Generating the control system function block "FB4" (REGLER)

```
FB 4

        NAME :REGLER

        000A      :C   DB   2         CALL DATA BLOCK 2
        000C      :JU FB 250          READ IN ANALOG VALUE
        000E NAME :RLG:AE
        0010 BG   :    KF +1          SLOT NUMBER FOR ANALOG INPUT MODULE
        0012 KNKT :    KY 0,6         CHANNEL NO., BIPOLAR FIXED-POINT NUMBER
        0014 OGR  :    KF +2047       UPPER LIMIT OF OUTPUT VALUE
        0016 UGR  :    KF -2047       LOWER LIMIT OF OUTPUT VALUE
        0018 EINZ :    F  100.0
        001A XA   :    DW   0         STORE ACTUAL VALUE IN DW0
        001C FB   :    F  100.1
        001E BU   :    F  100.2
        0020      :
```

```
0022      :JU FB 201               CALL CONTROLLER
0024 NAME :K-REG
0026 X    :    DW   0              ACTUAL VALUE
0028 W    :    DW   1              SETPOINT
002A YF   :    DW   2              CORRECTION
002C Y    :    DW   3              MANIPULATED VARIABLE
002E XD   :    DW   4              SYSTEM DEVIATION
0030 STEU :    DW   5              CONTROL WORD
0032 K    :    DW   6              GAIN
0034 P    :    DW   7              PROPORTIONAL GAIN
0036 TI   :    DW   8              INTEGRATION TIME
0038 TV   :    DW   9              DERIVATIVE-ACTION TIME
003A SEPD :    DW  10              SEPARATE D INPUT
003C Z    :    DW  11              FEEDFORWARD CONTROL
003E DBRE :    DB   2              CONTROLLER DATA BLOCK
0040 DBOB :    DB   3              INTERNAL DATA BLOCK
0042 PAFE :    DW  12              PARAMETER ASSIGNMENT ERROR
0044      :
0046      :C   DB   2              CALL DATA BLOCK 2
0048      :JU FB 208               DELAY CONTROLLER OUTPUT WITH
004A NAME :PT1                     PT1 ELEMENT TO EMULATE
004C EIN  :    DW   3              A CONTROLLED SYSTEM
004E AUS  :    DW  15
0050 T    :    DW  16
0052 INT1 :    DW  17
0054 PAFE :    DW  18              PARAMETER ASSIGNMENT ERROR
0056      :
0058      :JU FB 251               OUTPUT ANALOG VALUE
005A NAME :RLG:AA
005C XE   :    DW  15              READ IN MANIPULATED VARIABLE
005E BG   :    KF +0               SLOT NO. OF ANALOG OUTPUT MODULE
0060 KNKT :    KY 0,1              CHANNEL NUMBER, BIPOLAR
                                   FIXED-POINT NO.
0062 OGR  :    KF +2047            UPPER LIMIT OF OUTPUT VALUE
0064 UGR  :    KF -2047            LOWER LIMIT OF OUTPUT VALUE
0066 FEH  :    F  101.0
0068 BU   :    F  101.1
006A      :
006C      :BE
```

- Generate function block: Set 1 s sampling interval (FB 1: ABT.ZEIT)

```
FB 1

        NAME :ABT.ZEIT

        000A      :L   KF +100             CYCLE TIME = 100 * 10 MS
        000E      :T   RS  97              TRANSFER TO SYSTEM DATA WORD RS 97
        0010      :BE
```

- Generate organization blocks OB 21/22

```
OB 21
        0000      :JU FB 1
        0002 NAME :ABT.ZEIT
        0004      :BE


OB 22
        0000      :JU FB 1
        0002 NAME :ABT.ZEIT
        0004      :BE
```

- Generate organization block OB 13

```
OB 13
        0000      :JU FB 4                 CALL CONTROL SYSTEM
        0002 NAME :REGLER 1
        0004      :BE
```

You can display the parameters in data block DBRE and modify them as required with the programmer function "FORCE VAR".

```
OPERANDS:          SIGNAL STATES:

DB   2
DW   0              KF=+600
DW   1              KF=+600

DW   3              KF=+582
DW   4              KF=+0
DW   5              KM=10000000 00000001
DW   6              KF=+1000
DW   7              KF=+2000
DW   8              KF=+350
DW   9              KF=+0
DW  12              KM=00000000 00000000

DW  15              KF=+582
DW  16              KF=+10
DW  17              KF=+0
DW  18              KM=00000000 00000000
```

Please use the modified version of function block FB 4 (REGLER) shown below if you want to use the start-up example without analog modules:

```
FB 4

        NAME :REGLER

        000A      :
        000C      :JU FB 201                      CALL CONTROLLER
        000E NAME :K-REG
        0010 X    :    DW   0
        0012 W    :    DW   1
        0014 YF   :    DW   2
        0016 Y    :    DW   3
        0018 XD   :    DW   4
        001A STEU :    DW   5
        001C K    :    DW   6
        001E P    :    DW   7
        0020 TI   :    DW   8
        0022 TV   :    DW   9
        0024 SEPD :    DW  10
        0026 Z    :    DW  11
        0028 DBRE :    DB   2
        002A DBOB :    DB   3
        002C PAFE :    DW  12
        002E      :
        0030      :C   DB   2
        0032      :JU FB 208                      CALL PT1 ELEMENT
        0034 NAME :PT1
        0036 EIN  :    DW   3
        0038 AUS  :    DW   0
        003A T    :    DW  16
        003C INT1 :    DW  17
        003E PAFE :    DW  18
        0040      :
        0042      :
        0044      :BE
```

# 4 Description of the Standard FBs

## Figures



## Tables

**Tables (Cont.)**

# 4 Description of the Standard Function Bocks

## 4.1 Types of Controllers

### 4.1.1 Continuous-Action Controller - FB 201 - (K-REG)

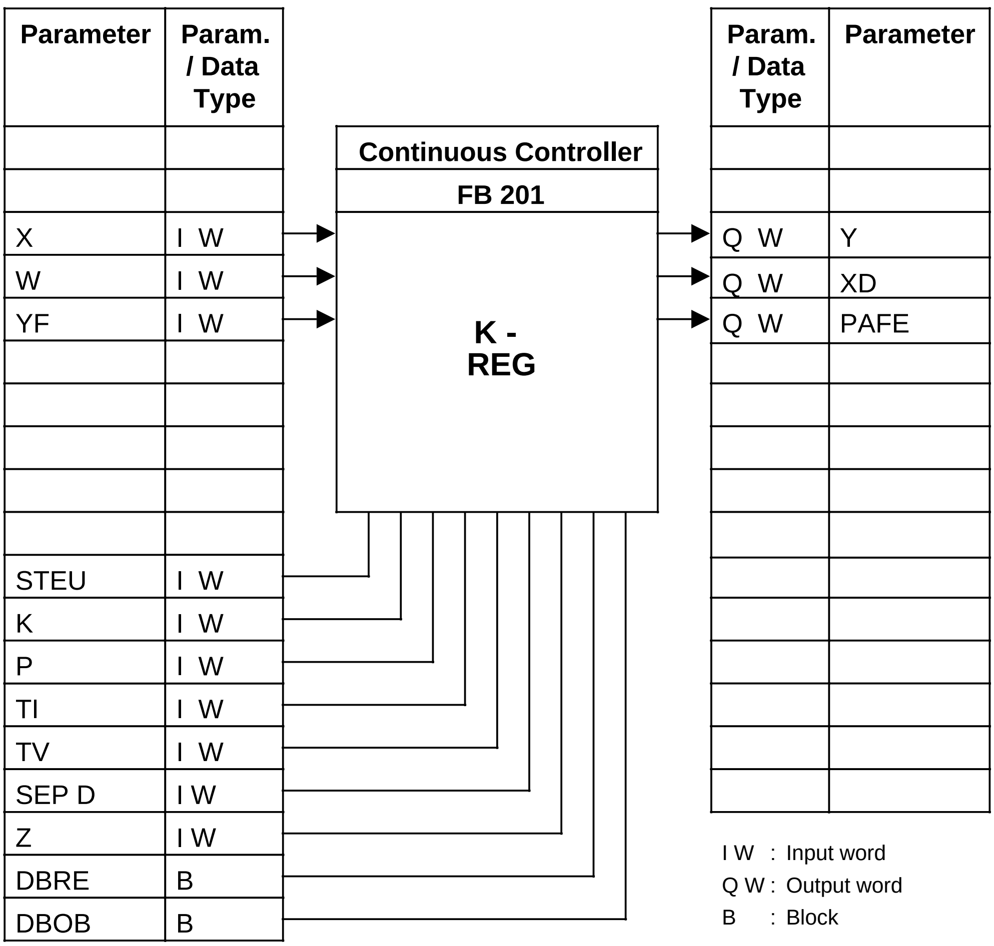



**Figure 4-1. Continuous-Action Controller -FB 201- (K-REG)**

A continuous-action controller can be used to implement a:

- P controller
- PI controller
- PD controller
- PID controller

The closed-loop control software package provides function block FB 201 for implementing a continuous-action controller. This function block can be used to simulate all of the controllers listed above. The figure below shows the structure of the continuous-action controller and how it is invoked and initialized.

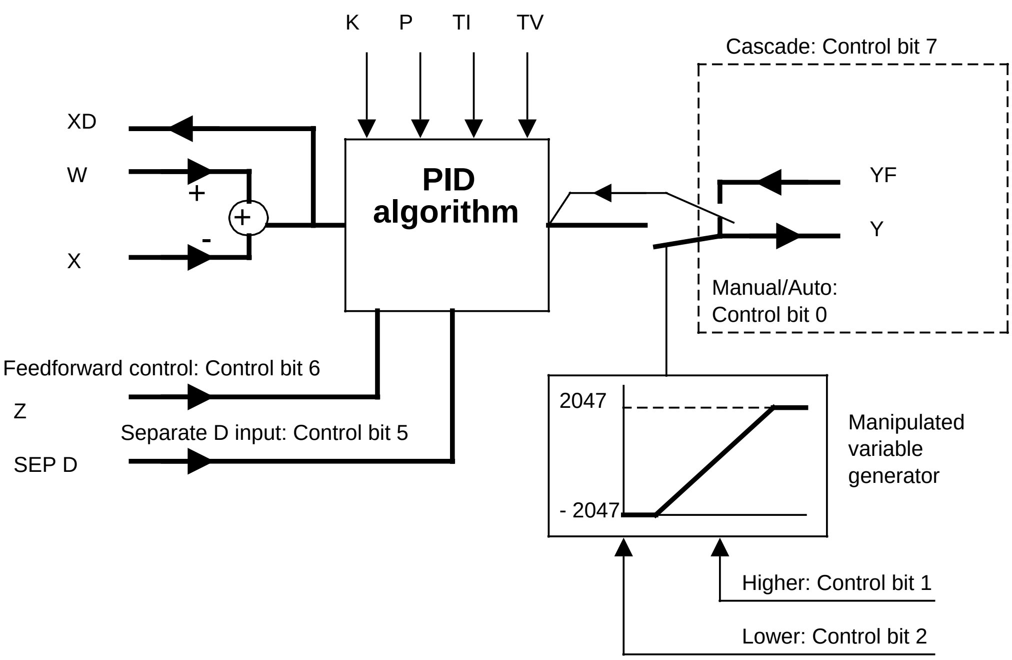



**Figure 4-2. Structure of the Continuous-Action Controller**

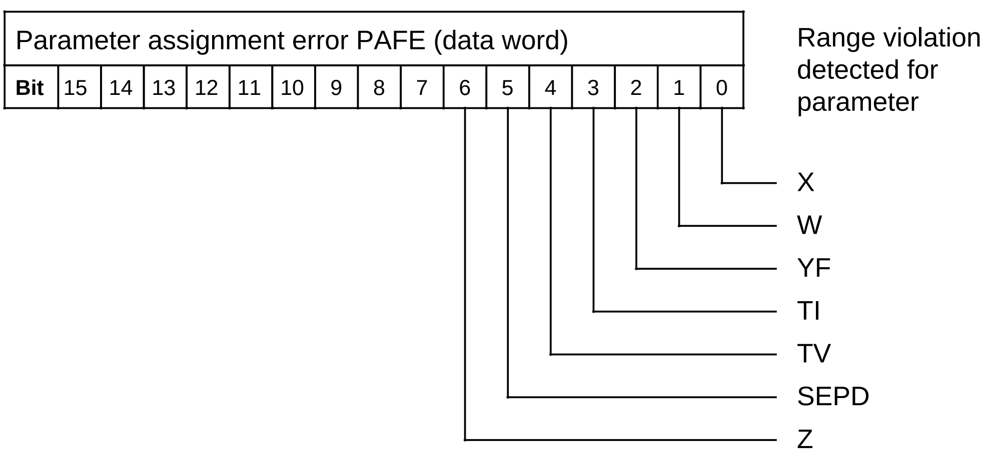
**Table 4-1. Calling and Initializing FB 201 (K-REG)**

| Param-eter | Description | Param. / Data Type | Assignment | STL |
|---|---|---|---|---|
| X | Actual value | I W | DW 0 to 255:<br>KF = - 2047 to 2047 | JU FB 201<br>NAME : K-REG<br>X :<br>W :<br>YF :<br>Y :<br>XD :<br>STEU :<br>K :<br>P :<br>TI :<br>TV :<br>SEPD :<br>Z :<br>DBRE :<br>DBOB :<br>PAFE : |
| W | Setpoint | I W | DW 0 to 255:<br>KF=- 2047 to 2047 | |
| YF | Correction only when used as master controller | I W | DW 0 to 255:<br>KF=- 2047 to 2047 | |
| Y | Value of the manipulated variable | Q W | DW 0 to 255:<br>KF=- 2047 to 2047 | |
| XD | System deviation | Q W | DW 0 to 255:<br>KF=- 4094 to 4094 | |
| STEU | Control word | I W | DW 0 to 255:<br>See Control word | |
| K | Gain | I W | DW 0 to 255:<br>KF=-32768<br>to +32767 | |
| P | Proportional gain * | I W | DW 0 to 255:<br>KF=-32768 to+32767 | |
| TI | Integration time<br>TI = 1000 / TN<br>TN = Integral-action time | I W | DW 0 to 255:<br>KF=0 to 9999 SEC<br>Factor 0.001 | |
| TV | Derivative-action time | I W | DW 0 to 255;<br>KF=0 to 999 SEC | |
| SEP D | Separate D input | I W | DW 0 to 255:<br>KF=0 to 999 SEC | |
| Z | Feedforward control | I W | DW 0 to 255:<br>KF=- 2047 to 2047 | |

* Implementation of D and I controllers
Default value: KF = 1000

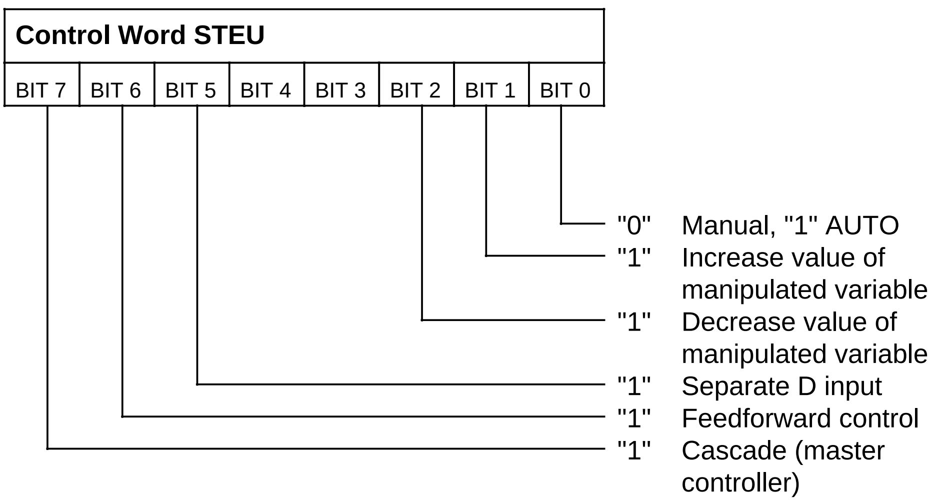
**Table 4-1. Calling and Initializing FB 201 (Cont.)**

| Param-eter | Description | Param. / Data Type | Assignment | STL |
|---|---|---|---|---|
| DBRE | Data block containing the controller parameters | B | DB 2 to 255 | |
| DBOB | Controller's internal data block | B | DB 2 to 255 | |
| PAFE | Parameter assignment error: Parameter range violation | Q W | DW 0 to 255<br>KM="0" or "1"<br>KM="0" or "1"<br>KM="1" indicates range violation | |



The parameters for FB 201 are in data words of data block DBRE. You must enter the actual parameter values in these data words. The controller uses data block DBOB (which must comprise at least 54 data words) as internal data block.

You can define the FB 201 options in the low-order byte (bits 0 - 7) of the control word (STEU parameter).




**Manual/AUTO: Bit 0**

The controller is in manual mode (i.e. nonautomatic operation) when bit 0 = 0. The manipulated variable can be set to an arbitrary value (with T DW). The specified value of the manipulated variable is retained until the controller is switched to AUTO mode (bit 0 = 1). The controller then takes over the value bumplessly when setpoint = actual value.

When you set bit 0 to 1, the controller is in AUTO mode. The value of the manipulated variable is then computed in accordance with the PID algorithm and subsequently output.

**Increase the value of the manipulated variable: Bit 1**

This option serves a practical purpose only for start-up or testing.
When you set bit 1 to 1, you have initialized the manipulated variable generator for the "Increase manipulated variable" option. This function is effective only when you have set bit 0 to 0 (manual mode).

The value input manually is then increased by KF = 41 / SEC (approx. 2 % / SEC) if you set the cycle time to one second. If the cycle time is set to a value other than one second, the value is increased according to the formula:

KF = 41 /Cycle time

**Decrease the value of the manipulated variable: Bit 2**

This option serves a practical purpose only for start-up or testing.
The value entered manually is decreased by KF=41 / SEC (approx. 2 % / SEC). In all other respects, the same applies as for the "Increase the value of the manipulated variable" option (bit 1).

**Separate D input: Bit 5**

A separate D input (SEP D parameter) is made available when you set bit 5 to 1.
A separate D input is used, for example, when only the actual value (with inverted sign) is applied to the D input.

**Feedforward control: Bit 6**

The FB's Z parameter is activated when you set bit 6 to 1. Feedforward control affects the controller's output.
If the disturbance is measurable, feedforward control can be used to shorten the controller's response time.

**Cascade: Bit 7**

You can set bit 7 to 1 when you want to use the continuous-action controller as master controller in a cascade control system. In this case, an additional input ("YF") is activated in FB 201.

In a normal control system, the value of the manipulated variable is taken over as initial value following the transfer to "AUTO" mode. The "YF" parameter plays no role in this case. This is different in a cascade control system, where two controllers are linked in such a way that the slave controller uses the master controller's manipulated variable as setpoint. In this case, it is assumed that switching of the cascade from "Manual" to "AUTO" is bumpless.

To implement cascade control, you must establish the following link between the two controllers in data block DBRE:

YF (master controller)= X (slave controller)

The system deviation for the slave controller is thus zero.

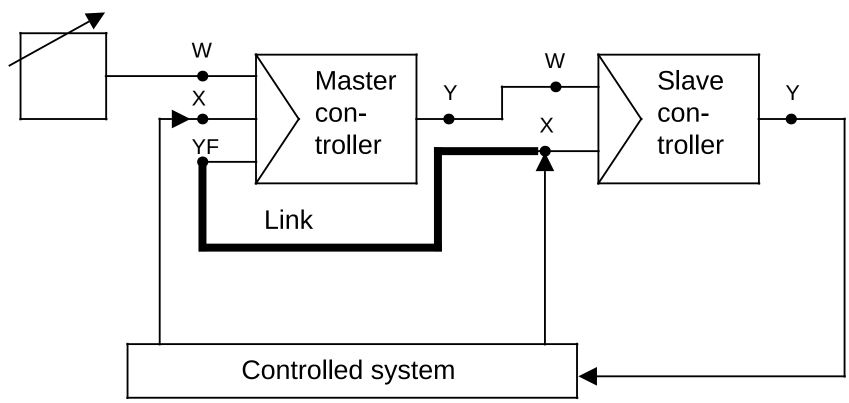



**Figure 4-3. Bumpless Transfer of a Cascade from ”Manual” to ”AUTO”**

**Note:**
If the continuous-action controller is switched from manual to automatic mode, the system deviation must be such that there is no setpoint step-change in the slave controller during transfer.

**Changing the sampling interval**

The continuous-action controller (FB 201) is preset for a sampling interval of one second.
If you have chosen a different sampling interval (TA), you must convert control parameters TI (integration time) and TV (derivative-action time) to accord with the specified sampling interval in order to obtain the actual values.

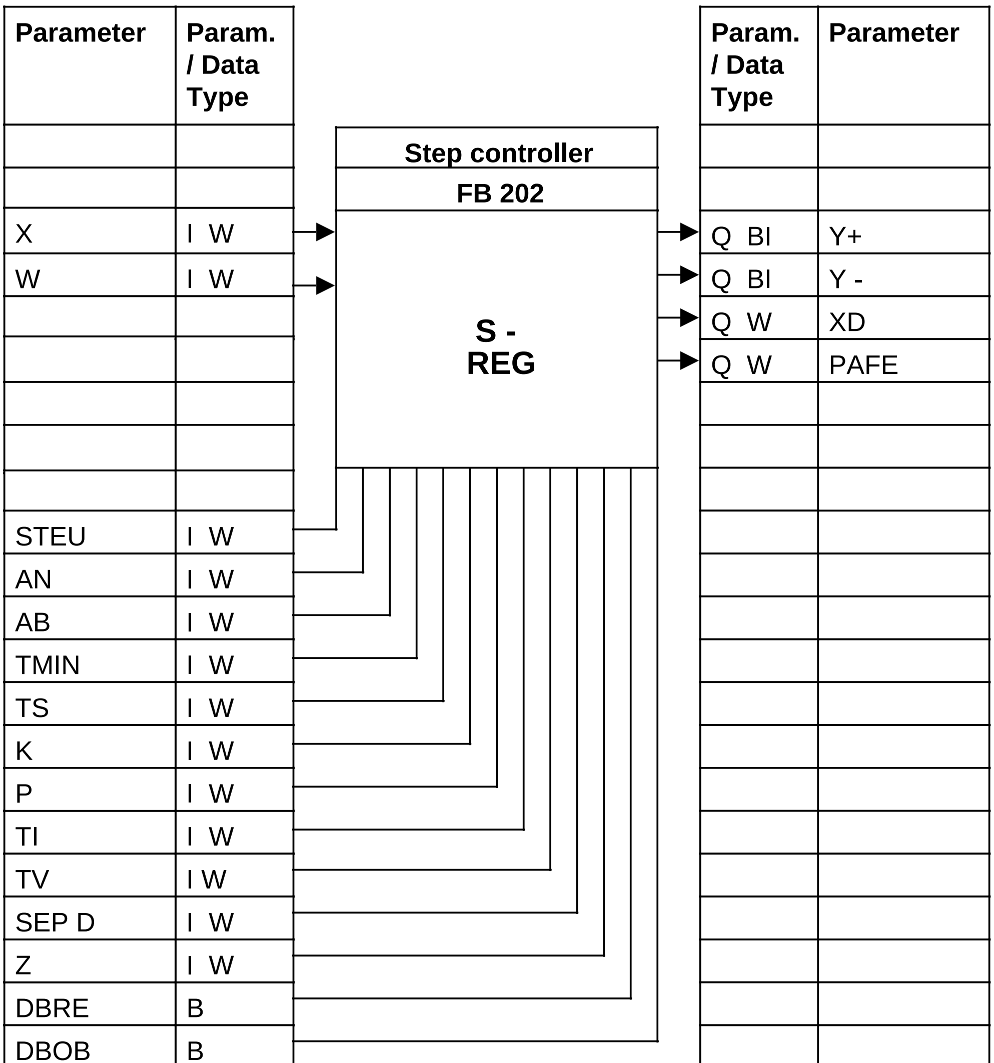
Integration time $TI = TI_{(1\,s)} \cdot TA$

Derivative-action time $TV = \frac{TV_{(1\,s)}}{TA}$

**Example:**

When sampling interval TA is 1 second, TI is set to six seconds and TV to six seconds. If sampling interval TA is increased to two seconds, TI must be set to 12 seconds and TV to three seconds in order to obtain approximately the same control response.

## 4.1.2 Step-Action Controller -FB 202- (S-REG)




I W : Input word
Q W : Output word
Q BI : Output bit
B : Block

**Figure 4-4. Step-Action Controller -FB 202- (S-REG)**

Step-action controllers provide three operating positions:

FORWARD - STOP - REVERSE

Only integral-action final control elements can be used in conjunction with a step-action controller. This means that all "Forward" and "Reverse" pulses are summated (e.g. screw). In the stable state, no pulses are output to the final control element. The final control element retains its steady-state condition.

The figure below shows the structure of the step-action controller (function block FB 202), and how it is invoked and initialized.

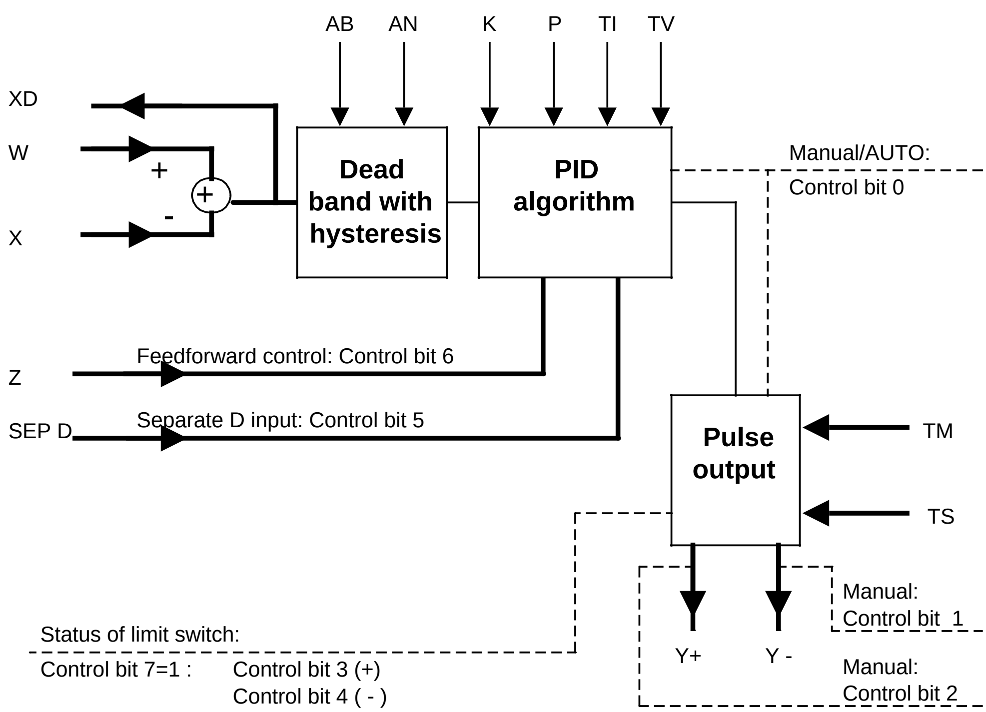



**Figure 4-5. Structure of the Step-Action Controller (S-REG)**

**Note:**

The step-action controller (FB 202) can function properly only when function block FB 214 (Dead band with hysteresis) is in PLC memory.

**Table 4-2. Calling and Initializing FB 202 (S-REG)**

<table>
<tr><th>Param-eter</th><th>Description</th><th>Param. / Data Type</th><th>Assignment</th><th>STL</th></tr>
<tr><td>X</td><td>Actual value</td><td>I W</td><td>DW 1 to 255:<br>KF = - 2047 to 2047</td><td rowspan="9"><pre>
       JU FB 202
NAME : K-REG
X    :
W    :
Y+   :
Y-   :
XD   :
STEU :
AN   :
AB   :
TMIN :
TS   :
K    :
P    :
TI   :
TV   :
SEPD :
Z    :
DBRE :
DBOB :
PAFE :
</pre></td></tr>
<tr><td>W</td><td>Setpoint</td><td>I W</td><td>DW 1 to 255:<br>KF=- 2047 to 2047</td></tr>
<tr><td>Y+</td><td>Actuator "FORWARD"</td><td>Q BI</td><td>"0" or "1":<br>When "1", positive actuating pulse is output</td></tr>
<tr><td>Y -</td><td>Actuator "REVERSE"</td><td>Q BI</td><td>"0" or "1":<br>When "1", negative actuating pulse is output</td></tr>
<tr><td>XD</td><td>System deviation</td><td>Q W</td><td>DW 0 to 255:<br>KF=-4094 to 4094</td></tr>
<tr><td>STEU</td><td>Control word</td><td>I W</td><td>DW 0 to 255:<br>See Control word</td></tr>
<tr><td>AN</td><td>Upp. resp. thresh. of hysteresis<br>AN > AB</td><td>I W</td><td>DW 0 to 255:<br>KF=0 to 2047</td></tr>
<tr><td>AB</td><td>Low. resp. thresh. of hysteresis<br>AN > AB</td><td>I W</td><td>DW 0 to 255:<br>KF=0 to 2047</td></tr>
<tr><td>TMIN</td><td>Minimum duration of the actuating pulse</td><td>I W</td><td>DW 0 to 255:<br>KF 1 in SEC.</td></tr>
</table>

**Table 4-2. Calling and Initializing FB 202 (Cont.)**

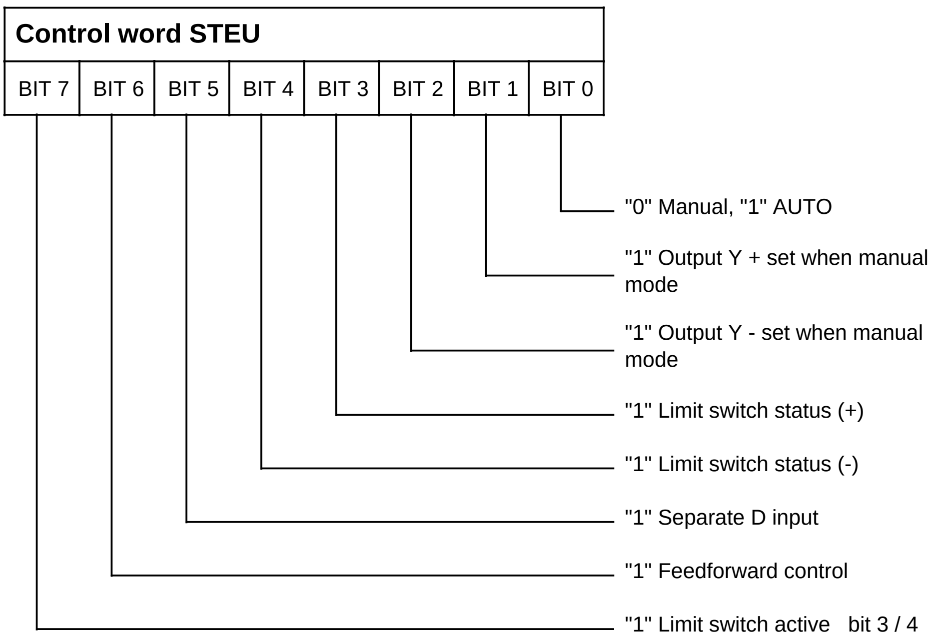
| Param-eter | Description | Param. / Data Type | Assignment | STL |
|---|---|---|---|---|
| TS | Actuating time: Period until input step to end stop... | I W | DW 0 to 255: KF 1 in SEC. | |
| K | Gain | Q B | DW 0 to 255: KF=-32768 to 32767 | |
| P * | Proportional gain | I W | DW 0 to 255: KF=-32768 to 32767 | |
| TI | Integration time TI=10000 / TN TN: Integral-action time | I W | DW 0 to 255: KF=0 to 9999 SEC Factor 0.001 | |
| TV | Derivative-action time | I W | DW 0 to 255: KF=0 to 999 SEC | |
| SEP D | Separate D input | I W | DW 0 to 255: KF=- 2047 to 2047 | |
| Z | Feedforward control | I W | DW 0 to 255: KF=- 2047 to 2047 | |
| DBRE | Data block containing control parameters | B | DB 2 to 255 | |
| DBOB | Controller's internal data block | B | DB 2 to 255 | |
| PAFE | Parameter assignment error: Parameter range violation | Q W | DW 0 to 255 KM="0" or "1" KM="1" indicates range violation | |

* Implementation of D or I controllers:
Default value: KF = 1000

| Parameter assignment error PAFE (data word) | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Range violation detected for parameter | | | | | | | Z | SEPD | TV | TI | TS | TMIN | AB | AN | W | X |

FB 202 is initialized via data words in data block DBRE. You must enter the actual parameter values in these data words. The control loop uses DBOB as internal data block.

The step-action controller also provides a number of options, which are defined in the low-order byte (bits 0 - 7) of the control word (STEU parameter).




**Manual/AUTO: Bit 0**
If the controller is in "Manual mode" (i.e. if bit 0 = 0), you can control outputs Y + and Y - independently of the control algorithm.
In AUTO mode (i.e. when bit 0 = 1), the actuating pulse is computed and output by the controller.

**Set output Y+ / Y -: Bits 1 / 2**
See Manual/AUTO: Bit 0

**Limit switch status (±) : Bits 3 / 4**
If your control system has limit switches that issue status signals, these signals must be entered in bits 3 / 4 of the control word. Prerequisite is that you have set bit 7 (i.e. limit switches active).

Bit 3 of the control word: Status signal for position (+) reached.
Bit 4 of the control word: Status signal for position ( - ) reached.

These status bits disable the controller's integral action.

**Separate D input, feedforward control: Bit 5 / 6**
These options are identical to those for the continuous controller.
4.1.1 (continuous-action controller, separate D input, feedforward control: Control bit 5 / 6).

**Limit switches active: Bit 7**
The "Limit switches active" function is enabled when you set this bit to "1". See "Limit Switch Status".
Limit switches available: Bit 7 = 1
No limit switches available: Bit 7 = 0

**Changing the sampling interval**
The step-action controller (FB 202) is preset to a sampling interval of one second.
If you have chosen another sampling interval, you must convert control parameters TM (minimum pulse duration) and TS (actuating time) to accord with the sampling interval selected in order to obtain the actual values.

| | | |
|---|---|---|
| Minimum pulse duration | $TM_{(\neq 1s)}$ | $= TM_{(1s)} \cdot TA_{(\neq 1s)}$ |
| Actuating time | $TS_{(\neq 1s)}$ | $= TS_{(1s)} \cdot TA_{(\neq 1s)}$ |

**Example:**

The sampling time remains set at 100 ms in OB 13. TM = 100 s when TA = 1 s.
$TM_{(\neq 1)} = 100\ s \cdot 0.1\ s = 10\ s$ (actual value).

**Dead band with hysteresis -FB 214- (HYSTERES)**

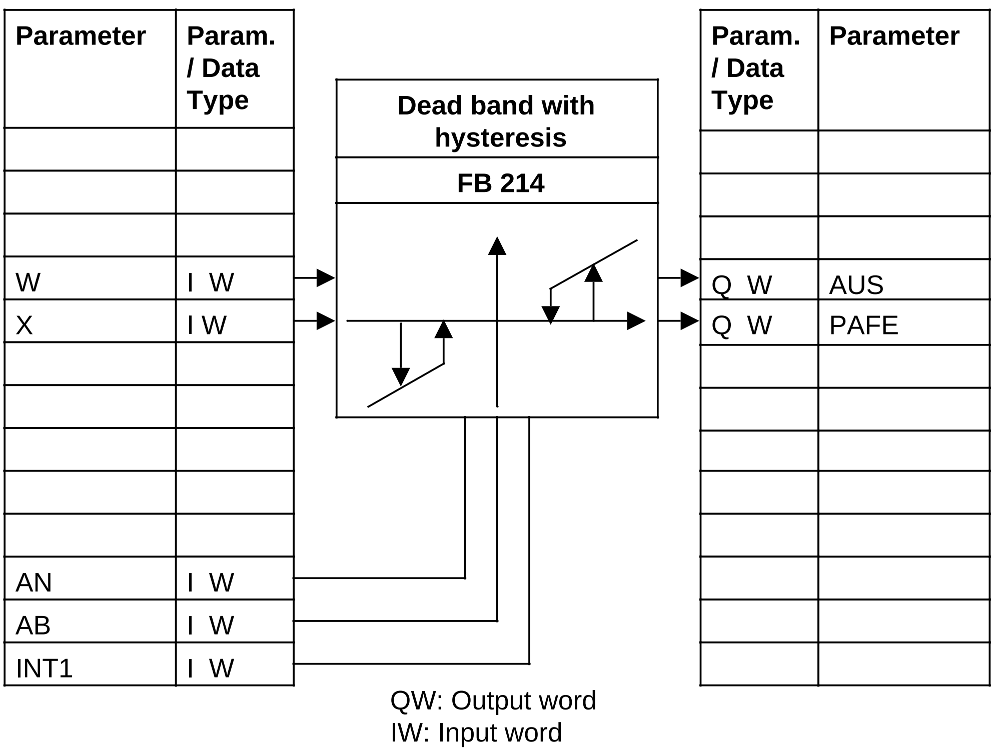



**Figure 4-6. Dead Band with Hysteresis -FB 214- (HYSTERES)**

To avoid excessive switching, it is usual to provide final control elements with a hysteresis or differential gap. The hysteresis prevents minor changes in controlled variable X around setpoint W (i.e. changes smaller than the differential gap) from being forwarded to the output. If the system deviation exceeds the differential gap, the input value is forwarded without change to the output. The hysteresis is an integral part of the step-action controller. Function block FB 214, however, lets you use the hysteresis independently of the step-action controller.

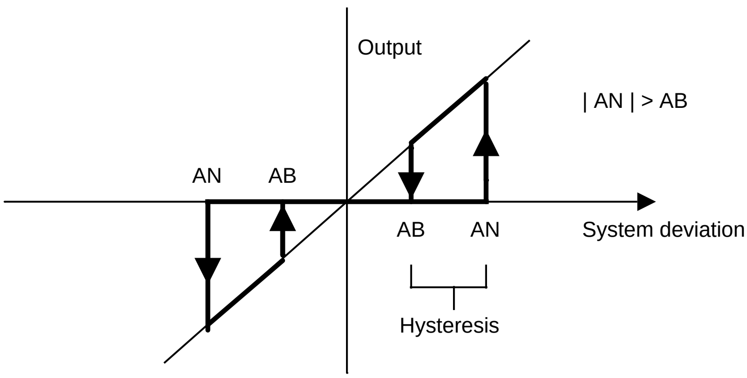



**Figure 4-7. Characteristic of a Step-Action Controller with Hysteresis**

**Table 4-3. Calling and Initializing the Hysteresis FB 214 (HYSTERES)**

| Parameter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| X | Setpoint | I W | DW 0 to 255: KF = -2047 to 2047 | JU FB 214<br>NAME : HYSTERES |
| W | Actual value | I W | DW 0 to 255: KF= -2047 to 2047 | W : |
| AUS | Output | Q W | DW 0 to 255: KF= -2047 to 2047 | X : |
| AN | Upp. resp. thresh. of hysteresis | I W | DW 0 to 255: KF= 0 to 2047 | AUS : |
| AB | Low. resp. thresh. of hysteresis | I W | DW 0 to 255: KF= 0 to 2047 | AN : |
| INT 1 | Internal data word | I W | DW 0 to 255: No initialization necessary; data word need only be assigned | AB :<br>INT 1:<br>PAFE : |
| PAFE | Parameter assignment error: Parameter range violation | Q W | DW 0 to 255 KM="0" or "1" KM="1" indicates range violation | |

| **Parameter assignment error PAFE (data word)** | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

- Bit 0: W
- Bit 1: X
- Bit 2: AN
- Bit 3: AB

The parameters for function block FB 214 are in data words of a data block. You must enter the actual parameter values in these data words and you must generate and call the relevant data block (which is freely selectable) yourself.

### 4.1.3 Pulse Controller -FB 203- (I-REG)

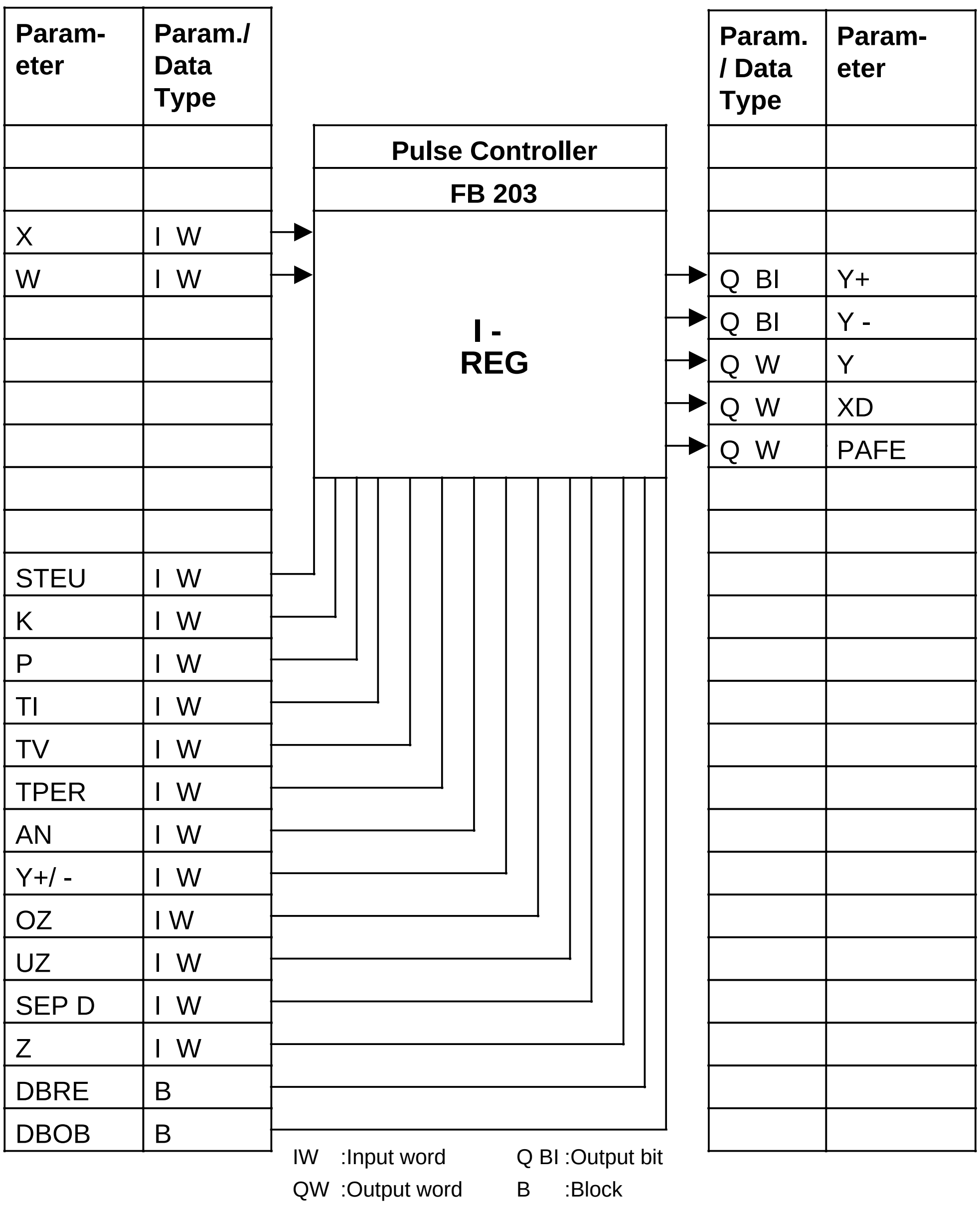



**Figure 4-8. Pulse Controller -FB 203- (I-REG)**

Function block FB 203 implements a pulse controller. The pulse controller can be used as as two-step (heating only) or three-step (heating and cooling) controller.

Structure, start-up and initialization of the pulse controller are illustrated below:

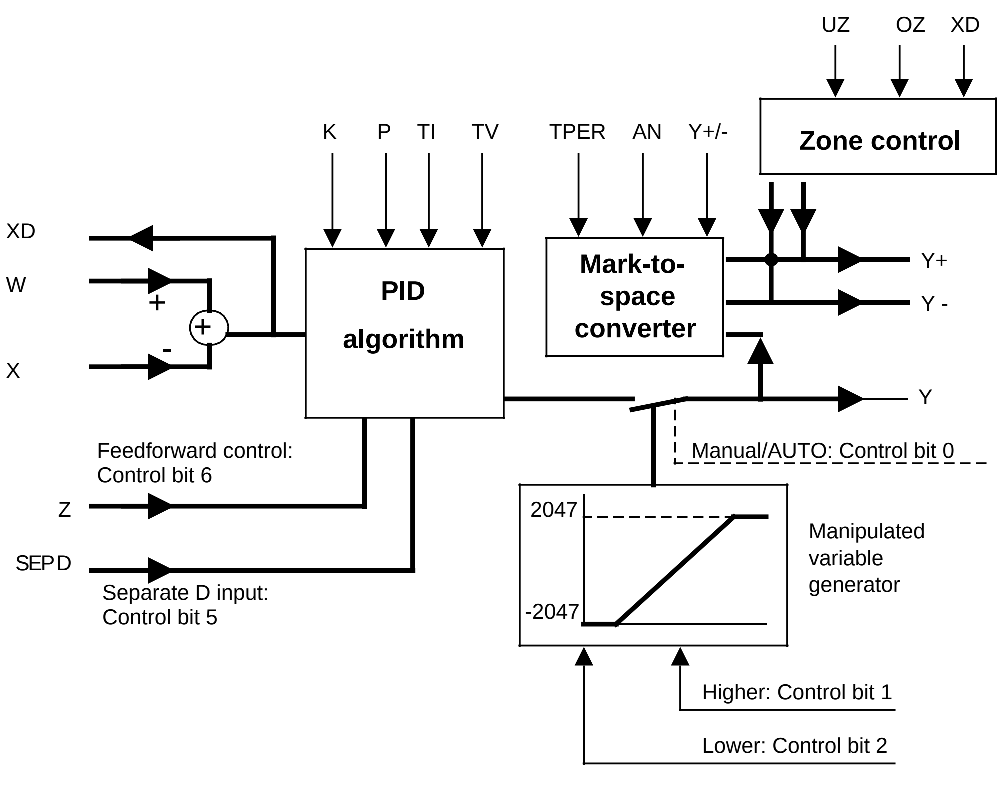



**Figure 4-9. Structure of the Pulse Controller -FB 203- (I-REG)**

**Table 4-4. Calling and Initializing the Pulse Controller FB 203 (I-REG)**

| Parameter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| X | Actual value | I W | DW 0 to 255:<br>KF = -2047 to 2047 | JU FB 203<br>NAME : IMPULS<br>X :<br>W :<br>Y+ :<br>Y- :<br>Y :<br>XD :<br>STEU :<br>K :<br>P :<br>TI :<br>TV :<br>TPER :<br>AN :<br>Y+/- :<br>OZ :<br>UZ :<br>SEPD :<br>Z :<br>DBRE :<br>DBOB :<br>PAFE : |
| W | Setpoint | I W | DW 0 to 255:<br>KF= -2047 to 2047 | |
| Y+ | Final control element for "HEATING" | Q BI | "0" or "1":<br>Heating pulse is generated if "1" | |
| Y - | Final control element for "COOLING" | Q BI | "0" or "1":<br>Cooling pulse is generated if "1" | |
| Y | Output manipulated variable additionally as analog signal | Q W | DW 0 to 255:<br>KF= 0 to 1023 | |
| XD | System deviation | Q W | DW 0 to 255:<br>KF= -4094 to 4094 | |
| STEU | Control word | I W | DW 0 to 255:<br>See Control word | |
| K | Gain | I W | DW 0 to 255:<br>KF= -32768 to 32767<br>Factor 0.001 | |
| P | Proportional gain * | I W | DW 0 to 255:<br>KF= -32768 to 32767 | |
| TI | Integration time<br>TI = 1000 / TN<br>TN: Integral-action time | I W | DW 0 to 255:<br>KF= 0 to 9999 SEC<br>Factor 0.001 | |

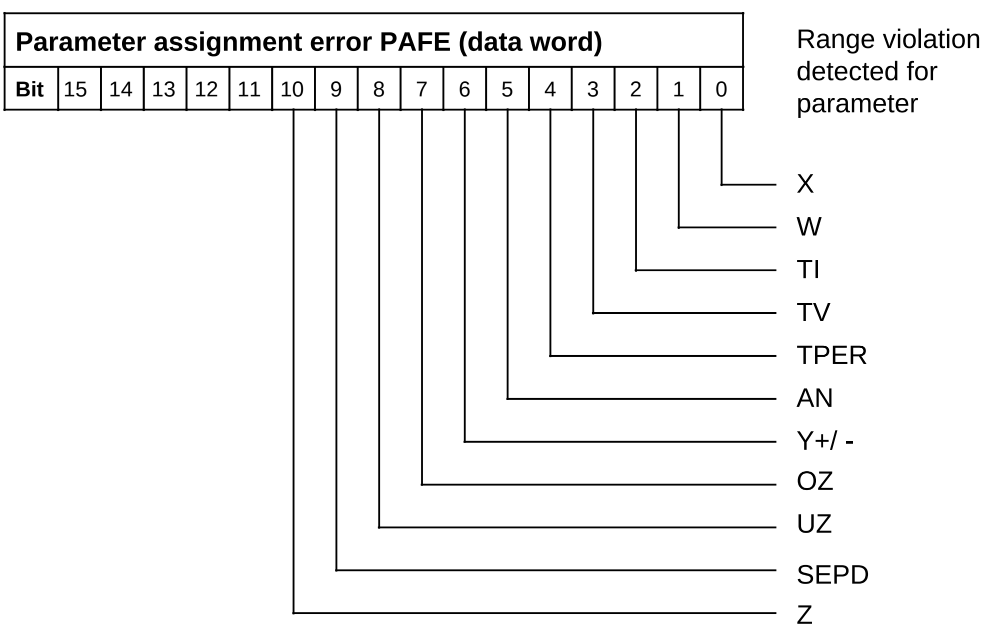
* Implementation of D and I controllers
Default value: KF = 1000

**Table 4-4. Calling and Initializing the Pulse Controller FB 203 (Cont.)**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| TV | Derivative-action time | I B | DW 0 to 255:<br>KF= 0 to 999 SEC | |
| TPER | Period: Sum of the duration of pulse and interpulse period | I W | DW 0 to 255:<br>KF 1 SEC | |
| AN | Response threshold of the pulse | I W | DW 0 to 255:<br>KF= 0 to 1023 | |
| Y+/- | Ratio between HEATING and COOLING | I W | DW 0 to 255:<br>KF 1, where<br>KF =32 corresponds to HEKU=1=100 %<br>KF=16: Double heating<br>KF=64: Double cooling | |
| OZ | Zone control: Upper control zone limit | I W | DW 0 to 255:<br>KF=0 to 2047 | |
| UZ | Zone control: Lower control zone limit | I W | DW 0 to 255:<br>KF=0 to 2047 | |
| SEP D | Separate D input | I W | DW 0 to 255:<br>KF=-2047 to 2047 | |
| Z | Feedforward control | I W | DW 0 to 255:<br>KF=-2047 to 2047 | |

**Table 4-4. Calling and Initializing Pulse Controller FB 203 (Cont.)**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| DBRE | Data block containing the control parameters | B | DB 2 to 255 | |
| DBOB | Controller's internal data block | B | DB 2 to 255 | |
| PAFE | Parameter assignment errors: Parameter range violations | Q W | DW 0 to 255<br>KM="0" or "1"<br>KM="1" indicates range violation | |



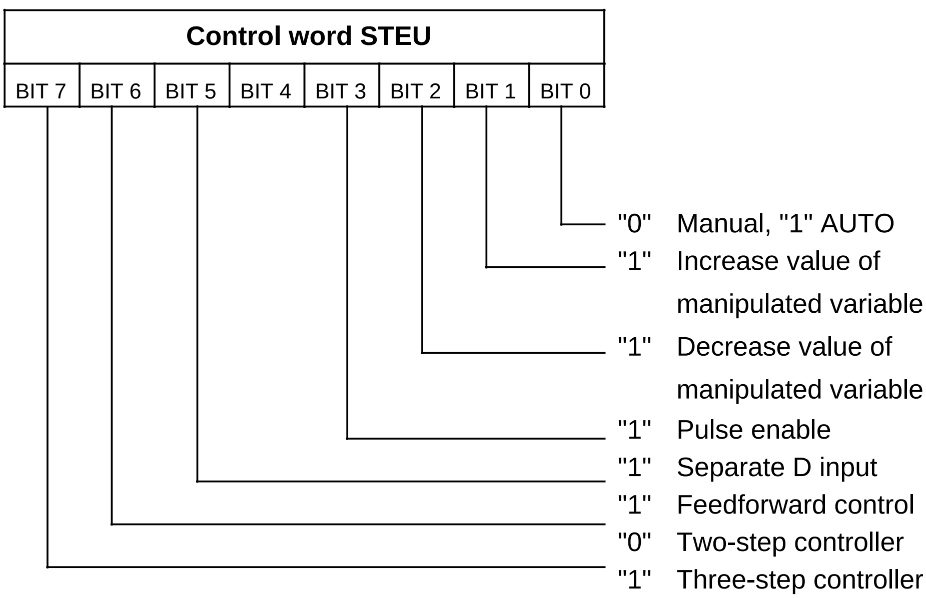
**Additional information on initialization of the pulse controller**

| | | | |
|---|---|---|---|
| TPER | | = | Time constant / 100 |
| TA | | = | TPER / 20 |
| Accuracy | | = | ST·TPER / 4 |
| | | | |
| TN | | = | (TU+TPER / 2+12 / ST)·1.665 |
| TI | | = | 100 / TN für TA=1s |
| TV | | = | (TU+TPER / 2+12 / ST)·0.6 |
| K | | = | 920000 / (ST·(TU+TPER / 2)+6) |
| UZ | = OZ | = | ST·(TU+TPER / 2)+3 |
| P | | = | 1000 |

TPER : Period in seconds
TA : Sampling intervals
ST : Slope of the curve in KF / sec.
(relates to scaled range KF = -2047...2047)
TI : Integration time
TN : Integral-action time
TU : Dead time in seconds
TV : Derivative-action time
K : Gain
UZ : Lower response threshold
OZ : Upper response threshold
P : Proportional gain

The parameters for FB 203 are in data words of data block DBRE. You must enter the actual parameter values in these data words. The control loop uses DBOB as internal data block.

The pulse controller options are also defined in the low-order byte (bits 0 to 7) of the control word (STEU parameter).




Bits 0, 1, 5 and 6 are identical with those for the continuous-action controller options ( 4.1.1). A detailed description is thus unnecessary at this point, as the pulse controller is nothing other than a continuous-action controller with pulse output.

**Two-step/Three-step controller: Bit 7**

The control structure for a two-step controller is activated when bit 7 is set to 1, and that for a three-step controller when bit 7 is set to 0.

- **Two-step controller**

If you set the AN parameter to zero, the value of the manipulated variable is converted as follows:

KF = 0 The heater is always off.
KF = 512 The heater is switched on for a half- period.
KF = 1023 The heater is always on.

If you set the AN parameter to a specific value, however, no actuating pulse is output as long as the value of the manipulated variable is lower than the specified value.
No interpulse space is output, even when the value of the manipulated variable exceeds 100 % of the response threshold.
The AN parameter can be set so as to avoid excessively short pulses and interpulse spacing.

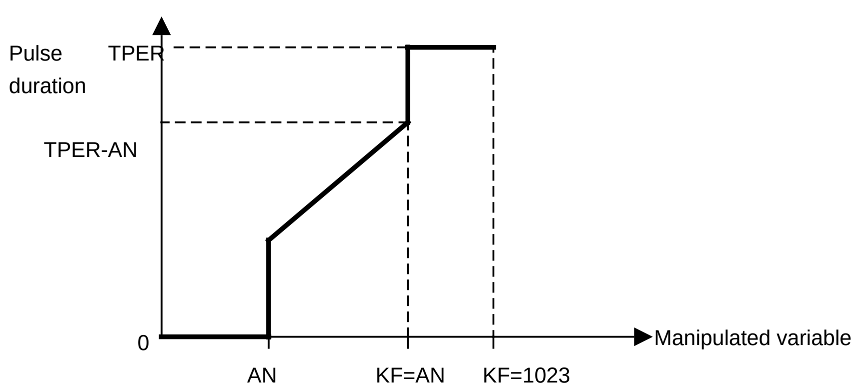



**Figure 4-10. Two-Step Controller: Pulse Duration as a Function of the Manipulated Variable**

- **Three-step controller**

If the three-step controller is initialized, the value of the manipulated variable is converted as follows when the response threshold is zero:

| | |
|---|---|
| KF = 0 | The heater is always off. The cooling unit is always on. |
| KF = 512 | Heater and cooling unit are off. |
| KF = 1023 | The heater is always on. |

The value range KF = 0 to 512 is for cooling, the value range KF = 512 to 1023 for heating.
When the response threshold is not zero, conversion is as shown in Figure 4-10.

"Heating" and "cooling" are always separate. This means that the cooling unit is never on when the heater is on.
FB parameter Y +/- allows you to change the heating-to-cooling ratio.

Y+/- KF = 32 : $\frac{\text{Heating}}{\text{Cooling}} = 1$

Y+/- KF = 16 : $\frac{\text{Heating}}{\text{Cooling}} = 0.5$

Y+/- KF = 64 : $\frac{\text{Heating}}{\text{Cooling}} = 2$

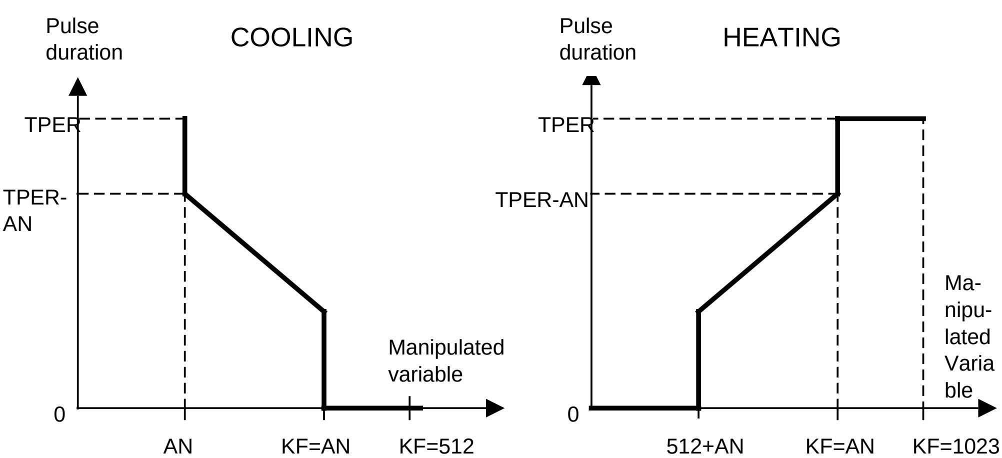



**Figure 4-11. Three-Step Controller: Pulse Duration as a Function of the Manipulated Variable**

- **Zone control**

One of the most important factors as regards a pulse controller is that it must not be in operation when the system deviation exceeds specific values, otherwise the controller's I component assumes excessively high values which greatly impair the control quality. You can set the appropriate limitations by initializing control parameters OZ (upper zone) and UZ (lower zone).

A two-step/three-step controller responds as follows when the temperature rises to such a degree that the system deviation exceeds the upper zone:

- Two-step controller: The heater is permanently switched off (no pulse is output).
- Three-step controller: The cooling unit is permanently switched on (continuous pulse).

If the temperature drops so much that the system deviation falls below the lower zone, the controller responds as follows:

- Two-step controller: The heater is permanently switched on (continuous pulse).
- Three-step controller: The cooling unit is permanently switched off (no pulse is output).

If the system deviation is within the specified zones, the manipulated variable is computed in accordance with the PID algorithm.

**Changing the sampling interval**

The pulse controller (FB 203) is preset to a 1-second sampling interval.
If you have chosen another sampling interval (TA), you must convert control parameters TI (integration time), TV (derivative-action time) and TPER (time period) to accord with the specified sampling time in order to obtain the actual values.

| | | | |
|---|---|---|---|
| Integration time | $TI_{(1\ s)}$ | = | $\frac{TI_{(1\ s)}}{TA_{(1 s)}}$ |
| Derivative-action time | $TV_{(1)s}$ | = | $\frac{TV_{(1\ s)}}{TA_{(1\ s)}}$ |
| Time period | $TPER_{(1\ s)}$ | = | $TPER_{(1\ s)} \times TA_{(1\ s)}$ |

**Example:**

The sampling time remains set to 100 ms.
When TA=1 second, TPER is=100 seconds

$$TPER_{(1\ s)}=100\ s \cdot 0.1=10\ s\ \text{(actual value)}$$

If the sampling interval is 100 ms, time period TPER must be set to 10 seconds.

**Note:**
The control quality depends on the sampling interval. The smaller the sampling interval (e.g. TA=100 ms), the better the quality.

## 4.2 Auxiliary Controller Functions

The function blocks described in the preceding sections are for a standard configuration. The auxiliary functions discussed below are expressly intended to support signal conditioning.

### 4.2.1 Actual Value / Setpoint Converter - FB 219 -(PHYSIKAL)

Function block FB 219 enables you to enter a physical setpoint and to output a physical actual value.
FB 219 serves only as interface between the control system and the user, and is thus independent of the control system itself.
You can invoke the actual value/setpoint converter at any point and as often as you want to input a setpoint and display an actual value.
It is recommended, however, that FB 219 be invoked in OB 1 (cyclically).

**Table 4-5. Calling and Initializing FB 219**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| XN | Actual value, scaled (input) | I W | DW 0 to 255 : KF = -2047 to 2047 | JU FB 219<br>NAME : PHYSIKAL<br>XN :<br>WP :<br>XP :<br>WN :<br>OGP :<br>UGP :<br>FAKT :<br>PAFE : |
| WP | Setpoint, physical (input) | I W | From setpoint adjuster (e.g. IW2), flag word or data word KF=-32768 to 32768 | |
| XP | Actual value, physical (output) | Q W | To digital display (QW6), flag word or data word KF=-32768 to 32768 | |
| WN | Setpoint, scaled (output) | Q W | DW 0 to 255 : KF= -2047 to 2047 | |

**Table 4-5. Calling and Initializing FB 219 (Cont.)**

| Param-eter | Description | Param. / Data Type | Assignment | STL |
|---|---|---|---|---|
| OGP | Upper limit of the physical value | D KF | KF=-32768 to 32767 | |
| UGP | Lower limit of the physical value | D KF | KF=-32768 to 32767 | |
| FAKT | Factor | D KF | Factor= $\frac{4096 \cdot 4096}{(OGP-UGP)} + 1$ <br> Condition: (OGP - UGP) > 512 <br> (OGP - UGP) < 512 | |
| PAFE | Parameter assignment error: Parameter range violation | A W | DW 0 to 255 <br> KM="0" or "1" <br> KM="1" indicates a range violation | |

| Parameter assignment error PAFE (data word) | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

Bit 0: XN

**Note:**

Enter the following values for the actual value/setpoint converter parameters when you use data block DB 130 (rough characteristic of a PT 100 resistance-type thermometer for FB: Polygon), which is part of the standard closed-loop control software package:

| | | | |
|---|---|---|---|
| Precision 1° | OGP : KF =266 | UGP : KF =-266 | FAKT : KF =31537 |
| Precision 0.1° | OGP : KF =2664 | UGP : KF=-2664 | FAKT : KF=3150 |

## 4.2.2 Influencing the Actual Value Branches

The S5-100U software provides auxiliary function blocks that allow you to influence the control system's actual value. These function blocks must be invoked immediately after the current actual value has been read in.

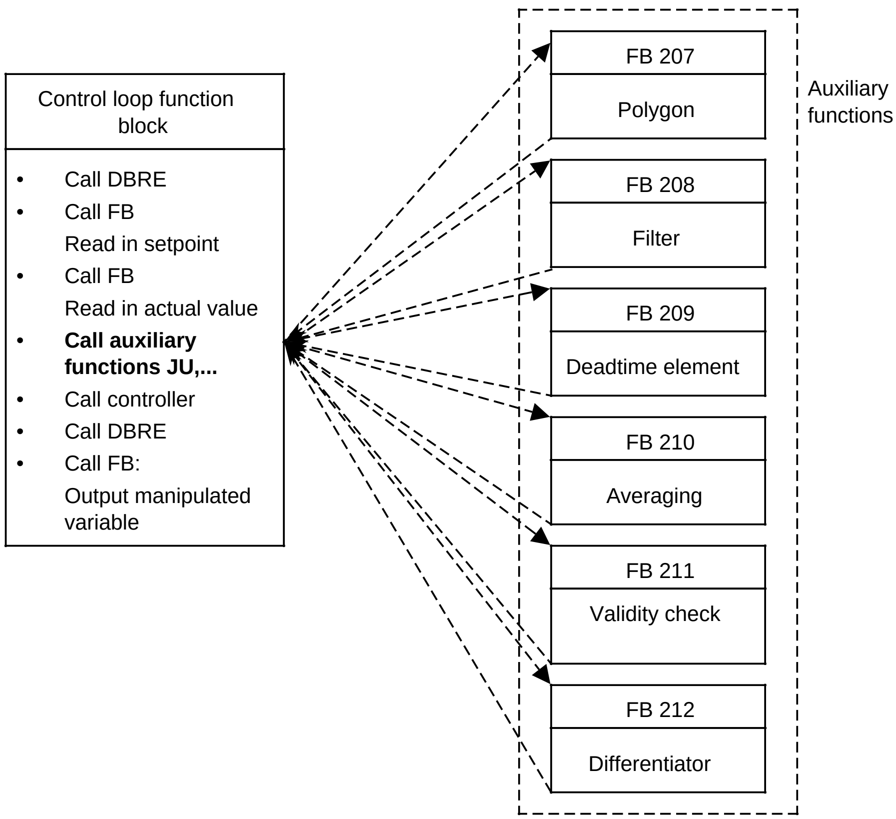



**Figure 4-12. Calling the Auxiliary Functions**

**Note:**
Before invoking an auxiliary function, you must first call the associated data block (C DB ...). As a rule, this is data block DBRE.

- **Polygon -FB 207- (POLYGON)**

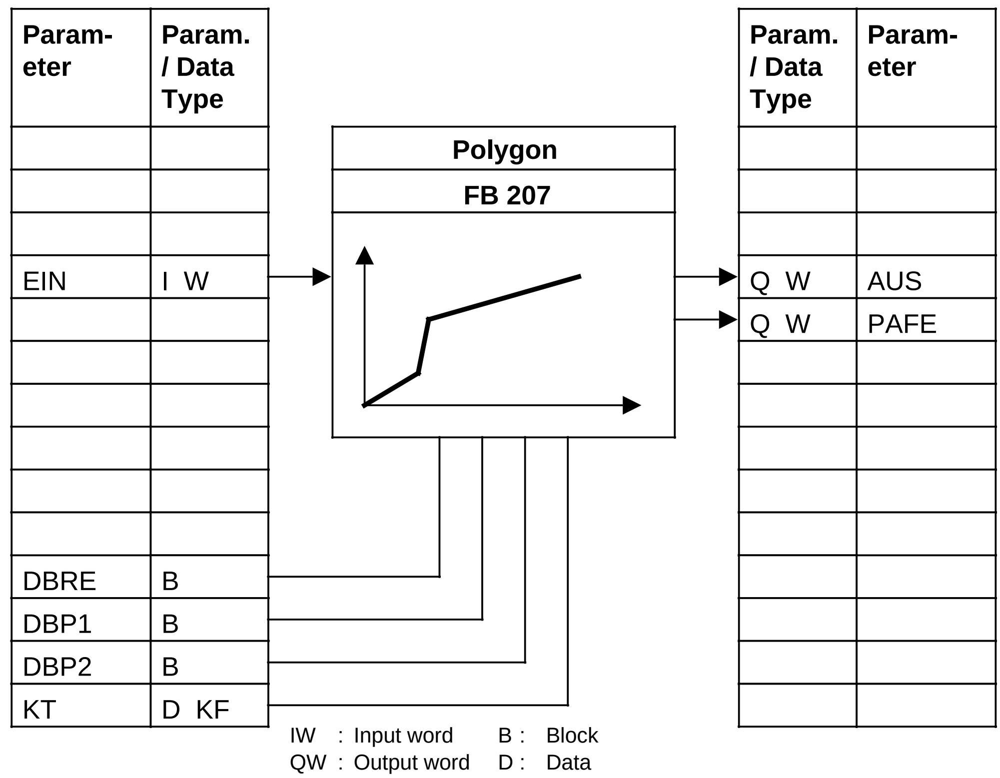



**Figure 4-13. Polygon -FB 207- (POLYGON)**

This function block allows you to convert the scaled input values KF=-2047 to 2047 to nonlinear output values; the output values must also be in the range KF=-2047 to 2047. Conversion is necessary when an actual value has a nonlinear characteristic (e.g. a PT 100 resistance-type thermometer).
Failure to convert the values in this case would result in the controller working with a falsified actual value.
The interpolation points of the nonlinear function y=f(x) are entered in data block DBP1 (for positive values) and DBP2 (for negative values). The interpolation points must lie within a scaled grid.

The mode of representation is set with the KT parameter:

- Unipolar representation: KT=0
  Data block DBP1 (positive values) is active
- Bipolar representation: KT=1
  Data blocks DBP1 (positive values) and DBP2 (negative values) are active.

You must generate separate DBP1/DBP2 data blocks for each nonlinear function.

You can choose either a rough characteristic or a precise characteristic by initializing data word DW3 in data blocks DBP1 / DBP2 accordingly:

- Rough characteristic: Data word DW3 (DBP1 / DBP2) KF = 0.
  The range of values KF=0 to 2047, 0 to -2047 is divided into 16 interpolation points.

- Precise characteristic: Data word DW3 (DBP1 / DBP2) KF=1.
  The selected subrange (zoom-in level) of the rough characteristic (e.g. KF=256 to 384) is also divided into eight interpolation points.

The associated data block must thus contain the following parameters:

- Rough/precise characteristic
- Interpolation points y
- Lower limit of input value X1
- Upper limit of input value X2

**Table 4-6. Initializing Data Block DBP1 / DBP2**

| Data Word | Contents for Rough Characteristic | Contents for Precise Characteristic | Description |
|---|---|---|---|
| DW0 | | | |
| DW1 | Internal | data word | |
| DW2 | Internal | data word | |
| DW3 | 0 | 1 | Rough/precise characteristic |
| DW4 | Y0 | Y0 | Interpolation point Y0 |
| DW5 | Y1 | Y1 | Interpolation point Y1 |
| DW6 | Y2 | Y2 | Interpolation point Y2 |
| DW7 | Y3 | Y3 | Interpolation point Y3 |
| DW8 | Y4 | Y4 | Interpolation point Y4 |
| DW9 | Y5 | Y5 | Interpolation point Y5 |
| DW10 | Y6 | Y6 | Interpolation point Y6 |
| DW11 | Y7 | Y7 | Interpolation point Y7 |
| DW12 | Y8 | Y8 | Interpolation point Y8 |
| DW13 | Y9 | | Interpolation point Y9 |
| DW14 | Y10 | | Interpolation point Y10 |
| DW15 | Y11 | | Interpolation point Y11 |
| DW16 | Y12 | | Interpolation point Y12 |
| DW17 | Y13 | | Interpolation point Y13 |
| DW18 | Y14 | | Interpolation point Y14 |
| DW19 | Y15 | | Interpolation point Y15 |
| DW20 | Y16 | | Interpolation point Y16 |
| DW21 | X1 | X1 | Lower limit of input value X1 |
| DW22 | X2 | X2 | Upper limit of input value X2 |

**Note:**

Output value AUS is not processed if the input value is not within the limits specified in DW21 and DW22.

**Table 4-7. Calling and Initializing FB 207 (POLYGON)**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| EIN | Input value (current DB) | I W | DW 0 to 255: KF=-2047 to 2047 | `: JU FB 207`<br>`NAME : POLYGON`<br>`EIN :`<br>`AUS :`<br>`DBRE :`<br>`DBP1 :`<br>`DBP2 :`<br>`KT :`<br>`PAFE :` |
| AUS | Output value (in DBRE) | Q W | DW 0 to 255: KF=-2047 to 2047 | |
| DBRE | Data block containing the controller parameters | B | DB 2 to 255 | |
| DBP1 | Data block containing the interpolation points for the positive input values | B | DB 2 to 255 | |
| DBP2 | Data block containing the interpolation points for the negative input values | B | DB 2 to 255 | |
| KT | Mode of representation: | D KF | "0" Unipolar (KF=0 to 2047)<br>"1" Bipolar (KF=-2047 to 2047) | |
| PAFE | Parameter assignment error: Parameter range violation | Q W | DW 0 to 255<br>KM="0" or "1"<br>KM="1" indicates a range violation | |

| Parameter assignment error PAFE (data word) | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

Bit 0: EIN

**Note:**
The software package contains a DBP1 data block (DB130), which represents a rough characteristic of the PT 100 (resistance-type thermometer). This characteristic is unipolar. Enter the following values for the OGR (upper limit) and UGR (lower limit) parameters if you are using function block FB 250:
OGR = 2047 UGR = 0

**Example:**

Representing the following value pairs for a nonlinear function by a rough characteristic (KF = 0 to 2047) and a precise characteristic (KF = 256 to 384) of the polygon.
The precise characteristic is a close-up of a subrange or section (KF = 256 to 384) of the rough characteristic.

Rough characteristic

| | X [KF] | Y [KF] |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 128 | 170 |
| 2 | 256 | 256 |
| 3 | 384 | 384 |
| 4 | 512 | 640 |
| 5 | 640 | 780 |
| 6 | 768 | 896 |
| 7 | 896 | 960 |
| 8 | 1024 | 1024 |
| 9 | 1152 | 1092 |
| 10 | 1280 | 1290 |
| 11 | 1408 | 1450 |
| 12 | 1536 | 1590 |
| 13 | 1664 | 1710 |
| 14 | 1792 | 1850 |
| 15 | 1920 | 1950 |
| 16 | 2048 | 2048 |

Precise characteristic

| | X [KF] | Y [KF] |
|---|---|---|
| 0 | 256 | 256 |
| 1 | 272 | 264 |
| 2 | 288 | 270 |
| 3 | 304 | 280 |
| 4 | 320 | 292 |
| 5 | 336 | 308 |
| 6 | 352 | 328 |
| 7 | 368 | 352 |
| 8 | 384 | 384 |

**Figure 4-14. Value Pairs**

You must generate a separate DBP0 data block for each pair of values.

```
Data block DBP1                              Data block DBP1
for rough characteristic: DB2                for precise characteristic: DB4

DB2                                          DB4

 0:  KF = +00000;                            0:  KF = +00000;
 1:  KF = +00000;                            1:  KF = +00000;
 2:  KF = +00000;                            2:  KF = +00000;
 3:  KF = +00000;  ROUGH                     3:  KF = +00001;  PRECISE
                   CHARACTERISTIC                              CHARACTERISTIC
 4:  KF = +00000;  Y0                        4:  KF = +00256;  Y0
 5:  KF = +00170;  Y1                        5:  KF = +00264;  Y1
 6:  KF = +00256;  Y2                        6:  KF = +00270;  Y2
 7:  KF = +00384;  Y3                        7:  KF = +00280;  Y3
 8:  KF = +00640;  Y4                        8:  KF = +00292;  Y4
 9:  KF = +00780;  Y5                        9:  KF = +00308;  Y5
10:  KF = +00896;  Y6                       10:  KF = +00328;  Y6
11:  KF = +00960;  Y7                       11:  KF = +00352;  Y7
12:  KF = +00000;  Y8                       12:  KF = +00384;  Y8
13:  KF = +01092;  Y9                         :
14:  KF = +01290;  Y10                        :
15:  KF = +01450;  Y11                        :
16:  KF = +01590;  Y12                        :
17:  KF = +01710;  Y13                        :
18;  KF = +01850;  Y14                        :
19:  KF = +01950;  Y15                        :
20:  KF = +02048;  Y16                        :
21:  KF = +00000;  LOWER X VALUE            21:  KF = +00256;  LOWER X VALUE
22:  KF = +02048;  UPPER X VALUE            22:  KF = +00384;  UPPER X VALUE
```

Calling FB: Polygon in the controller function block

```
FB 4

NAME :POLYGON

0005      :JU FB  207                  CALL POLYGON FOR DB2
0006 NAME :POLYGON
0007 EIN  :    DW   1                  INPUT VALUE FROM DBRE
0008 AUS  :    DW   2                  OUTPUT VALUE IN DBRE
0009 DBRE :    DB   1
000A DBP1 :    DB   2
000B DBP2 :    DB   3
000C KT   :    KF  +0                  UNIPOLAR REPRESENTATION
000D PAFE :    DW   4                  PARAMETER ASSIGNMENT ERROR
000E      :JU FB  207                  CALL POLYGON FOR DB4
000F NAME :POLYGON
0010 EIN  :    DW   2                  OUTPUT VAL. FROM ROUGH CHAR. FROM DBRE
0011 AUS  :    DW   3                  OUTPUT VAL. FROM PRECISE CHAR. IN DBRE
0012 DBRE :    DB   1
0013 DBP0 :    DB   4
0014 DBP1 :    DB   5
0015 KT   :    KF  +0                  UNIPOLAR REPRESENTATION
0016 PAFE :    DW   5                  PARAMETER ASSIGNMENT ERRORS
0017      :BE
```

- **Filter -FB 208- (PT1)**

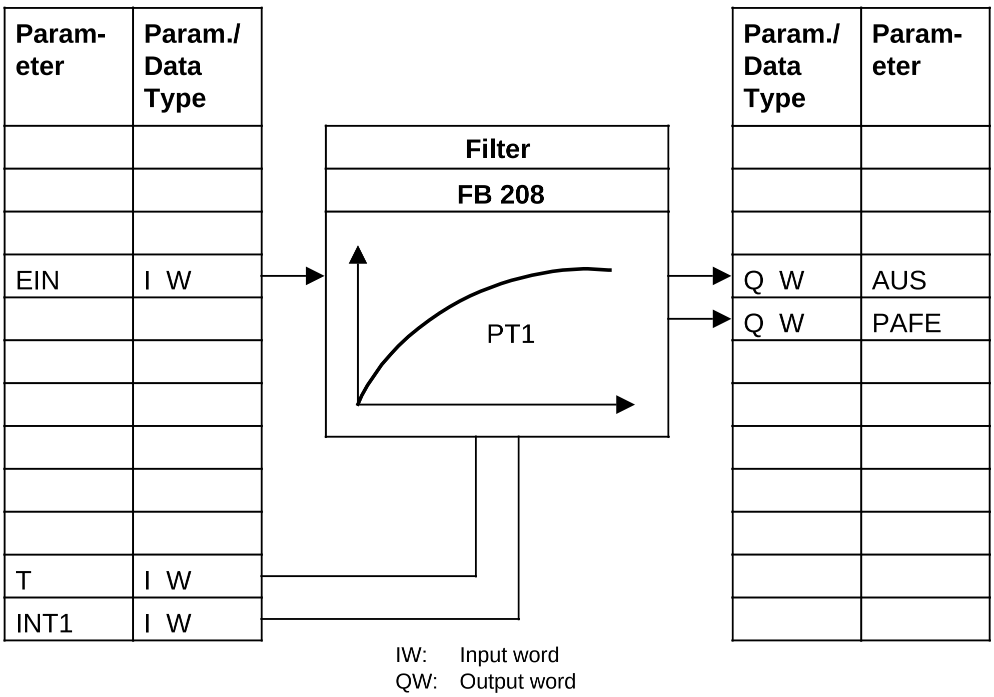



**Figure 4-15. Filter -FB 208- (PT1)**

An actual value is filtered by simulating a PT1 element. Function block FB89 thus implements the following transfer function:

$$K_{(p)} = \frac{1}{1+p \cdot T} \qquad \text{T: Time constant}$$

You can set the filtering quality via time constant T.

**Table 4-8. Calling and Initializing FB 208 (PT1)**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| EIN | Input value | I W | DW 0 to 255:<br>KF = -2047 to 2047 | `JU FB 208`<br>`NAME : PT1`<br>`EIN :`<br>`AUS :`<br>`T :`<br>`INT1 :`<br>`PAFE :` |
| AUS | Output value | Q W | DW 0 to 255:<br>KF=-2047 to 2047 | |
| T | Time constant | I W | DW 0 to 255:<br>KF=1 to 999 SEC | |
| INT1 | Internal data word | I W | DW 0 to 255:<br>No initialization required; data word need only be allocated | |
| PAFE | Parameter assignment errors:<br>Parameter range violation | Q W | DW 0 to 255<br>KM="0" or "1"<br>KM="1" indicates a range violation | |

| **Parameter assignment error PAFE (data word)** | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

- Bit 0: EIN
- Bit 1: T

- **Deadtime Element -FB 209- (VERZ)**

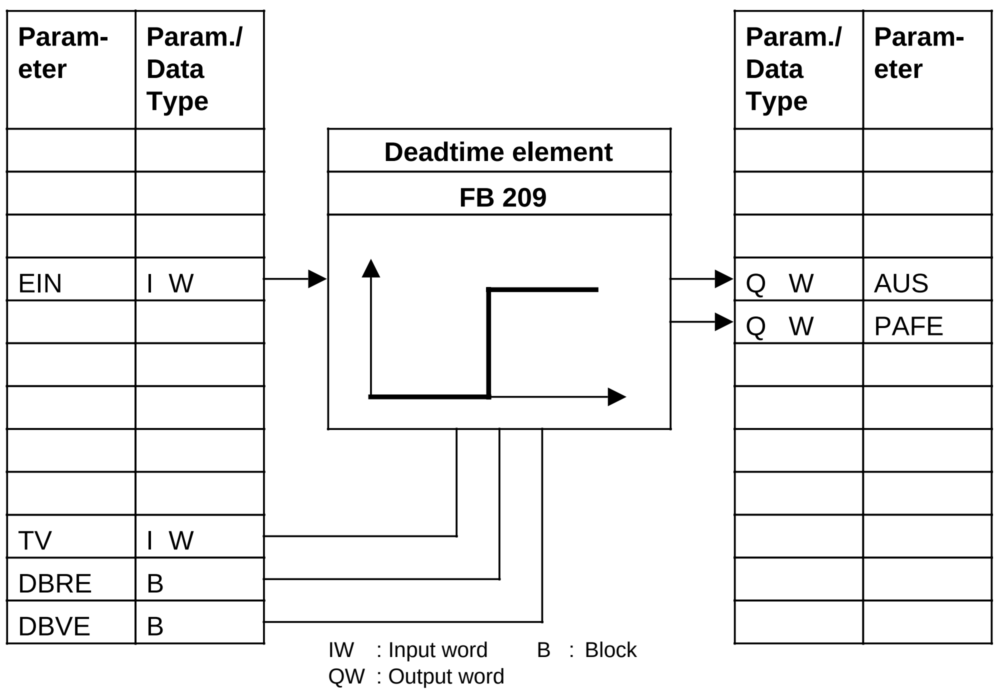



**Figure 4-16. Deadtime Element -FB 209- (VERZ)**

Function block FB 209 delays an input value (a measured value, for instance) by a specifiable amount of time. This delay (TV parameter) may be up to 30 seconds. You must first generate a DBVE data block comprising 31 data words.

If you require deadtimes of more than 30 seconds, you can call FB 209 several times, but you need a separate DBVE data block for each additional call.

**Table 4-9. Calling and Initializing FB 209 (VERZ)**

| Parameter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| EIN | Input value | I W | DW 0 to 255:<br>KF = -2047 to 2047 | JU FB 209<br>NAME: VERZ<br>EIN :<br>AUS :<br>TV :<br>DBRE:<br>DBVE:<br>PAFE: |
| AUS | Output value | Q W | DW 0 to 255:<br>KF=-2047 to 2047 | |
| TV | Delay | I W | DW 0 to 255:<br>KF=0 to 30 in sec. | |
| DBRE | Data block containing the control parameters | B | DB 2 to 255 | |
| DBVE | Data block for the deadtime element | B | DB 2 to 255 | |
| PAFE | Parameter assignment errors: Parameter range violation | Q W | DW 0 to 255<br>KM="0" or "1"<br>KM="1" indicates a range violation | |

| Parameter assignment error PAFE (data word) | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

Bit 0: EIN
Bit 1: TV

- **Averaging -FB 210- (MITTEL)**

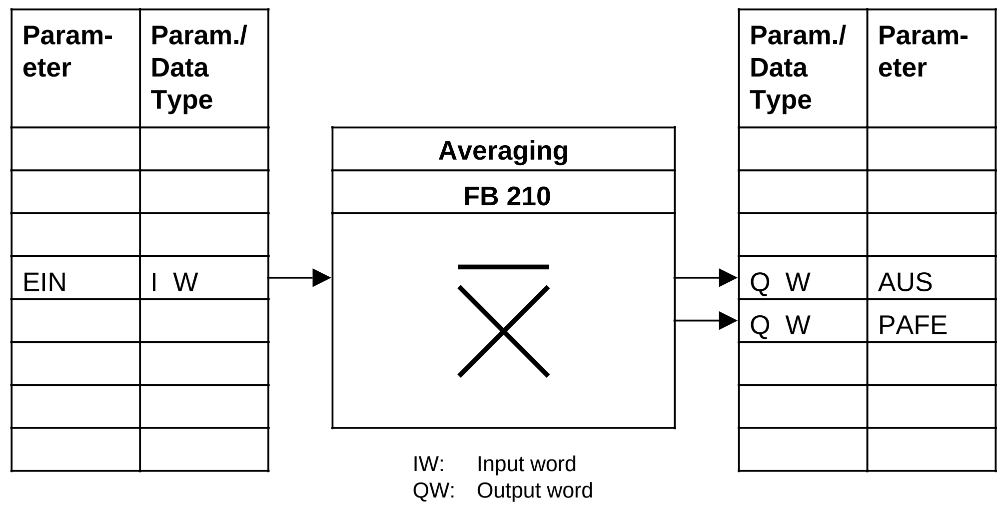



**Figure 4-17. Averaging -FB 210- (MITTEL)**

You can use this function block to average an actual value corrupted by disturbances.
FB 210 generates a new output value based on the old (averaged) output value.

**Note:**
Averaging corrupts correct values.
The "old" actual value is retained if the deviation between the actual value and the averaged output value is minimal.

**Table 4-10. Calling and Initializing FB 210 (MITTEL)**

| Param- eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| EIN | Input value | I W | DW 0 to 255:<br>KF = -2047 to 2047 | JU FB 210<br>NAME : MITTEL<br>EIN :<br>AUS :<br>PAFE : |
| AUS | Output value | Q W | DW 0 to 255:<br>KF=-2047 to 2047 | |
| PAFE | Parameter assignment errors: Parameter range violation | Q W | DW 0 to 255<br>KM="0" or "1"<br>KM="1" indicates a range violation | |

| Parameter assignment errors PAFE (data word) | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

Bit 0 — EIN

- **Validity check -FB 211- (PLAUS)**

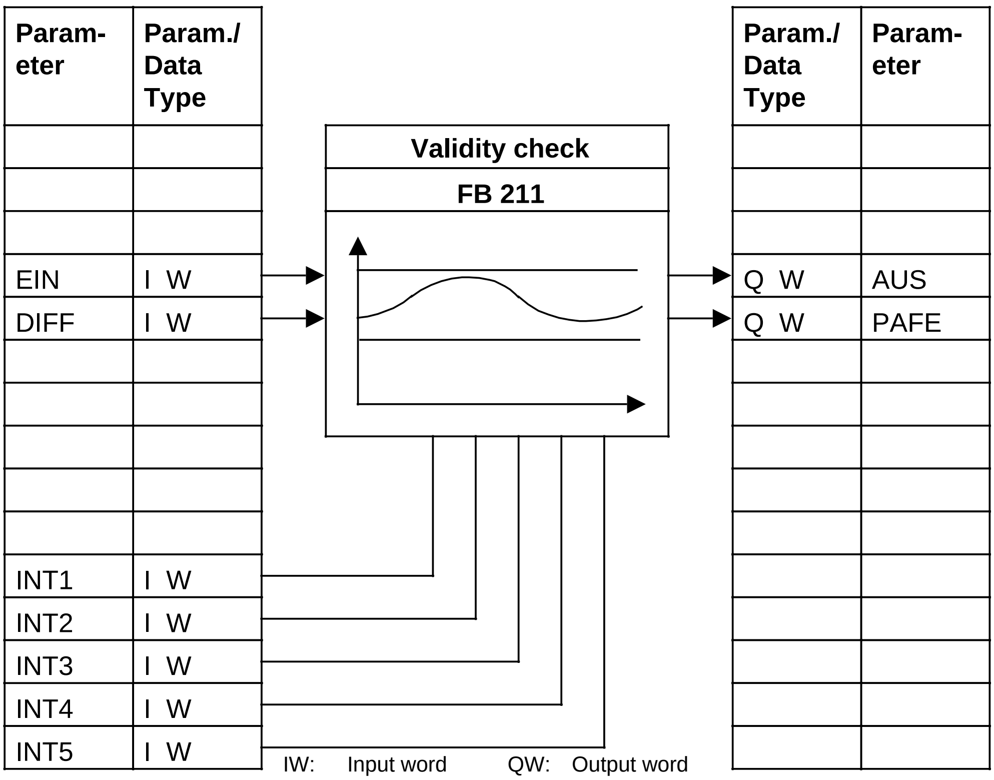



**Figure 4-18. Validity Check -FB 211- (PLAUS)**

Function block FB 211 suppresses disturbances which exceed a specifiable value (DIFF parameter) by comparing two successive sampling values (actual value, last sampling value) with the DIFF value.
If the difference is within the permissible range, the actual value is taken as the output value.
If the difference exceeds the permissible range, the current output value is computed as follows, taking into account the last three sampling periods:

$$AUSG = \frac{3}{2} \left( INT\,1 \; \frac{1}{3} \quad INT\,3 \right)$$

If the difference in the second sampling period is still out of range, the actual value is regarded as valid and is taken as the output value.

**Note:**
In contrast to the averaging function, the validity check function does not corrupt valid (correct) values.

**Table 4-11. Calling and Initializing FB 211 (PLAUS)**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| EIN | Input value | I W | DW 0 to 255:<br>KF = -2047 to 2047 | `JU FB 211`<br>`NAME : PLAUS`<br>`EIN :`<br>`AUS :`<br>`DIFF :`<br>`INT1 :`<br>`INT2 :`<br>`INT3 :`<br>`INT4 :`<br>`INT5 :`<br>`PAFE :` |
| AUS | Output value | Q W | DW 0 to 255:<br>KF=-2047 to 2047 | |
| DIFF | Permissible dif-ference | I W | DW 0 to 255:<br>KF=0 to 2047 | |
| INT1<br>INT2<br>INT3<br>INT4<br>INT5 | Internal data words | I W | DW 0 to 255:<br>No initialization necessary; data words need only be assigned | |
| PAFE | Parameter assignment errors: Para-meter range violations | Q W | DW 0 to 255<br>KM="0" or "1"<br>KM="1" indicates a range violation | |

| Parameter assignment errors PAFE (data word) | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

- Bit 0: EIN
- Bit 1: DIFF

- **Differentiator -FB 212- (DIFF)**

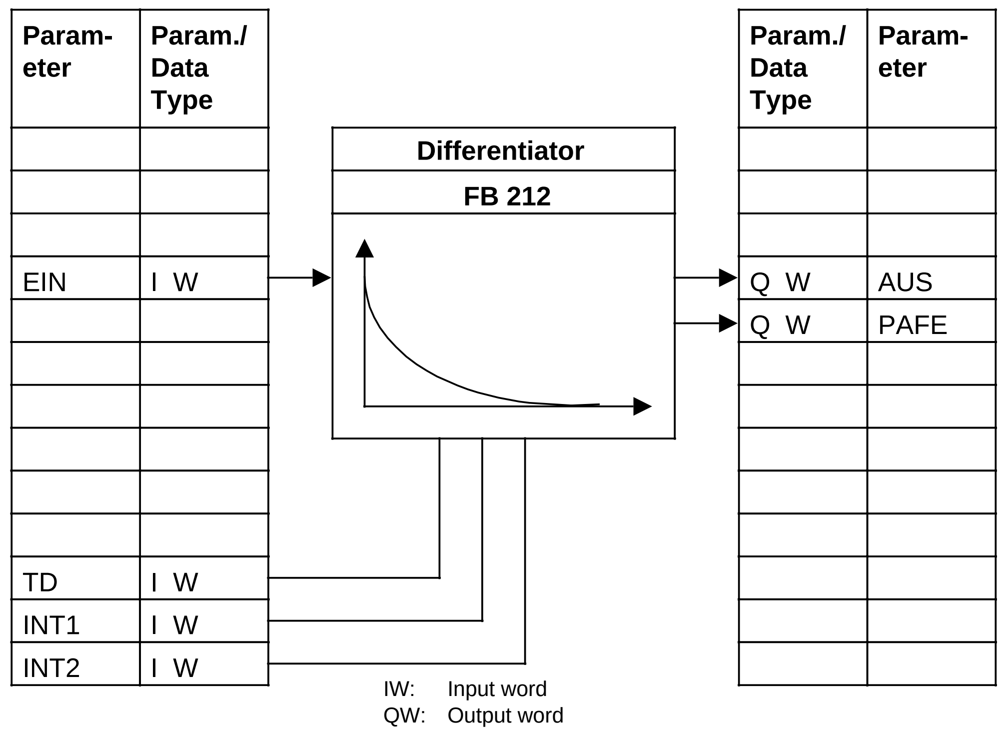



**Figure 4-19. Differentiator -FB 212- (DIFF)**

This function block enables differentiation of an actual value, which results in the following transfer function:

$$K_{(p)} = \frac{p \cdot dT}{(1+p \cdot dT)} \qquad dT\text{: Time constant}$$

Note:
Diffentiator FB 212 can function properly only when function block FB 208 (Filter) is in PLC memory.

**Table 4-12. Calling and Initializing FB 212 (DIFF)**

| Param-eter | Description | Param. / Data | Assignment | STL |
|---|---|---|---|---|
| EIN | Input value | I W | DW 0 to 255:<br>KF = -2047 to 2047 | `JU FB 212`<br>`NAME : DIFF`<br>`EIN :`<br>`AUS :`<br>`TD :`<br>`INT1 :`<br>`INT2 :`<br>`PAFE :` |
| AUS | Output value | Q W | DW 0 to 255:<br>KF = -2047 to 2047 | |
| TD | Time constant (derivative-action time) | I W | DW 0 to 255:<br>KF = 1 to 999 SEC | |
| INT1 | Internal data word | I W | DW 0 to 255<br>No initialization necessary; data word need only be assigned | |
| INT2 | Internal data word | Q W | DW 0 to 255:<br>No initialization necessary; data word need only be assigned | |
| PAFE | Parameter assignment errors: Para-meter range violations | Q W | DW 0 to 255<br>KM="0" or "1"<br>KM="1" indicates range violation | |

| Parameter assignment errors PAFE (data word) | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

Bit 0: EIN

Bit 1: TD

### 4.2.3 Influencing the Setpoint Branches

The function blocks listed below enable you to influence the control system's setpoint. The relevant auxiliary functions are invoked immediately after the current setpoint has been read in.

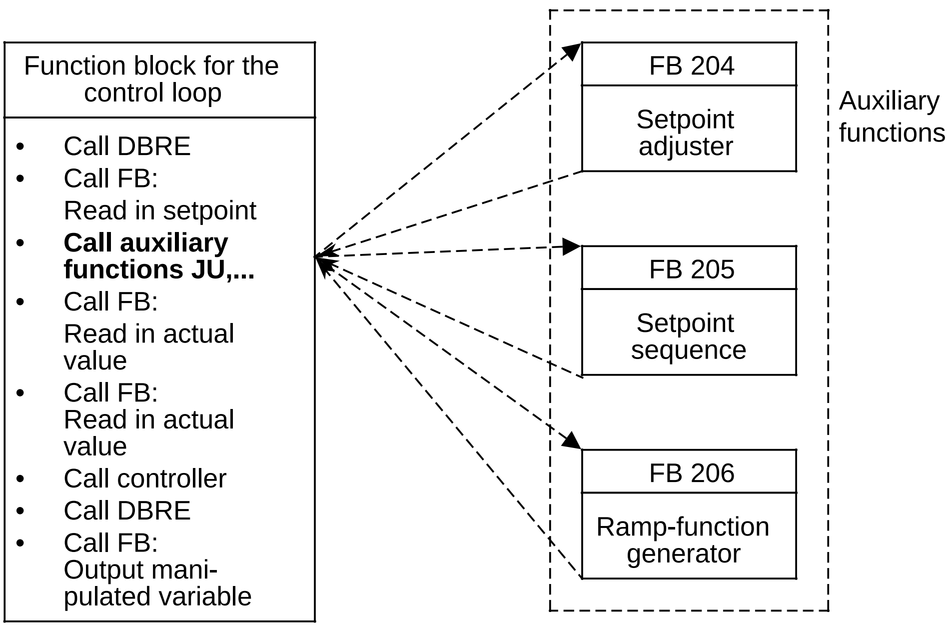



**Figure 4-20. Calling the Auxiliary Functions**

**Note:**
Before invoking an auxiliary function, you must first call the associated data block (C DB...). Normally, this is data block DBRE.

- **Setpoint adjuster - FB 204 - (SOLLW. ST)**

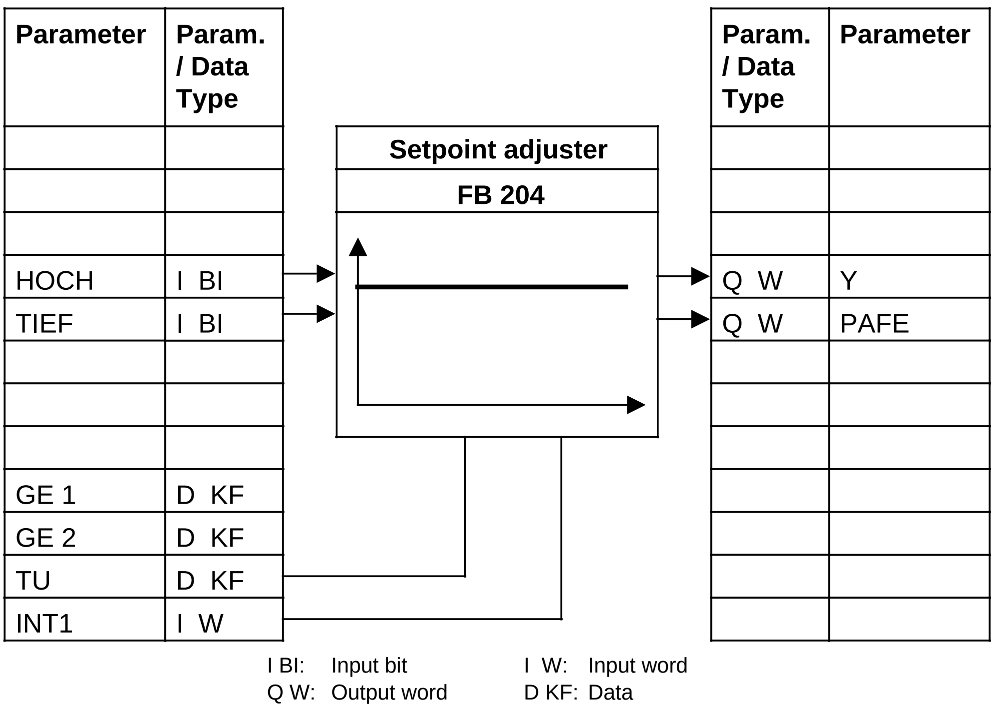



**Figure 4-21. Setpoint Adjuster - FB 204 - (SOLLW. ST)**

Function block FB 204 allows you to modify the setpoint (Y parameter) in accordance with the HOCH (raise) or TIEF (lower) parameter. If the HOCH or TIEF parameter is set permanently, the setpoint is modified at the faster rate (GE2 parameter) when the specified time (TU parameter) has elapsed.

**Table 4-13. Calling and Initializing FB 204 (SOLLW. ST)**

| Param-eter | Description | Param./ Data Type | Assignment | STL |
|---|---|---|---|---|
| HOCH | Raise setpoint | I BI | "0" or "1"<br>Setpoint is raised if "1" | JU FB 204<br>NAME :SOLLW. ST<br>HOCH :<br>TIEF :<br>GE 1 :<br>GE 2 :<br>TU :<br>INT 1 :<br>Y :<br>PAFE : |
| TIEF | Lower setpoint | I BI | "0" or "1"<br>Setpoint is lowered if "1" | |
| GE1 | Slow | D KF | KF = 0 to 2047 | |
| GE2 | Fast | D KF | KF = 0 to 2047 | |
| TU | Switch from slow to fast | D KF | KF = 0 to 2047 in seconds | |
| INT1 | Internal data word | I W | DW 0 to 255:<br>No initialization necessary; data word need only be reserved | |
| Y | Setpoint | Q W | KF = -2047 to 2047 | |
| PAFE | Parameter assignment errors: Parameter range violations | Q W | DW 0 to 255<br>KM = "0" or "1"<br>KM = "1" indicates range violation | |

| **Parameter assignment errors PAFE (data word)** | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

- Bit 0: GE 1
- Bit 1: GE 2
- Bit 2: TU

• **Setpoint sequence - FB 205 - (SOFOLGE)**

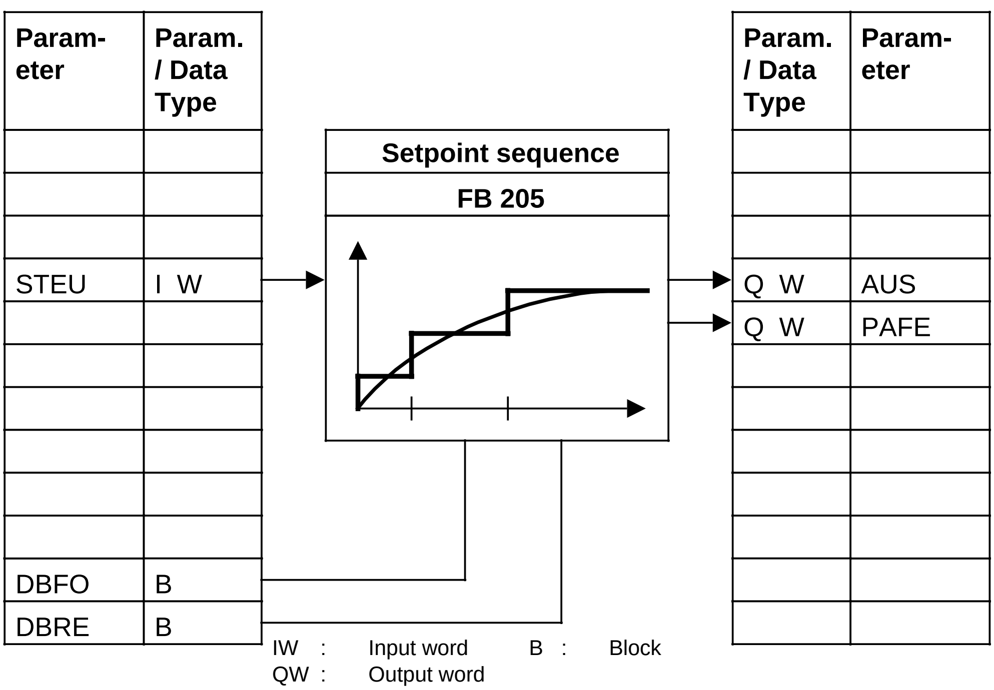



**Figure 4-22. Setpoint Sequence - FB 205 - (SOFOLGE)**

A setpoint sequence is generated from N = 1 to 122 setpoint interpolation points. You may choose between a staircase and a linear characteristic.

The interpolation points of the setpoint profile are stored in data block DBFO. Data block DBFO can accommodate up to 122 setpoint interpolation points, which are stored beginning with data word DW10. The number of interpolation points is stored in data word DW 6. You can choose between two output profiles by entering the appropriate value in data word DW 5:

- Setpoint is output in staircase form: DW5 KF = 0
- Setpoint is interpolated linearly between two interpolation points: DW5 KF = 1

The function block is activated by setting bit 0 in control word STEU. When the specified setpoint profile has been processed, bit 0 is reset in the control word and the current setpoint transferred without change to controller data block DBRE.

**Table 4-14. Calling and Initializing FB 205 (SOFOLGE)**

| Parameter | Description | Param. / Data Type | Assignment | STL |
|---|---|---|---|---|
| STEU | Control word (in DBRE) | I W | DW 0 to 255:<br>Bit 0 = 1, Activate setpoint sequence<br>Bit 0 = 0, Setpoint sequence deactivated | `JU FB 205`<br>`NAME : SOFOLGE`<br>`STEU :`<br>`AUS :`<br>`DBFO :`<br>`DBRE :` |
| AUS | Output value (in DBRE) | Q W | DW 0 to 255:<br>KF = -2047 to 2047 | |
| DBFO | Data block containing the parameters for the setpoint sequence | B | DB 2 to 255 | |
| DBRE | Data block containing the controller parameters | B | DB 2 to 255 | |

**Note:**
The control word for the setpoint sequence is not identical to the controller's control word, as bit 0 of the control word is reset automatically after the setpoint sequence has been processed.

**C A U T I O N:**
The interpolation points in data block DBF0 must be in the range
KF= -2047 to 2047.

**Table 4-15. Initializing Data Block DBF0**

| | |
|---|---|
| DW0 : | |
| DW1 : | |
| DW2 : | |
| DW3 : | Internal data word |
| DW4 : | Internal data word |
| DW5 : | KF = "0: Staircase; 1: Linear" |
| DW6 : | KF = "No. of interpolation points (max. 122)" |
| : | |
| : | |
| DW10: | KF = "TIME 0 in seconds (initial value KF = 0)" |
| DW11: | KF = "SETP 0 (KF = -2047 to 2047)" |
| DW12: | KF = "TIME 1 in seconds (KF = 0 to 1000)" |
| DW13: | KF = "SETP 1 (KF = -2047 to 2047)" |
| : | |
| : | |

**Note:**
Time 0 (starting instant) is always zero. The times for the setpoint interpolation points are summated (maximum value 32,767), beginning from the starting instant. The maximum difference between $TIME_N$ and $TIME_{N-1}$ may not exceed KF = 1000. If the sampling interval is set to one second, the maximum active time T = 32,767 seconds, or approximately 9 hours. The maximum number of interpolation points is 122.

- **Ramp-function generator - FB 206 - (HOCHLAUF)**

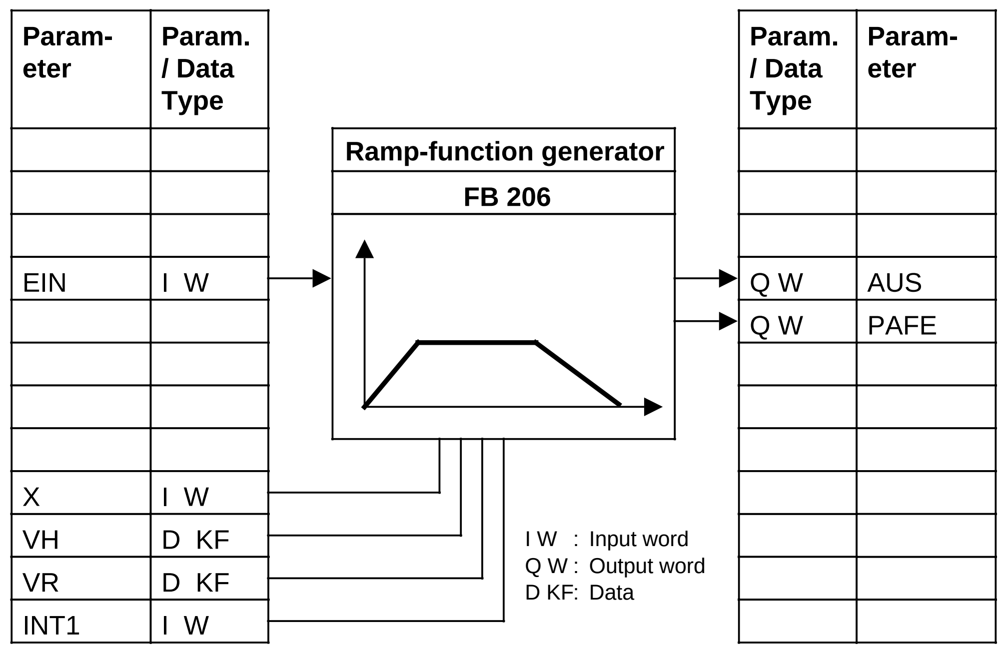



**Figure 4-23. Ramp-Function Generator - FB 206 - (HOCHLAUF)**

In the event of a step-change in the setpoint, this function block generates an output value (AUS), which ramps up to the setpoint value SOLL.
If the step change in the setpoint is in the direction of the actual value, the ramp-function generator starts ramping up at the actual value.

Two values are required to initialize the ramp-function generator:

VH: The ramp-up rate is the value by which the output increases per sampling interval in the direction of the input value.
The ramp-up rate applies for input steps in the direction of ± 100 %.

VR: The ramp-down rate is the value by which the output decreases per sampling interval in the direction of the input value.
The ramp-down rate applies for input steps in the direction of 0 %.

**Table 4-16. Calling and Initializing FB 206 (HOCHLAUF)**

| Param-eter | Description | Param. / Data Type | Assignment | STL |
|---|---|---|---|---|
| X | Actual value | I W | DW 0 to 255:<br>KF = - 2047 to 2047 | JU FB 206<br>NAME : HOCHLAUF<br>X :<br>EIN :<br>AUS :<br>VH :<br>VR :<br>INT1 :<br>PAFE : |
| EIN | Input value | I W | DW 0 to 255:<br>KF = - 2047 to 2047 | |
| AUS | Output value | Q W | DW 0 to 255:<br>KF = - 2047 to 2047 | |
| VH | Ramp-up rate | D KF | DW 0 to 255:<br>KF= 0 to 2047 | |
| VR | Ramp-down rate | D KF | DW 0 to 255:<br>KF= 0 to 2047 | |
| INT1 | Internal data word | I W | DW 0 to 255:<br>No initialization necessary; data word need only be reserved | |
| PAFE | Parameter assignment errors: Parameter range violations | Q W | DW 0 to 255<br>KM = "0" or "1"<br>KM = "1" indicates range violation | |

| **Parameter assignment errors PAFE (data word)** | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

- Bit 0: X
- Bit 1: EIN
- Bit 2: VH
- Bit 3: VR

## 4.2.4 Limit Monitor - FB 213 - (GRENZ)

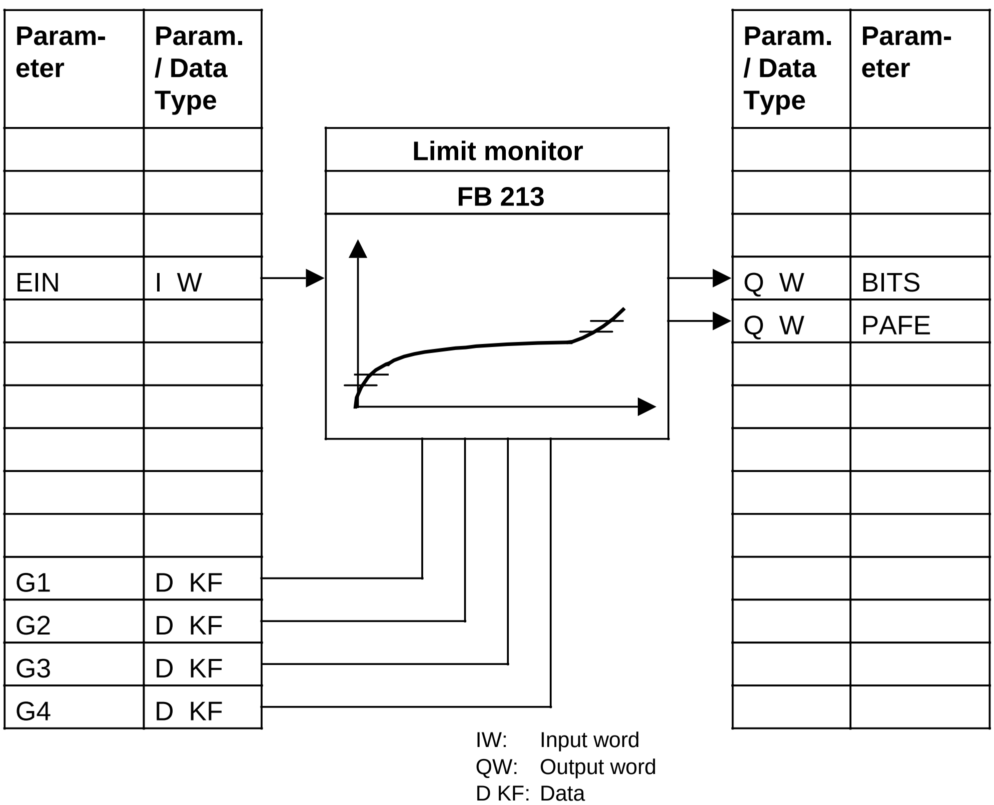



**Figure 4-24. Limit Monitor - FB 213 - (GRENZ)**

Function block FB 213 compares input signal EIN with four specifiable limit values (GR1 to GR4). Bits are set in the data word or flag word specified as BITS parameter when the input signal exceeds positive, or falls below negative, limiting values.

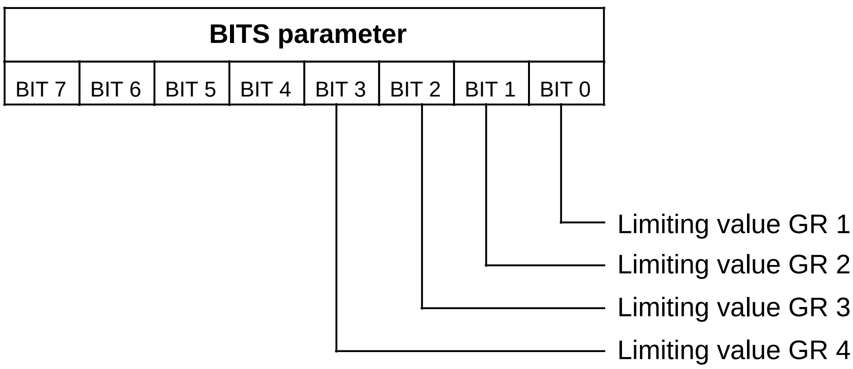



**Table 4.17. Calling and Initializing FB 213 (GRENZ)**

| Param-eter | Description | Param. / Data Type | Assignment | STL |
|---|---|---|---|---|
| EIN | Input value | I W | DW 0 to 255:<br>KF= - 2047 to 2047 | JU FB 213<br>NAME : GRENZ<br>EIN :<br>BITS :<br>G1 :<br>G2 :<br>G3 :<br>G4 :<br>PAFE : |
| BITS | Limiting value bits | Q W | MW0 to 255 or<br>DW 0 to 255 | |
| G1 | Limiting value 1 | D KF | KF= - 2047 to 2047 | |
| G2 | Limiting value 2 | D KF | KF= - 2047 to 2047 | |
| G3 | Limiting value 3 | D KF | KF= - 2047 to 2047 | |
| G4 | Limiting value 4 | D KF | KF= - 2047 to 2047 | |
| PAFE | Parameter assignment errors:<br>Parameter range violations | Q W | DW 0 to 255<br>KM="0" or "1"<br>KM= "1"<br>indicates range violation | |

| **Parameter assignment errors PAFE (data word)** | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Range violation detected for parameter

EIN

G1

G2

G3

G4

# 5 Sophisticated Applications

**Figures**

# 5 Sophisticated Applications

## 5.1 Modular Control

The modularity of the "S5-100U Closed-Loop Control" software allows you to generate any control structure you may require. The controllers in such a modular control system are linked via data interfaces, which are handled in a manner similar to the interfaces used in hardwired systems. The data interfaces (data words) are located in the controller's DBRE data block. It is thus possible to establish an individually-tailored and modular control system.

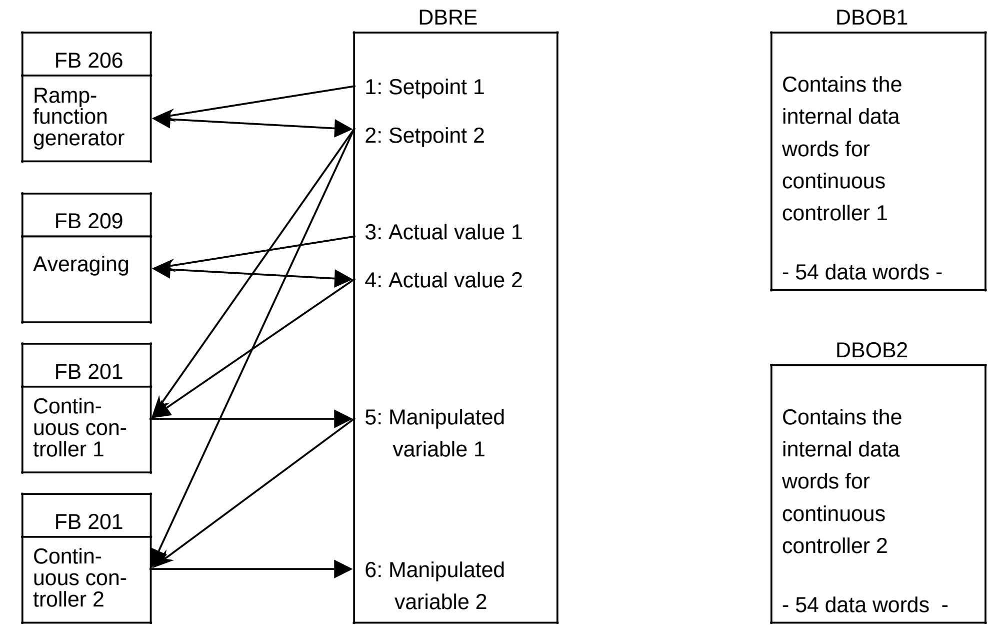


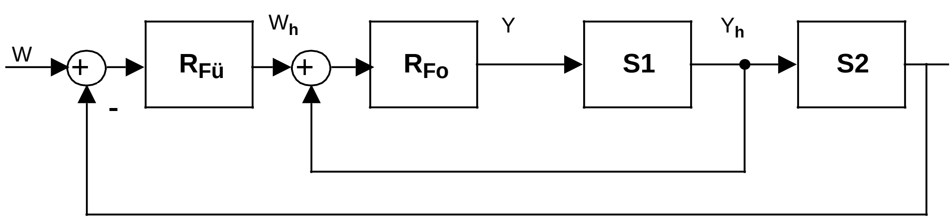

**Figure 5-1. Structure of a Modular Control System**

**Note:**

The data interfaces in this modular closed-loop control system enable you to manipulate intermediate values using suitable STEP 5 statements. This is necessary, for instance, for flow-rate measurement (extracting the root of the output value of the orifice plate).

## 5.2 Cascade Control

The control characteristics of the single-loop-system can be improved by cascading two controllers. The master controller, which has to maintain the primary controlled variable at the setpoint W, sends the slave controller the manipulated variable $Y_F$ as auxiliary setpoint $W_h$; the slave controller then sees to it that the auxiliary controlled variable attains the value of the auxiliary setpoint. A cascade control system can also have several auxiliary controlled variables, which means that several slave controllers can be connected to a single master controller.

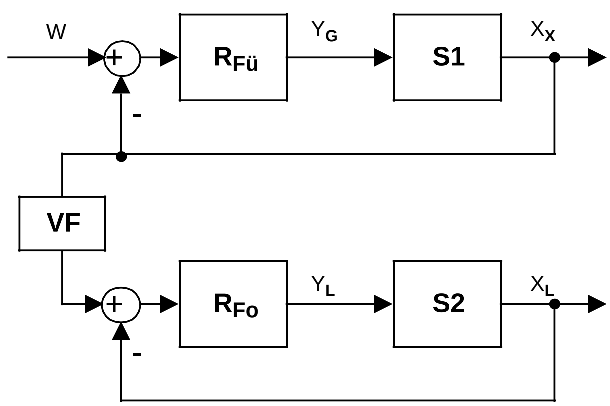



| | | | | | |
|---|---|---|---|---|---|
| X | Actual value | W | Setpoint | $R_{Fo}$ | Slave controller |
| $X_h$ | Auxiliary actual value | $W_h$ | Auxiliary setpoint | S1 | Controlled subsystem 1 |
| Y | Manipulated variable | $R_{Fü}$ | Master controller | S2 | Controlled subsystem 2 |

**Figure 5-2. Signal Flow in a Cascade Control System**

## 5.3 Ratio Control

A ratio control system is used to maintain several process parameters in a constant ratio to one another. One of the simplest examples of ratio control is the regulation of the supply of gas and air to a furnace. The master controller regulates the flow of gas according to the temperature setpoint; the slave controller uses the actual value of the master controller to regulate the flow of air. The ratio between the two process parameters for gas and air is set with the ratio factor on the slave controller.



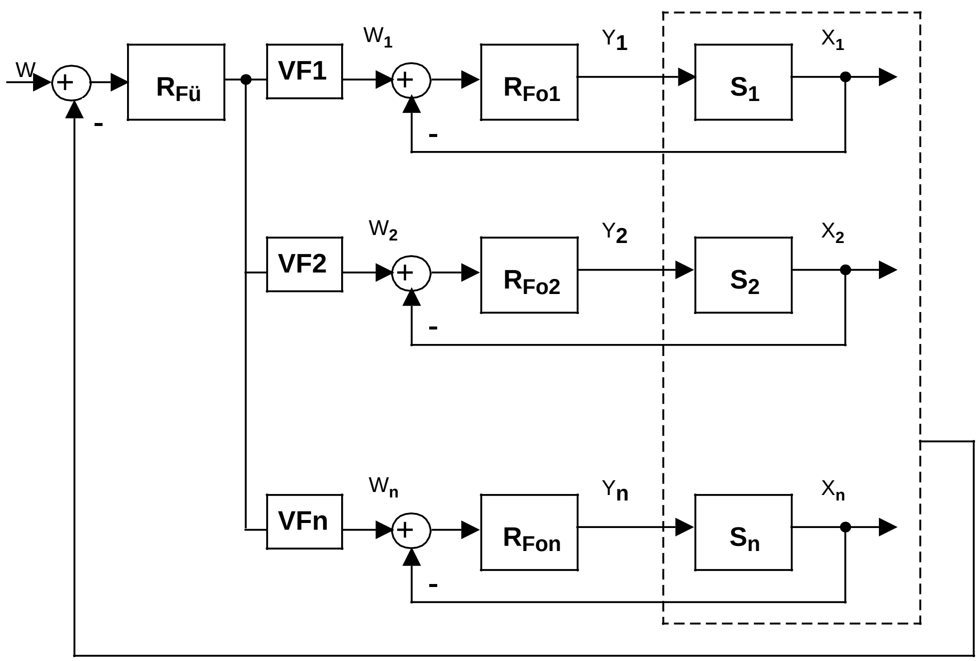

| | | | | | |
|---|---|---|---|---|---|
| X | Actual temperature | W | Temperature setpoint | S2 | Controlled system for air flow |
| $X_L$ | Actual air flow | $R_{Fü}$ | Master controller | VF | Multiplier for ratio factor |
| $Y_G$ | Manipulated variable of gas flow | $R_{FO}$ | Slave controller | | |
| $Y_L$ | Manipulated variable of air flow | S1 | Controlled system for gas flow | | |

**Figure 5-3. Signal Flow in a Ratio Control System**

## 5.4 Blending Control

The task of a blending control system is to maintain several blending components, comprising main components and additives, in a constant ratio to each other. The master controller regulates the total quantity by supplying all the subordinate component controllers with its manipulated variable as setpoint. The percentage of each component in the total quantity is set with the ratio factor.

The sum of all ratio factors is 100 %.




| | | | |
|---|---|---|---|
| $x_1...x_n$ | Actual value of components 1...n | $R_{Fü}$ | Master controller |
| $x_g$ | Actual value of total quantity | $P_{Fo}$ | Slave controller for components 1...n |
| $y_1...y_n$ | Manipulated variable for components 1...n | $S_1...S_n$ | Controlled systems for components 1...n |
| | | $VF_1...VF_n$ | Multiplier for ratio factors 1...n |
| w | Setpoint of total quantity | | |
| $w_1 ...w_n$ | Setpoint for components 1...n | | |

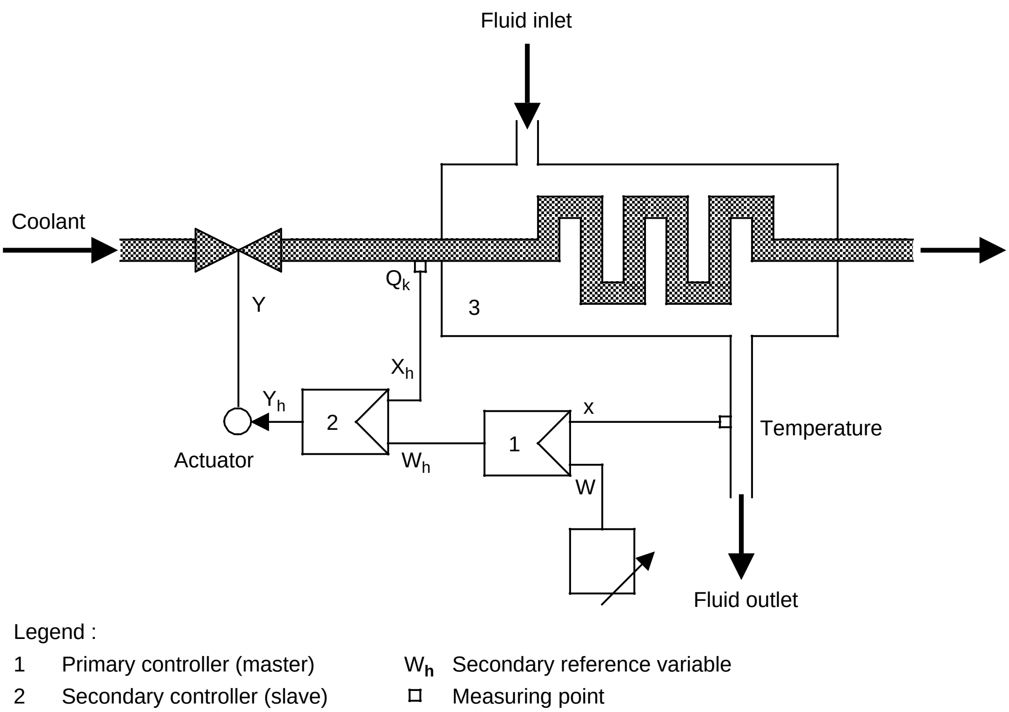
**Figure 5-4. Signal Flow in a Blending Control System**

# 6 Complex Example

**Figures**

# 6 Complex Example

The following example shows the application of cascade control for temperature and volumetric flow.
The primary controlled variable (x) is the temperature of a fluid that is to be cooled to a predetermined setpoint by means of a heat exchanger.
The controller output variable $y_R$ regulates the supply of coolant to the heat exchanger, and thus the temperature (controlled variable) of the fluid. The coolant flow $Q_k$ serves as the secondary controlled variable (auxiliary actual value $X_h$).




Legend :

| | | | |
|---|---|---|---|
| 1 | Primary controller (master) | $W_h$ | Secondary reference variable |
| 2 | Secondary controller (slave) | □ | Measuring point |
| 3 | Heat exchanger | $Q_k$ | Coolant flow |
| $X_h$ | Secondary controlled variable | | |

**Figure 6-1. Example of a Cascade Control System for Temperature and Volumetric Flow**

This cascade control system is to be implemented with the "S5-100U Closed-Loop Control" software. This entails the following prerequisites:

- Primary controller : Continuous-action controller
- Master controller : Continuous-action controller
- Setpoint adjuster
- Validity check in the actual value branch

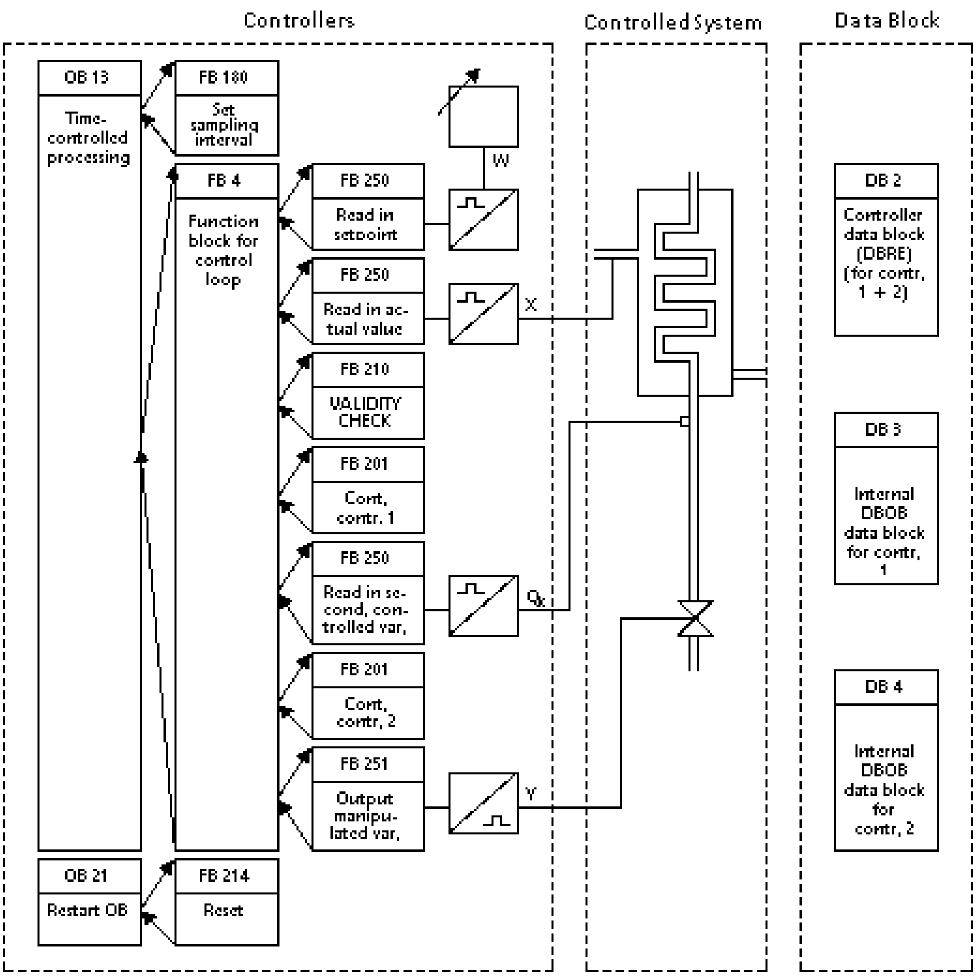



**Figure 6-2. Structure of the Control Loop**

# Appendix A

**Figures**



**Tables**

# A Appendix

## A.1 Overview

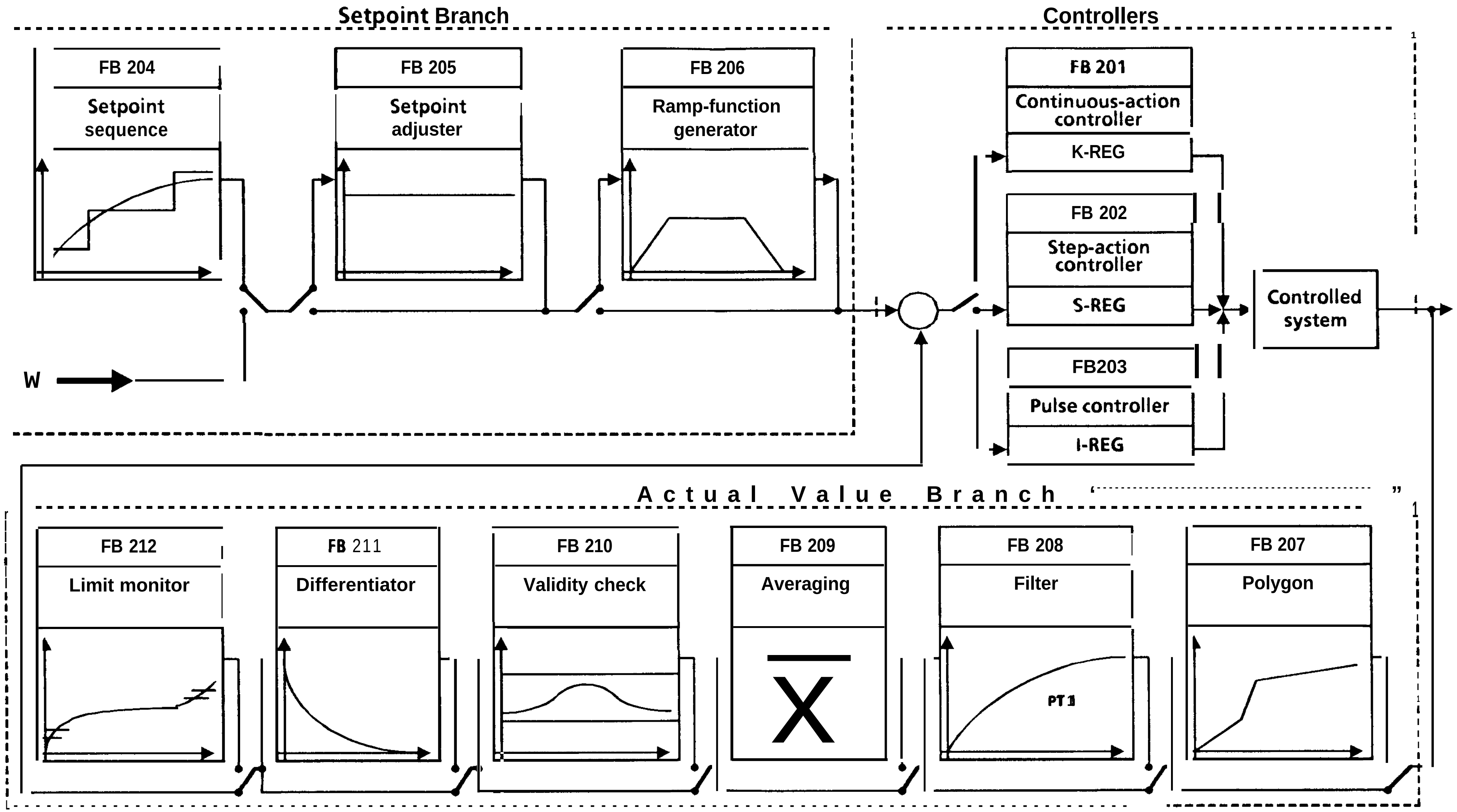



Figure A-1. Overview

## A.2 Function Blocks

**Table A-1. Function Blocks**

| FB No. | Brief Description | STL / Parameters |
| --- | --- | --- |
| FB 201 | Continuous-action controller<br><br>K-<br>REG | : JU FB 201<br>NAME : K-REG<br>X : Actual value<br>W : Setpoint<br>YF : Correction<br>Y : Manipulated variable<br>XD : System deviation<br>STEU : Control word for the controller<br>K : Gain<br>P : Proportional gain<br>TI : Integration time<br>TV : Derivative-action time<br>SEPD : Separate D input<br>Z : Feedforward control<br>DBRE : Controller data block<br>DBOB : Internal data block<br>PAFE : Parameter assignment errors |
| FB 202 | Step-action controller<br><br>S-<br>REG | : JU FB 202<br>NAME : S-REG<br>X : Actual value<br>W : Setpoint<br>Y+ : "FORWARD" actuator<br>Y- : "REVERSE" actuator<br>XD : System deviation<br>STEU : Control word for the controller<br>AN : Upper resp. thresh. for hysteresis<br>AB : Lower resp. thresh. for hysteresis<br>TMIN : Minimum pulse duration<br>TS : Actuating time<br>K : Gain<br>P : Proportional gain<br>TI : Integration time<br>TV : Derivative-action time<br>SEPD : Separate D input<br>Z : Feedforward control<br>DBRE : Controller data block<br>DBOB : Internal data block<br>PAFE : Parameter assignment errors |

**Table A-1. Function Blocks (Cont.)**

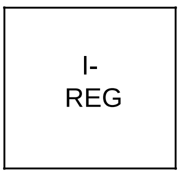
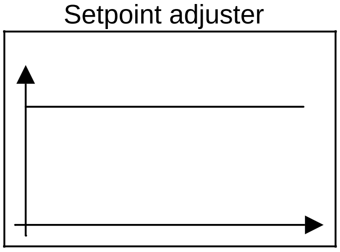
| FB No. | Brief Description | STL / Parameters |
|---|---|---|
| FB 203 | Pulse controller<br><br> | `     : JU FB 203`<br>`NAME : IMPULS`<br>`X    :` Actual value<br>`W    :` Setpoint<br>`Y +  :` Actuator "HEATING"<br>`Y -  :` Actuator "COOLING"<br>`Y    :` Manipulated variable<br>`XD   :` System deviation<br>`STEU :` Control word for the controller<br>`K    :` Gain<br>`P    :` Proportional gain<br>`TI   :` Integration time<br>`TV   :` Derivative-action time<br>`TPER :` Time period<br>`AN   :` Pulse response threshold<br>`Y+/- :` Ratio: HEATING / COOLING<br>`OZ   :` Zone control: Upper response threshold<br>`UZ   :` Zone control: Lower response threshold<br>`SEPD :` Separate D input<br>`Z    :` Feedforward control<br>`DBRE :` Controller data block<br>`DBOB :` Int. data block<br>`PAFE :` Parameter assignment errors |
| FB 204 | Setpoint adjuster<br> | `     : JU FB 204`<br>`NAME : SOLLW.ST.`<br>`HOCH :` Raise setpoint<br>`TIEF :` Lower setpoint<br>`GE1  :` Slow<br>`GE2  :` Fast<br>`TU   :` Changeover time<br>`INT1 :` Int. data word<br>`Y    :` Setpoint<br>`PAFE :` Parameter assignment errors |

**Table A-1. Function Blocks (Cont.)**

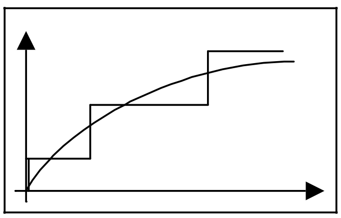
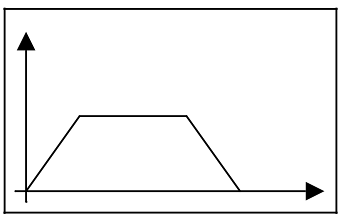
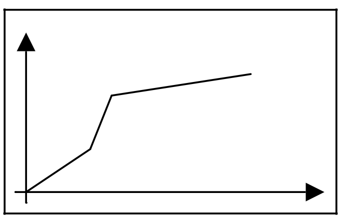
| FB No. | Brief Description | STL / Parameters |
|---|---|---|
| FB 205 | Setpoint sequence  | : JU FB 205<br>NAME : SOFOLGE<br>STEU : Control word<br>AUS : Output value<br>DBF0 : DB for setpoint sequencer<br>DBRE : Controller DB |
| FB 206 | Ramp-function generator  | : JU FB 206<br>NAME : HOCHLAUF<br>X : Actual value<br>EIN : Input value<br>AUS : Output value<br>VH : Ramp-up rate<br>VR : Ramp-down rate<br>PAFE : Parameter assignment errors |
| FB 207 | Polygon  | : JU FB 207<br>NAME : POLYGON<br>EIN : Input value<br>AUS : Output value<br>DBRE : Controller data block<br>DBP1 : DB containing the interpolation points for pos. input values<br>DBP2 : DB containing the interpolation points for neg. input values<br>KT : Representation mode<br>PAFE : Parameter assignment errors |

**Table A-1. Function Blocks (Cont.)**

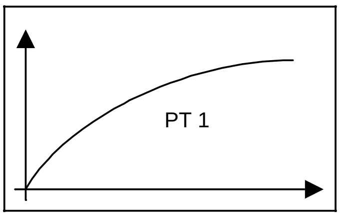
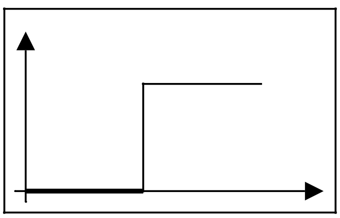
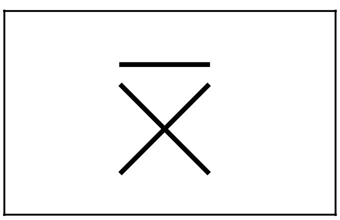
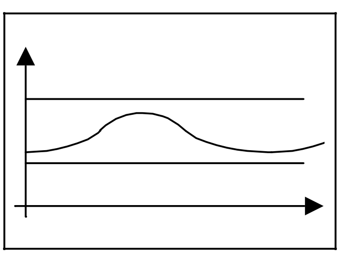
| FB No. | Brief Description | STL / Parameters |
|---|---|---|
| FB 208 | Filter<br> | `      : JU FB 208`<br>`NAME  : PT1`<br>`EIN   :` Input value<br>`AUS   :` Output value<br>`T     :` Time constant<br>`INT1  :` Internal data word<br>`PAFE  :` Parameter assignment errors |
| FB 209 | Deadtime element<br> | `      : JU FB 209`<br>`NAME  : VERZ`<br>`EIN   :` Input value<br>`AUS   :` Output value<br>`TV    :` Dead time<br>`DBRE  :` Controller DB<br>`DBVE  :` DB for deadtime element<br>`PAFE  :` Parameter assignment errors |
| FB 210 | Averaging<br> | `      : JU FB 210`<br>`NAME  : MITTEL`<br>`EIN   :` Input value<br>`AUS   :` Output value<br>`PAFE  :` Parameter assignment errors |
| FB 211 | Validity check<br> | `      : JU FB 211`<br>`NAME  : PLAUS`<br>`EIN   :` Input value<br>`AUS   :` Output value<br>`DIFF  :` Perm. difference<br>`INT1  :` Internal data word<br>`INT2  :` Internal data word<br>`INT3  :` Internal data word<br>`INT4  :` Internal data word<br>`INT5  :` Internal data word<br>`PAFE  :` Parameter assignment errors |

**Table A-1. Function Blocks (Cont.)**

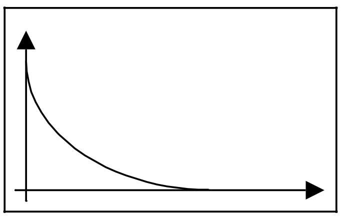
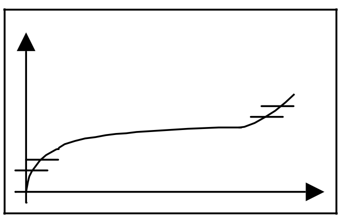
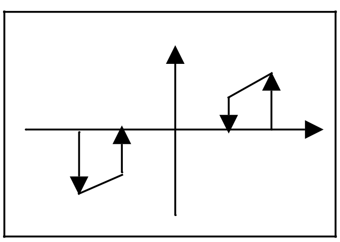
| FB No. | Brief Description | STL / Parameters |
|---|---|---|
| FB 212 | Differentiator<br> | : JU FB 212<br>NAME : DIFF<br>EIN : Input value<br>AUS : Output value<br>TD : Time constant<br>INT1 : Internal data word<br>INT2 : Internal data word<br>PAFE : Parameter assignment errors |
| FB 213 | Limit monitor<br> | : JU FB 213<br>NAME : GRENZ<br>EIN : Input value<br>BITS : Limit bits<br>G1 : Limiting value 1<br>G2 : Limiting value 2<br>G3 : Limiting value 3<br>G4 : Limiting value 4<br>PAFE : Parameter assignment errors |
| FB 214 | Dead band with hysteresis<br> | : JU FB 214<br>NAME : HYSTERES<br>W : Setpoint<br>X : Actual value<br>AUS : Output<br>AN : Upper resp. thresh. of hysteresis<br>AB : Lower resp. thresh. of hysteresis<br>INT1 : Internal data word<br>PAFE : Parameter assignment errors |
| FB 215 | Reset | : JU FB 215<br>NAME : ANLAUF<br>DB01 : DBOBs to be reset (2 DBs)<br>DB02 : DBOBs to be reset (2 DBs) |

**Table A-1. Function Blocks (Cont.)**

| FB No. | Brief Description | STL / Parameters |
|---|---|---|
| FB 216 | Time-slice distributor | `     : JU FB 216`<br>`NAME : START`<br>`S3S2 :`<br>`S5S4 :`<br>`TA   :` |
| FB 217 | Save scratch flags | `     : JU FB 217`<br>`NAME : MERET` |
| FB 218 | Load scratch flags | `     : JU FB 218`<br>`NAME : MELAD` |
| FB 219 | Actual value/setpoint converter | `     : JU FB 219`<br>`NAME : PHYSIKAL`<br>`XN   :` Scaled actual value<br>`WP   :` Physical setpoint<br>`XP   :` Physical actual value<br>`WN   :` Scaled setpoint<br>`OGP  :` Upper limit of the phys. value<br>`UGP  :` Lower limit of the phys. value<br>`FAKT :` Factor<br>`PAFE :` Parameter assignment errors |
| FB 250 | Read in and scale analog value | `     : JU FB 250`<br>`NAME : RLG:AE`<br>`BG   :` Slot number<br>`KNKT :` Channel number, channel type<br>`OGR  :` Upper limit of the output value<br>`UGR  :` Lower limit of the output value<br>`EINZ :` Irrelevant<br>`XA   :` Output value<br>`FB   :` Error bit<br>`BU   :` Range violation |
| FB 251 | Output analog value | `     : JU FB 251`<br>`NAME : RLG:AA`<br>`XE   :` Analog value to be output<br>`BG   :` Slot numbers<br>`KNKT :` Channel number, channel type<br>`OGR  :` Upper limit of the output value<br>`UGR  :` Lower limit of the output value<br>`FEH  :` Limit specification error<br>`BU   :` Input value exceeds UGR or OGR limiting values |

## A.3 Organization Blocks

**Table A-2. Organization Blocks**

| Name | Description |
|---|---|
| OB 1 | Cyclic scanning of the application program |
| OB 13 | Time-controlled processing of the application program |
| OB 21 | Restart OB:<br>OB 21 is executed once when the PLC goes from "STOP" to "RUN". |
| OB 22 | Restart OB:<br>On power-up (ON/OFF switch set to "1"), OB 22 is executed once if the programmable controller was in "RUN" mode prior to power-down and if there is no reason for a STOP on power-up. |
| OB 251 | PID algorithm: Is the basis of S5-100U closed-loop control.<br>The standard function blocks generate the environment for the required control structure. |

## A.4 Data Blocks

**Table A-3. Data Blocks**

| Name | Description |
|---|---|
| DBRE | Data block shared by all controllers. Represents the interface to the user. |
| DBOB | Internal, controller-specific data block. Serves organization block OB 251 as data interface. |
| DBF0 | Data block for standard function block FB 205<br>(Setpoint Sequence) |

**Table A-3. Data Blocks (Cont.)**

| Name | Description |
|---|---|
| DBP1/ DBP2 | Data blocks for standard function block FB 207 (Polygon) |
| DBVE | Data block for standard function block FB 209 (Deadtime Element) |
| DB01/ DB02 | Data block for standard function block FB 215 (Reset) |

# Index

# Index

**Siemens AG**
**AUT E1114B**
**Postfach 1963**
**Werner-von-Siemens-Str. 50**

**D-92209 Amberg**
**Fed. Rep. of Germany**

| | |
|---|---|
| Suggestions<br><br>Corrections | |
| **For Publication/Manual:** | |
| **Titel: Standard Function Blocks S5-100U Closed-Loop Control** | |
| **Order No.: 6ES5840-4BC21** | From:<br>Name ______ |
| **Release: 1** | Company/Dept. ______ |
| Should you come across any printing errors when reading this publication, please use this form to let us know. We would also welcome any suggestions you may have to improve the manual. | Address ______<br><br>Tel. |

Suggestions and or corrections**:**

## REMARKS FORM

Your comments and recommendations will help us to improve the quality and usefulness of our publications. Please take the first available opportunity to fill out this questionnaire and return it to Siemens.

Returning this form assures you of receiving all changes and updates to this document.

Please check the industry that applies to you:

- ☐ Automotive
- ☐ Chemical
- ☐ Electrical Machinery
- ☐ Food
- ☐ Instrument and Control
- ☐ Nonelectrical Machinery
- ☐ Petrochemical
- ☐ Pharmaceutical
- ☐ Plastic
- ☐ Pulp and Paper
- ☐ Textile
- ☐ Transportation
- ☐ Other____________

| | | | |
|---|---|---|---|
| 1. Is this document well organized? | | Yes | No |
| 2. Is the information easy to find? | | Yes | No |
| 3. Is the text easy to understand? | | Yes | No |
| 4. Is the document technically accurate? | | Yes | No |
| 5. Does the level of technical detail meet your requirements? | | Yes | No |
| 6. Would you like more illustrations? | | Yes | No |
| 7. Please rate the quality of the graphics: | Excellent | Good | |
| | Adequate | Poor | |

Additional Comments:

___________________________________________________________

___________________________________________________________

___________________________________________________________

___________________________________________________________

___________________________________________________________

*Manual: S5-100U Closed-Loop Control Order No.: 6ES5 840-4BC21 Edition 1*

Name: ______________________________________________

Title: ______________________________________________

Company Name: ______________________________________

Address: ____________________________________________

City: __________________ State: ________________ Zip: ____________