

Learn-/Training Document

Siemens Automation Cooperates with Education (SCE) | From Version V14 SP1

TIA Portal Module 031-500 Analog Values for SIMATIC S7-1200

siemens.com/sce



Matching SCE Trainer Packages for these Learn-/Training Document

- SIMATIC S7-1200 AC/DC/RELAY (set of 6) "TIA Portal" Order no.: 6ES7214-1BE30-4AB3
- SIMATIC S7-1200 DC/DC/DC (set of 6) "TIA Portal" Order no.: 6ES7214-1AE30-4AB3
- Upgrade SIMATIC STEP 7 BASIC V14 SP1 (for S7-1200) (set of 6) "TIA Portal" Order no.: 6ES7822-0AA04-4YE5

Please note that these trainer packages are replaced with successor packages when necessary. An overview of the currently available SCE packages is provided at: <u>siemens.com/sce/tp</u>

Continued training

For regional Siemens SCE continued training, please contact your regional SCE contact siemens.com/sce/contact

Additional information regarding SCE

siemens.com/sce

Information regarding use

The SCE Learn-/Training Document for the integrated automation solution Totally Integrated Automation (TIA) was prepared for the program "Siemens Automation Cooperates with Education (SCE)" specifically for training purposes for public educational facilities and R&D institutions. Siemens AG does not guarantee the contents.

This document is to be used only for initial training on Siemens products/systems, which means it can be copied in whole or part and given to those being trained for use within the scope of their training. Circulation or copying this Learn-/Training Document and sharing its content is permitted within public training and advanced training facilities for training purposes.

Exceptions require written consent from the Siemens AG contact person: Roland Scheuerer roland.scheuerer@siemens.com.

Offenders will be held liable. All rights including translation are reserved, particularly if a patent is granted or a utility model or design is registered.

Use for industrial customer courses is explicitly not permitted. We do not consent to commercial use of the Learn-/Training Document.

We wish to thank the TU Dresden, particularly Prof. Dr.-Ing. Leon Urbas and the Michael Dziallas Engineering Corporation and all other involved persons for their support during the preparation of this Learn-/Training Document.

Table of contents

1	Goa	۱	4
2	Prei	requisite	4
3	Req	uired hardware and software	5
4	The	ory	6
	4.1	Analog signals	6
	4.2	Measuring transducers	7
	4.3	Analog modules – A/D converter	7
	4.4	Data types of the SIMATIC S7-1200	8
	4.5	Reading/outputting analog values	9
	4.6	Normalizing analog values	11
5	Tas	k	12
6	Plar	nning	12
	6.1	Analog control of the conveyor speed	12
	6.2	Technology diagram	13
	6.3	Reference list	14
7	Stru	ctured step-by-step instructions	15
	7.1	Retrieve an existing project	15
	7.2	Create the "MOTOR_SPEEDCONTROL" function	17
	7.3	Configuration of the analog output channel	24
	7.4	Expand the tag table to include analog signals	25
	7.5	Call the block in the organization block	26
	7.6	Save and compile the program	29
	7.7	Download the program	30
	7.8	Monitor program blocks	31
	7.9	Archive the project	33
8	Che	cklist	34
9	Exe	rcise	35
	9.1	Task – Exercise	35
	9.2	Technology diagram	36
	9.3	Reference list	37
	9.4	Planning	37
	9.5	Checklist – Exercise	38
1() Add	itional information	39

Analog Values for SIMATIC S7-1200

1 Goal

In this chapter, you will become acquainted with the analog value processing of the SIMATIC S7-1200 with the TIA Portal programming tool.

The module explains the acquisition and processing of analog signals and gives a step-by-step description of read and write access to analog values in the SIMATIC S7-1200.

The SIMATIC S7 controllers listed in Chapter 3 can be used.

2 Prerequisite

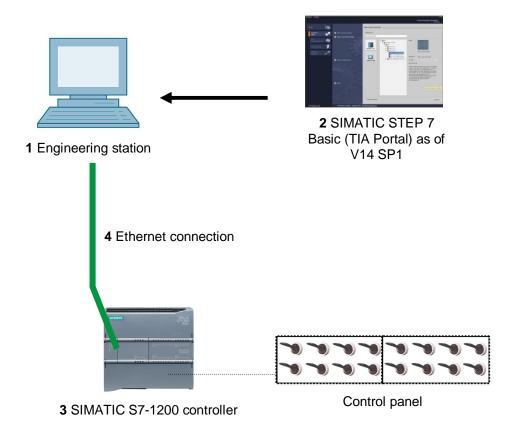
This chapter builds on the chapter IEC Timers and Counters with the SIMATIC S7 CPU1214C. You can use the following project for this chapter, for example: SCE_EN_031-300_IEC_Timers_Counters_S7-1200.zap14

3 Required hardware and software

- 1 Engineering station: requirements include hardware and operating system (for additional information, see Readme on the TIA Portal Installation DVDs)
- 2 SIMATIC STEP 7 Basic software in TIA Portal as of V14 SP1
- 3 SIMATIC S7-1200 controller, e.g. CPU 1214C DC/DC/DC with ANALOG OUTPUT SB1232 signal board, 1 AO – Firmware as of V4.2.1

Note: The digital inputs and analog inputs and outputs should be fed out to a control panel.

4 Ethernet connection between engineering station and controller



For unrestricted use in educational / R&D institutions. © Siemens AG 2018. All rights reserved. SCE_EN_031-500 Analog Values S7-1200_R1709.docx

4 Theory

4.1 Analog signals

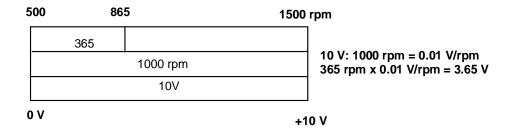
In contrast to a binary signal, which can assume only two signal states ("Voltage present +24 V" and "Voltage not present 0 V"), analog signals can assume any value within a defined range. A typical example of an analog sensor is a potentiometer. Depending on the position of the knob, any resistance can be set, up to the maximum value.

Examples of analog quantities in control engineering:

- Temperature -50 to +150 °C
- Flow rate 0 to 200 l/min
- Speed -500 to +50 rpm
- etc.

4.2 Measuring transducers

These quantities are converted to electrical voltages, currents or resistances with the help of a measuring transducer. If, for example, a speed is to be measured, the speed range of 500 to 1500 rpm can be converted to a voltage range of 0 to +10 V using a measuring transducer. At a measured speed of 865 rpm, the measuring transducer would output a voltage value of +3.65 V.

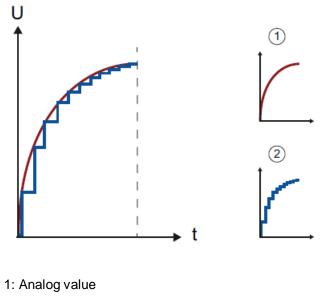


4.3 Analog modules – A/D converter

These electrical voltages, currents or resistances are then connected to an analog module that digitizes this signal for further processing in the PLC.

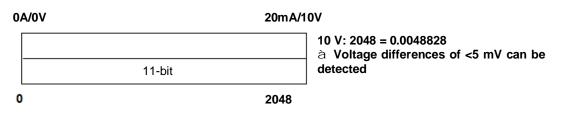
If analog quantities will be processed with a PLC, the read-in voltage, current or resistance value must be converted to digital information. The analog value is converted to a bit pattern. This conversion is referred to as analog-to-digital conversion (A/D conversion). This means, for example, that the voltage value of 3.65 V is stored as information in a series of binary digits.

The result of this conversion is always a 16-bit word for SIMATIC products. The integrated ADC (analog-to-digital converter) of the analog input module digitizes the analog signal being acquired and approximates its value in the form of a stepped curve. The most important parameters of an ADC are its resolution and conversion rate.



The more binary digits the digital representation uses, the finer the resolution is. For example, if only 1 bit was available for the voltage range of 0 to +10 V, you would only know whether the measured voltage is between 0 and +5 V or between +5 V and +10 V. With 2 bits, the range can be divided into 4 individual ranges, i.e., 0 to 2.5 / 2.5 to 5 / 5 to 7.5 / 7.5 to 10 V. Conventional A/D converters in control engineering use 8 bits, 11 bits or more for converting.

With 8 bits you have 256 individual ranges, while 11 bits provide a resolution of 2048 individual ranges.



4.4 Data types of the SIMATIC S7-1200

The SIMATIC S7-1200 has many different data types for representing different numerical formats. A list of some of the elementary data types is given below.

Data type	Size (bits)	Range	Example of constant entry
Bool	1	0 to 1	TRUE, FALSE, O, 1
Byte	8	16#00 to 16#FF	16#12, 16#AB
Word	16	16#0000 to 16#FFFF	16#ABCD, 16#0001
DWord	32	16#00000000 to 16#FFFFFFF	16#02468ACE
Char	8	16#00 to 16#FF	'A', 'r', '@'
Sint	8	-128 to 127	123,-123
Int	16	-32,768 to 32,767	123, -123
Dint	32	-2,147,483,648 to 2,147,483,647	123, -123
USInt	8	0 to 255	123
UInt	16	0 to 65,535	123
UDInt	32	0 to 4,294,967,295	123
Real	32	+/-1.18 x 10 ⁻³⁸ to +/-3.40 x 10 ³⁸	123.456, -3.4, -1.2E+12, 3.4E- 3
LReal	64	+/-2.23 x 10 $^{-308}$ to +/-1.79 x 10 308	12345.123456789 -1.2E+40
Time	32	T#-24d_20h_31 m_23s_648ms to T#24d_20h_31 m_23s_647ms Saved as: -2,147,483,648 ms to +2,147,483,647 ms	5#-2d
String	Variable	0 to 254 characters in byte size	'ABC'

Note: The **'INT'** and **'REAL'** data types play a large role in analog value processing. This is because read-in analog values exist as 16-bit integers in the **'INT'** format, and in order to ensure exact further processing only **'REAL'** floating-point numbers should be used due to rounding errors in the case of **'INT'**.

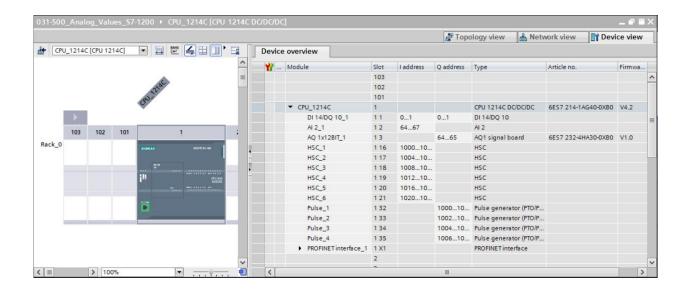
4.5 Reading/writing analog values

Analog values are read into the PLC or output from the PLC as word information. These words are accessed, for example, with the following operands:

%IW 64	Analog input word 64
%QW 64	Analog output word 64

Each analog value ("channel") occupies one input or output word. The format is 'Int', an integer.

The addressing of input and output words conforms to the addressing in the device overview. For example:

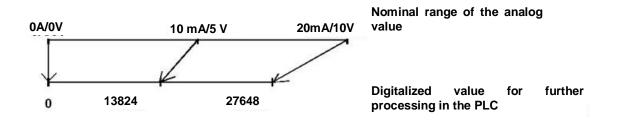


Here, the address of the first analog input would be %IW 64, and the address of the second analog input would be %IW 66.

The address of the analog output would be %QW 64.

The analog value transformation for further processing in the PLC is the same for analog inputs and analog outputs.

The digitized value ranges are as follows:



Often, these digitized values still have to be normalized by further processing them in the PLC in an appropriate manner.

4.6 Normalizing analog values

If an analog input value exists as a digitized value in the range +/- 27648, it must usually still be normalized so that the numerical values correspond to the physical quantities in the process.

Likewise, the analog output usually results from setting of a normalized value that then still has to be scaled to the output value +/- 27648.

In the TIA Portal, ready-made blocks or arithmetic operations are used for normalizing and scaling.

For this to be carried out as exactly as possible, the values for the normalizing must be converted to the REAL data type to minimize rounding errors.

5 Task

In this chapter, a function for analog control of the conveyor speed will be added to the program from chapter "SCE_EN_031-300 IEC Timers and Counters S7-1200".

6 Planning

The of the conveyor speed will be programmed analog control in the "MOTOR SPEEDCONTROL" [FC10] function as an expansion of the "SCE EN 031-300 IEC Timers and Counters S7-1200" project. This project must be retrieved from the archive in order to add this function. The "MOTOR SPEEDCONTROL" [FC10] function will be called in the "Main" [OB1]" organization block and wired. The control of the conveyor motor must be changed to -Q3 (conveyor motor -M1 variable speed).

6.1 Analog control of the conveyor speed

The speed will be set at an input of the "MOTOR_SPEEDCONTROL" [FC10] function in revolutions per minute (range: +/- 50 rpm). The data type is 32-bit floating-point number (Real).

First, the function will be checked for correct entry of the speed setpoint in the range +/- 50 rpm.

If the speed setpoint is outside the range +/- 50 rpm, the value 0 with data type 16-bit integer (Int) will be output at the output. The return value of the function (Ret_Val) will then be assigned the value TRUE (1).

If the speed setting is within the range +/-50 rpm, this value will first be normalized to the range 0...1 and then scaled to +/-27648 with data type 16-bit integer (Int) for output as the speed manipulated value at the analog output.

The output will then be connected with signal U1 (manipulated value speed of the motor in 2 directions +/- 10V corresponds to +/- 50 rpm).

6.2 Technology diagram

Here you see the technology diagram for the task.

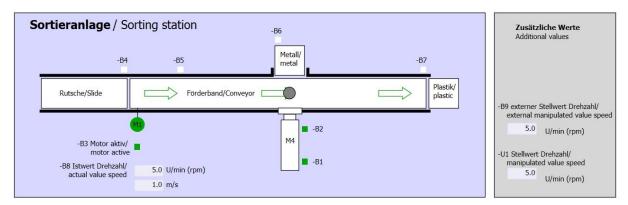


Figure 1: Technology diagram

Schalter der Sortieranlage Switches of sorting station	Automatikbetrieb Automatic mode	Handbetrieb / Manual mode -S3 Tippbetrieb -M1 vorwärts/ Manual -M1 forwards
-P1 enton -P4 aktiviet/vactive		 -S4 Tippbetrieb -M1 rückwärts/ Manual -M1 backwards -P7 ausgefahren/extended -S6 Zylinder -M4 ausfahren/ cylinder -M4 extend -S5 Zylinder -M4 einfahren/ cylinder -M4 retract

Figure 2: Control panel

6.3 Reference list

DI	Туре	Identifier	Function	NC/NO
I 0.0	BOOL	-A1	Return signal emergency stop OK	NC
I 0.1	BOOL	-K0	Main switch "ON"	NO
I 0.2	BOOL	-S0	-S0 Mode selector manual (0)/ automatic (1)	
				Auto = 1
I 0.3	BOOL	-S1	Pushbutton automatic start	NO
I 0.4	BOOL	-S2	Pushbutton automatic stop	NC
I 0.5	BOOL	-B1	Sensor cylinder -M4 retracted	NO
l 1.0	BOOL	-B4	Sensor part at slide	NO
l 1.3	BOOL	-B7	Sensor part at end of conveyor	NO

The following signals are required as global operands for this task.

DO	Туре	Identifier	Function	
Q 0.2	BOOL	-Q3	Conveyor motor -M1 variable speed	
QW 64	BOOL	-U1	Manipulated value speed of the motor in 2 directions +/- 10V corresponds to +/- 50 rpm	

Legend for reference list

- DI Digital Input DO Digital Output
- AI Analog Input AO Analog Output
- I Input

Q Output

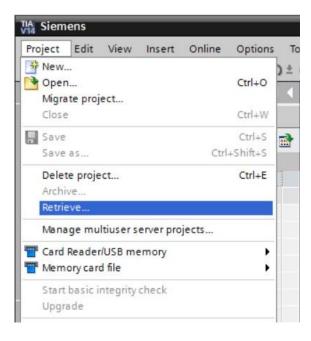
- NC Normally Closed
- NO Normally Open

7 Structured step-by-step instructions

You can find instructions on how to carry out planning below. If you already have a good understanding of everything, it will be sufficient to focus on the numbered steps. Otherwise, simply follow the detailed steps in the instructions.

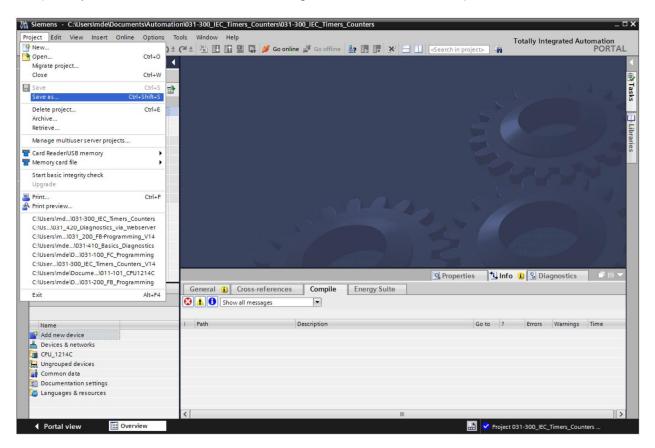
7.1 Retrieve an existing project

- Before we can expand the "SCE_EN_031-300_IEC_Timers_Counters_S7-1200.zap14" project from chapter "SCE_EN_031-300_IEC_Timers_Counters_S7-1200", we must retrieve this project from the archive. To retrieve an existing project that has been archived, you must select the relevant archive with
 Project
 Retrieve in the project view. Confirm your selection with Open.
 - (® Project ® Retrieve ® Select a .zap archive ® Open)



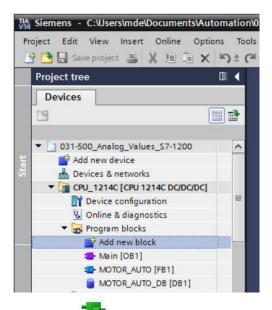
- ® The next step is to select the target directory where the retrieved project will be stored. Confirm your selection with "OK".
 - (® Target directory ® OK)

® Save the opened project under the name 031-500_Analog_Values_S7-1200.
 (® Project ® Save as ... ® 031-500_Analog_Values_S7-1200 ® Save)



7.2 Create the "MOTOR_SPEEDCONTROL" function

- Select the 'Program blocks' folder of your CPU_1214C and then click "Add new block" to
 create a new function there.
 - (® CPU_1214C [CPU 1214C DC/DC/DC] ® Add new block)



® Select fin the next dialog and rename your new block to: "MOTOR_SPEEDCONTROL". Set the language to FBD and manually assign the number "10". Select the "Add new and open" check box. Click "OK".

	Language:	FBD	-		
OB	Number:	10	•		
Organization		💽 Manual			
block		O Automatic			
FB	Description:				
Function block	Functions are o	ode blocks or subroutin	nes without dedic	ated memory.	
FC					
Function					
DB					
Data block					

® Create the local tags with their comments as shown here and change the data type of the 'Return' tag from 'Void' to 'Bool'. (® Bool)

			bla	DR_SPEEDCONTROL	Data tura	Default value	Comment
				Input	Data type	Default value	Comment
5				Setpoint_speed	Real		
2				Output	Redi		
1				Manipulated_variable_speed_AO	Int		
5				InOut	in.		
5				Temp			
7				Setpoint_speed_OK	Bool		
3				Manipulated_variable_speed_Norm	Real		
9	4			Constant			
10	-		•	Return			
11	-			MOTOR_SPEEDCONTROL	Bool		
					AOM_IDENT	~	
					Bool	=	
					Byte		
					CONN_ANY		
					CONN_OUC		
_		_			CONN_PRG		
2	5	2	= 1	[??] → -•I → -[=]	Char		
					DB_ANY	~	

Note: Be sure to use the correct data types.

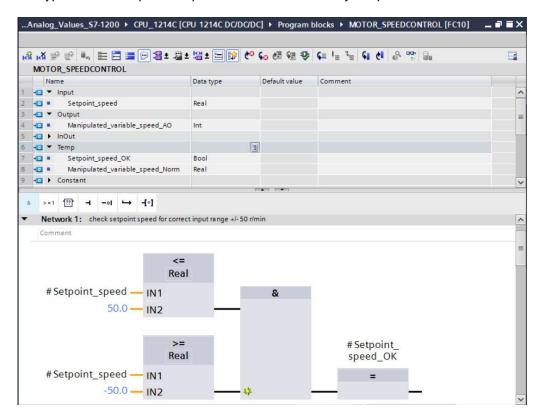
- Insert an Assignment ' ^{-[-]} in the first network and an 'And' * ' in front of it. Then use drag & drop to move the 'Comparator operation' 'Less or equal' from the 'Basic instructions' onto the first input of the * AND logic operation.
- ..C [CPU 1214C DC/DC/DC] + Program blocks + MOTOR_SPEEDCONTROL [FC10] 📃 🖬 🖬 🗙 Options init init ' 🗔 🛄 MOTOR_SPEEDCONTROL ✓ Favorites Name Data type Default value Comment & >=1 ?? - - - - - - → 🕣 🔻 Input ^ Setpoint_speed Real -[=] 🕣 🔻 Output ------Manipulated_variable_speed_AO Int InOut 🕣 🔻 Temp -0. Setpoint_speed_OK Bool Manipulated_variable_speed_Norm Real ✓ Basic instructions 🕣 🕨 Constant Name Description 10 🕣 🔻 Return General ~ NOTOR CREEDCONT Bit logic operati... < > III Timer operations & >=1 ??? - -ol → -[=] + H Counter operati... Comparator op... Block title: Speed control via analog output -E CMP == Equal Not equal CMP >= Greater or equal Network 1: check setpoint speed for correct input range +/- 50 r/min CMP <= Less or equal Comment E CMP> Greater than E CMP < Less than ~77 7~ & IN_Range Value within ra. <??.?>-= CUT_Range Value outside r. -IOKI-Check validity -INOT_OKI-Check invalidity Variant V
- (
 R I Basic instructions
 Comparator operations
 CMP<=)

R Next use drag & drop to move the 'Comparator operation' 'Greater or equal' onto the second input of the AND logic operation.

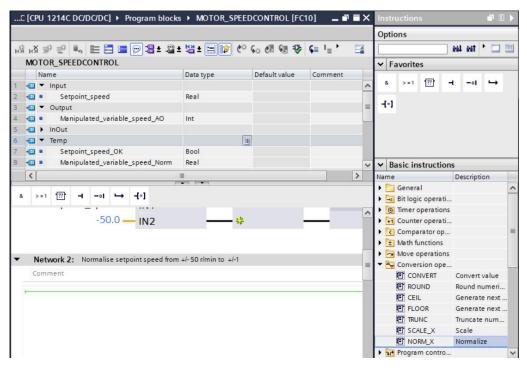
C [CPU 1214C DC/DC/DC] Program blocks						ructions		-
					Opti	ons		_
े 🖓 🕸 👘 💼 🔚 🔚 🗩 🖀 t 🖀 t	별± 드 😥 🕐	° 🕼 🕮 🥵 🤣	¢≡ '≡ '	-			ivit ivit ' 🗔	
MOTOR_SPEEDCONTROL					▼ F	avorites		
Name	Data type	Default value	Comment					
- Input				^	8	>=1 ???	∘I →	
Setpoint_speed	Real				-[-]			
Output				-	1.1			
Manipulated_variable_speed_AO	Int			-				
InOut								
- Temp								
Setpoint_speed_OK	Bool							
Manipulated_variable_speed_Norm	Real				✓ B	asic instructi	ons	
Constant					Name	,	Description	
- Return	Daal	_		~	۱ 🗀	General		
				>	• -	Bit logic opera	ti	
	and human				10	Timer operatio	ons	
>=1 [??] → -0 → -[=]					+ +1	Counter opera	ti	
Block title: Speed control via analog output				^	- <	Comparator o	p	
comment				-		E CMP ==	Equal	
Juniterit						🖭 CMP 🗇	Not equal	
Network 1: check setpoint speed for correct	t input range +/- 50	r/min				CMP >=	Greater or equ	Ja
Comment						CMP <=	Less or equal	
3647111116-115				=		CMP>	Greater than	
<=						CMP <	Less than	
777						IN_Range		а.
?? — IN1	8					OUT_Range		
?? — IN2	Contract of Contra	?.?				🗉 - ОК -	Check validity	
142		-					 Check invalidit 	ty
?.? -••		_				🛅 Variant		
					<			>

(
 Basic instructions
 Comparator operations
 CMP>=)

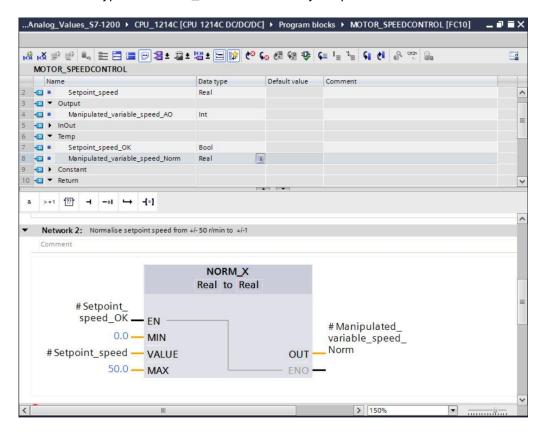
 Connect the contacts in Network 1 with the constants and local tags as shown here. The data types in the comparator operations are automatically adapted to 'Real'.



® Use drag & drop to move the 'Conversion operation' 'NORM_X' into Network 2 in order to normalize the speed setpoint of +/- 50 rpm to +/- 1. (® Basic instructions ® Conversion operations ® NORM_X)



® Connect the contacts in Network 2 with the constants and local tags as shown here. The data types in 'NORM_X' are automatically adapted to 'Real'.

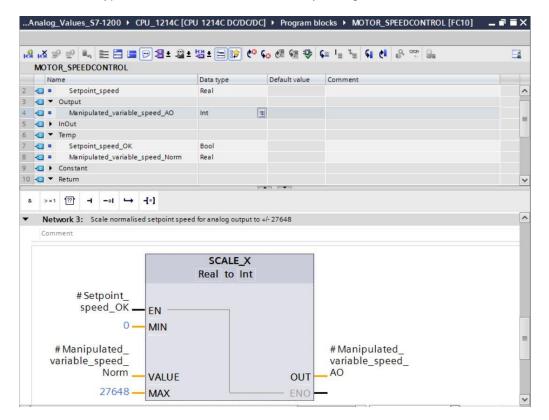


® Use drag & drop to move the 'Conversion operation' 'SCALE_X' into Network 3 in order to scale the speed setpoint from the normalized +/- 1 onto the range for the analog output +/- 27468.

C [CPU 1214C DC/DC/DC] Program blo	ocks MOTOR_SP	EEDCONTROL [FC1	0] 💶 🖬 i	×	Instructions 📑 🛙	
					Options	
ත් ශ් 🖻 🖆 🐛 🗮 🚍 🔛 🛢 🛨 .	🖀 ± 🖼 ± 🚍 😥 (t° 🕻 🖑 🖏 🖓	⊊ I _≡ *		🔲 ' tiù ki	
MOTOR_SPEEDCONTROL					✓ Favorites	
Name	Data type	Default value	Comment			
2 🚾 = Setpoint_speed	Real			~	& >=1 [??] → -0 →	
3 🕣 🔻 Output					11	
4 📹 = Manipulated_variable_speed_AO	Int			=	-{-1	
5 🕣 🕨 InOut						
6 🕣 🔻 Temp						
7 🕣 🔹 Setpoint_speed_OK	Bool					
8 🕣 🔹 Manipulated_variable_speed_Nor	m Real					
9 🕣 🕨 Constant				~	✓ Basic instructions	
<				>	Name Description	
# Setpoint_speed VALUE		OUT ENO	Variable Norm	e_ ^	O Timer operations O Timer operations O Toperations O Comparator op Math functions Move operations O Conversion ope	
 Network 3: Scale normalised setpoint: 	speed for analog outpu	t to +/- 27648			CONVERT Convert value	
Comment				-	ROUND Round numeri	-
Comment				-	E CEIL Generate next	
E				- 11	FLOOR Generate next	
					TRUNC Truncate num	•
				=	E SCALL_A Scale	
					NORM_X Normalize	
				-	Program contro	1
					<	•
					> Extended instructions	
				~	> Technology	
<	> 150%				> Communication	_

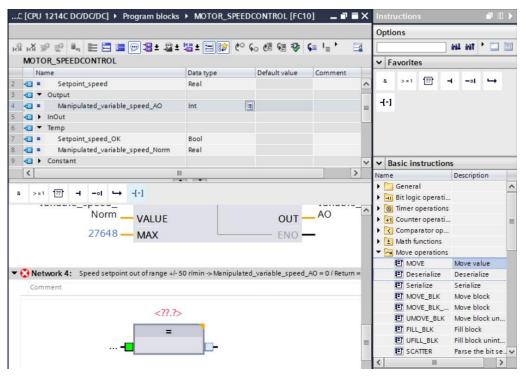
(
 Basic instructions
 Conversion operations
 SCALE_X)

® Connect the contacts with the constants and local tags in Network 3 as well, as shown here. The data types in 'SCALE_X' are automatically changed to 'Real' or 'Int'.

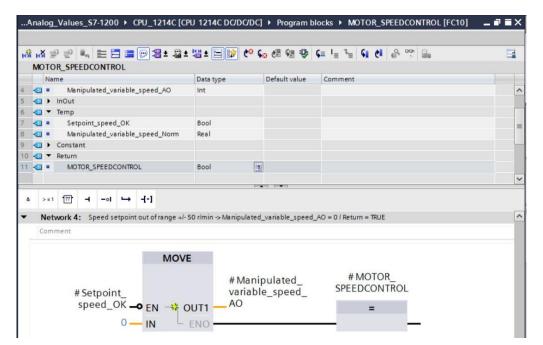


For unrestricted use in educational / R&D institutions. © Siemens AG 2018. All rights reserved. SCE_EN_031-500 Analog Values S7-1200_R1709.docx

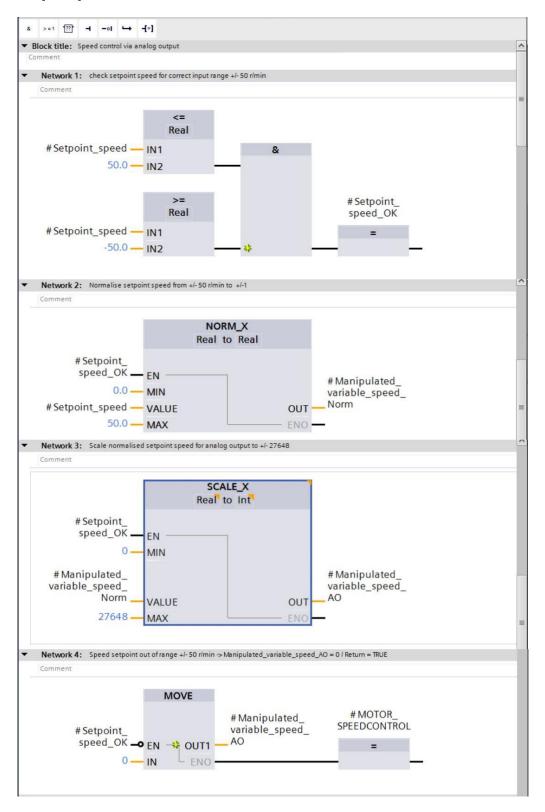
- Insert an Assignment ' f⁻¹' in the fourth network. Use drag & drop to move the 'Move' command from the 'Move operations' folder under 'Basic instructions' in front of the Assignment.
 - (
 R -[=]
 Basic instructions
 Move operations
 MOVE)



In the contacts in Network 4 will now be connected with constants and local tags as shown here. If the speed setpoint is not within the range +/- 50 rpm, the value '0' is output at the analog output and the value TRUE is assigned to the return value (Return) of the "MOTOR_SPEEDCONTROL" function.

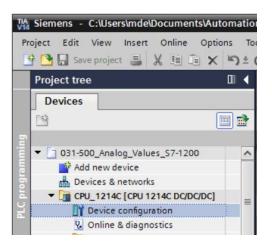


® Do not forget to click Save project. The finished function "MOTOR_SPEEDCONTROL" [FC10] in FBD is shown below.



7.3 Configuration of the analog output channel

® Double-click the 'Device configuration' to open it.

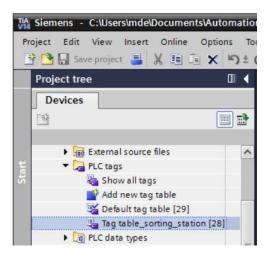


® Check the address setting and the configuration of the analog output channel 0. (® Q address: 64...65 ® Properties ® General ® Analog outputs ® Reaction to CPU STOP: Use substitute value ® Channel 0 ® Analog output type: Voltage ® Substitute value for channel on a change from RUN to STOP: 0.000 V ® Enable short circuit diagnostics)

		alues_\$7-1200\031-500_Analog_V	alues_\$7-12	00				-			
	Options Tools Window Help	🔄 💋 Go online 🖉 Go offline	17 UR UR	× = 🗆	-Search i	n project>	ally Integrated Autom P	oRT/			
031-500_Analog_Values_\$7-12	200 • CPU_1214C [CPU 1214C DO	CIDCIDC]					_	a = 1			
					F Topolog	gy view 🔒 Netwo	rk view Device	view			
H CPU_1214C [CPU 1214C]	• 🖽 🖭 🚄 🎞 🛄 • 🖼 🔲	Device overview									
- b			Slot	Laddress	Q address	Type	Article no.	Fir			
103 102 101	1 2 🔳		103			1.04-					
Rack_0			102								
5164	MENS BRUTE BY BRUTE		101								
	[CPU_1214C 	1			CPU 1214C DC/DC/DC	6ES7 214-1AG40-0XB0	V4.2			
10		DI 14/DQ 10_1	11	01	01	DI 14/DQ 10					
	CPU dw2 000000	AI 2_1	12	6467		AI 2					
		AQ 1x12BIT_1	13		6465	AQ1 signal board	6ES7 232-4HA30-0XB0	V1.0			
		HSC_1	116	100010		HSC					
	×	HSC_2	1 17	100410		HSC					
Ⅲ > 100%		<			0			>			
General IO tags Syste	Analog outputs										
Contraction and the second states of the second sta								_			
	Reaction to CPU STOP:	Use substitute value						•			
Analog outputs I/O addresses	Reaction to CPU STOP: [Use substitute value						•			
Analog outputs I/O addresses											
Analog outputs I/O addresses	Channel0Channel address: [QW64									
Analog outputs I/O addresses	Channel0Channel address: [Analog output type:	QW64 Voltage									
Analog outputs I/O addresses	Channel0 Channel address: Analog output type: Voltage range:	QW64									
Analog outputs I/O addresses	Channel0Channel address: [Analog output type:	QW64 Voltage +/-10 V									
Analog outputs I/O addresses	Channel0 Channel address: [Analog output type: Voltage range: [Substitute value for channel on a change from RUN to STOP: [QW64 Voltage +/-10 V						•			
Analog outputs I/O addresses	Channel ddfress: Channel address: Analog output type: Voltage range: Substitute value for channel on a change from RUN to STOP:	QW64 Voltage +-10 V 0.000				Fender auschgeliden		•			
Analog outputs I/O addresses	Channel ddress: Channel address: Analog output type: Voltage range: Substitute value for channel on a change from RUN to STOP:	QW64 Voltage +/-10 V 0.000 C Enable short circuit diagnostics				Fenster ausschneitiden		•			

7.4 Expand the tag table to include analog signals

® Double-click the 'Tag table_sorting station' to open it.



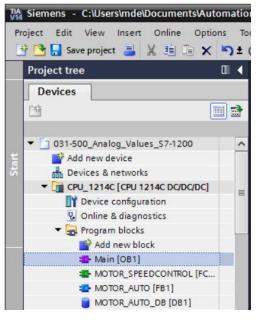
R Add the global tags for the analog value processing to the "Tag table_sorting station". An analog input B8 and an analog output U1 must be added.

(® U1 ® %QW64 ® B8 ® %IW64)

									🕣 Tags 🛛 🗉 User constant	s
ý	*	• 🕂 😤 🕯	h							
1	ag ta	ble_sorting_	station							
	N	lame	Data type	Address	Retain	Acces	Writa	Visibl	Comment	
11	-	-B6	Bool	%11.2					sensor part in front of cylinder -M4 (no)	1
12	-	-B7	Bool	%11.3					sensor part at end of conveyor (no)	
13	-	-53	Bool	%11.4					pushbutton manual mode conveyor –M1 forwards (no)	
14	-	-54	Bool	%11.5					pushbutton manual mode conveyor –M1 backwards (no)	
15		-55	Bool	%11.6					pushbutton manual mode cylinder -M4 retract (no)	
6	-	-56	Bool	%11.7					pushbutton manual mode cylinder -M4 extend (no)	
17	-	-Q1	Bool	%Q0.0					conveyor motor -M1 forwards fixed speed	
8	-	-Q2	Bool	%Q0.1					conveyor motor -M1 backwards fixed speed	
9	-	-Q3	Bool	%Q0.2					conveyor motor -M1 variable speed	
20	-	-M2	Bool	%Q0.3					cylinder -M4 retract	
21		-MB	Bool	%Q0.4					cylinder -M4 extend	
22	-	-P1	Bool	%Q0.5					display "main switch on"	
23	-	-P2	Bool	%Q0.6					display "manual mode"	1
24		-P3	Bool	%Q0.7					display "automatic mode"	
25	-	-P4	Bool	%Q1.0					display "emergency stop activated"	
26	-	-P5	Bool	%Q1.1					display "automatic mode started"	
27		-P6	Bool	%Q1.2					display cylinder -M4 "retracted"	
28	-	-P7	Bool	%Q1.3					display cylinder -M4 "extended"	
29	-	-U1	Int	%QW64					manipulated value speed in 2 directions +/- 10V	
30	-	-88	Int	%IW64					sensor actual value speed 010V	
31		<add new=""></add>		1						

7.5 Call the block in the organization block

® Open the "Main [OB1]" organization block with a double-click.



Add the temporary tag 'Motor_speed_monitoring_Ret_Val' to the local tags of OB1. These
 will be needed in order to interconnect the return value of the "MOTOR_SPEEDCONTROL"
 function.

(
 Temp
 Motor_speed_monitoring_Ret_Val
 Bool)

03	1-5	00	_Analog_Values_S7-1200 → CPU_	_1214C [CPU 12140	DC/DC/DC] • F	Program blocks 🔸 Main [OB1]	_∎≡×
				- 101 - 500 10			-
	Ma		0 2 4, E 🗄 🖻 🗩 岩 🛎	±≌±⊟⊮ €	' 60 (m 98 V		
		Na	me	Data type	Default value	Comment	
1	-	•	Input				
2			Initial_Call	Bool		Initial call of this OB	
3	-		Remanence	Bool		=True, if remanent data are available	
4		•	Temp				
5			Motor_speed_monitoring_Ret_Val	Bool]	Return value of FC10 MOTOR_SPEEDCONTROL	
6			<add new=""></add>				
7	-	•	Constant				
8			<add new=""></add>				

Select the block title of OB1 and then click '¹¹/₁₀' to insert a new Network 1 in front of the other networks ([®]/₁₀)

In						
		network				
-	Na	me	Data type	Default value	Comment	
	•	Input				
•	•	Initial_Call	Bool		Initial call of this OB	
•	•	Remanence	Bool		=True, if remanent data are available	
	•	Temp				
	•	Motor_speed_monitoring_Ret_Val	Bool		Return value of FC10 MOTOR_SPEEDCONTROL	
5		<add new=""></add>				
-0	•	Constant				
3		<add new=""></add>				
				-		
8	>=1	1??? → → → -1=]				
Ble	ock	title: "Main Program Sweep (Cycle)"				
Con	nmer	nt				
	Mad	work 1: Control conveyor motor forwar	de la consensate availe.			
		•	ds in automatic mode			
	Com	ment				
				%DB1		
				"MOTOR_AUTO DB"	_	
				%FB1		

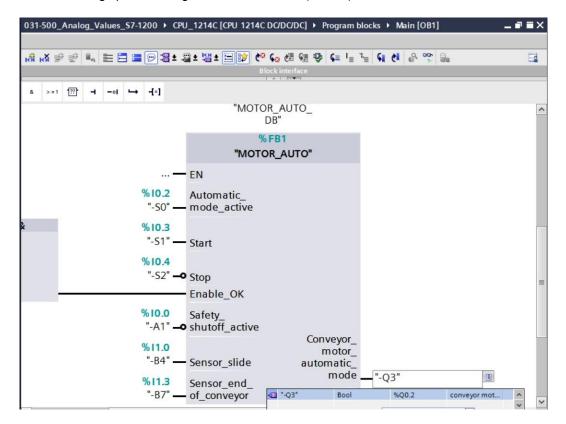
® Use drag & drop to move your "MOTOR_SPEEDCONTROL [FC10]" function onto the green line in Network 1.

TIA S	emens - C:\Users\mde\Documents\Aut	tomatio	n\0	31-500_	Analog_Values_S7-1200\031-	500_Anal	og_Values_S7-120	0		_ _ X
Proje	ect Edit View Insert Online Optio	ons To	ols	Windo	w Help				Totally Integrated A	utomation
营	💁 🛃 Save project 📕 🐰 🏥 🛅 🗙	₽)±	C		🗓 🏠 🖳 🞇 💋 Go online 👔	Go offi	ine 🎝 🖪 🖪	¥ 🗄 💷 🗠	earch in project>	PORTAL
P	roject tree		0	1-500	_Analog_Values_S7-1200 🕨	CPU_12	214C [CPU 1214C	DC/DC/DC] >	Program blocks 🔸 Main [OB1]	_ # = × 4
	Devices									
E,	3	•	13) = 1 = = = - 1	+ 2+	19 · E 10 / 10	6 ## 6# 40b		In s
-			- PC	Main				*0 (21 12) 🗸	→ = = → → → → → →	Instructions
1	031-500 Analog Values 57-1200	^		Na	me	l r)ata type	Default value	Comment	đ;
Ē.	Add new device		1	-			, and type	Deroon Toroc		3
B -	A Devices & networks		2	-0-	Initial_Call	E	lool		Initial call of this OB	
ž.	- CPU_1214C [CPU 1214C DC/DC/DC]	_	з		Remanence	E	lool		=True, if remanent data are available	8
ž	T Device configuration	=	4	-	Temp					2 Testing
	🛂 Online & diagnostics		5	•	Motor_speed_monitoring_Ret	t_Val E	Bool		Return value of FC10 MOTOR_SPEEDCONTROL	stin
	 Program blocks 		6		<add new=""></add>					ē
	Add new block		7	- 🕞	Constant					-
	🖀 Main [OB1]		8		<add new=""></add>					Tasks
	MOTOR_SPEEDCONTROL [FC	-								Tas
	MOTOR_AUTO [FB1]			>=1	1??? → → → → +[=]					ks
	Technology objects									
	External source files				title: "Main Program Sweep (Cyc	cle)"				<u>^</u>
	PLC tags		1	Comme	nt					Libraries
	PLC data types		-	Net	work 1: Speed monitoring com	veyor moto	or			Tie
	Watch and force tables			Com	ment					
<	m	>		Com						
	Details view	-	1	-	MOTOR_SP	EEDCONTR	IOL [FC10]			
-										
										~
										the second se

® Connect the contacts with the constants and global and local tags here as shown.

03	1-5	00	_Analog_Values_S7-120	0 → CPU_	1214C [CPU 12140	C DC/DC/DC] ▶ I	Program blocks 🔸 Main [OB1]	_ # # X
ň	ί κ	X 3	# # 🐛 🖿 🗖 🖉	🤊 📲 ± 🖉	± 🕲 ± 🖃 🎲 🦿	° 🐅 🖑 🖓 😵		
	Ma	ain						
		Na	ame		Data type	Default value	Comment	
		-	Input					^
	-		Initial_Call		Bool		Initial call of this OB	
			Remanence		Bool		=True, if remanent data are available	Ξ
į.		•	Temp					
5			Motor_speed_monitorin	g_Ret_Val	Bool		Return value of FC10 MOTOR_SPEEDCONTROL	
			<add new=""></add>					
	in the second	-	Constant					~
	<							>
				"MC	% FC10 TOR_SPEEDCC	NTROL"		=
			—	EN	м	Ret_Val anipulated_ variable_ speed_AO	%QW64	
				Setpoint	t speed	ENO		_
				berboun	-speed	LINO		
_								~
<			1	1			> 150%	

® Change the connection of output tag "Conveyor_motor_automatic_mode" in Network 2 to '-Q3' (Conveyor motor -M1 variable speed) so that the conveyor motor is controlled taking the analog speed setting into consideration. (® -Q3)



For unrestricted use in educational / R&D institutions. © Siemens AG 2018. All rights reserved. SCE_EN_031-500 Analog Values S7-1200_R1709.docx

7.6 Save and compile the program

To save your project, select the save project button in the menu. To compile all blocks, click the "Program blocks" folder and select the icon for compiling in the menu.
 ((e) Save project (e) Program blocks (e) (finite))

TA Siemens - C:\Users\mde\Documents\Automatio	n/031-500_Analog_Values_\$7-1200/031-500_Analog_Values_\$7-1200 _ 🗆 🗙
Project Edit View Insert Online Options To	ols Window Help Totally Integrated Automation
📑 📑 🔒 Save project 🚢 🐰 🏥 🚊 🗙 🍤 🛨	🖆 🗄 🗓 🛱 📓 🥵 💋 Go online 🖉 Go offline 🎄 🖪 📑 🔆 🔄 📋 <eret hin="" projects="" td="" 🖓<=""></eret>
Project tree 🔲 🕻	031-50(Compile J_Values_\$7-1200 + CPU_1214C [CPU 1214C DC/DC/DC] + Program blocks + Main [OB1]
Devices	
	Block interface
▼ 031-500_Analog_Values_S7-1200 ∧	
Add new device	a >=1 [??] → -0 ↦ -[=]
Devices & networks	▼ Block title: "Main Program Sweep (Cycle)"
▼ [] CPU_1214C [CPU 1214C DC/DC/DC]	Comment
Device configuration	Comment V Network 1: Speed monitoring conveyor motor
Conline & diagnostics	▼ Network 1: Speed monitoring conveyor motor
Program blocks	Comment
Add new block	
- Main [OB1]	% FC10
MOTOR_SPEEDCONTROL [FC	
MOTOR_AUTO [FB1]	"MOTOR_SPEEDCONTROL"
MOTOR_AUTO_DB [DB1]	
Technology objects	# Motor_speed_ monitoring_Ret_
External source files Department	monitoring_Ret_
La PLC tags La PLC data types	Ret_Val — Val
Watch and force tables	Manipulated
	variable %QW64
	— EN speed_AO — "-U1"
✓ Details view	
	15.0 — Setpoint_speed ENO —
Name Details	
Add new block	Network 2: Control conveyor motor forwards in automatic mode
📲 Main OB1	Comment
MOTOR_SPEEDCONTROL FC10	
MOTOR_AUTO FB1	
MOTOR_AUTO_DB DB1	% DB1
	< III > 150% · · · · · · · · · · · · · · · · · · ·
	🔍 Properties 🚺 Info 👔 😨 Diagnostics 💷 🖃
	General 3 Cross-references Compile Energy Suite Syntax
✓ Portal view Serview	Main (081)

® The "Info", "Compile" area shows which blocks were successfully compiled.

		S. P	roperties	fo 追 🖪 Dia	agnostics	
(General (1) Cross-references	Compile Energy Suite Syntax]			
Ο	🔒 📵 Show all messages		1.			
C	ompiling finished (errors: 0; warning	5:0)				
1	Path	Description	Go to ?	Errors	Warnings	Time
0	▼ CPU_1214C		7	0	0	1:27:45 PM
0	 Program blocks 		7	0	0	1:27:45 PM
0	MOTOR_SPEEDCONTRO	Block was successfully compiled.	~			1:27:45 PM
0	Main (OB1)	Block was successfully compiled.	Recht Alges A			1:27:49 PM
Ø		Compiling finished (errors: 0; warnings: 0)				1:27:49 PM
<		III				>

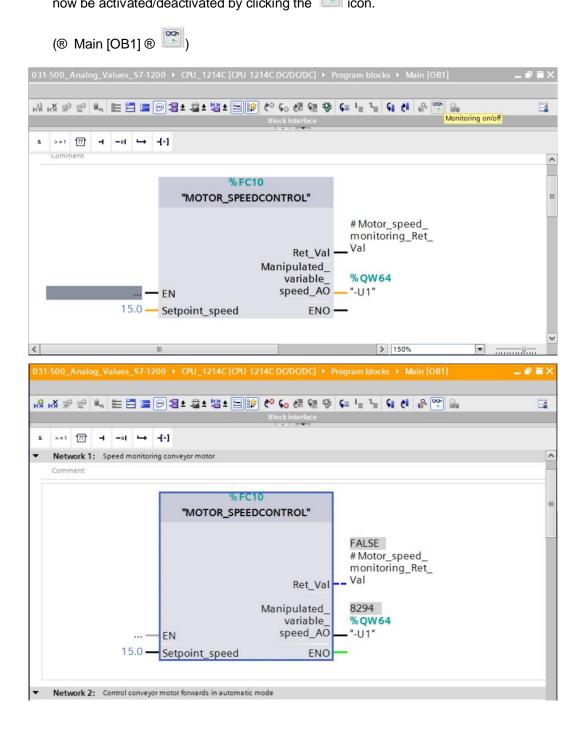
7.7 Download the program

 After successful compilation, the complete controller with the created program including the hardware configuration can, as described in the previous modules, be downloaded. (
 ID)

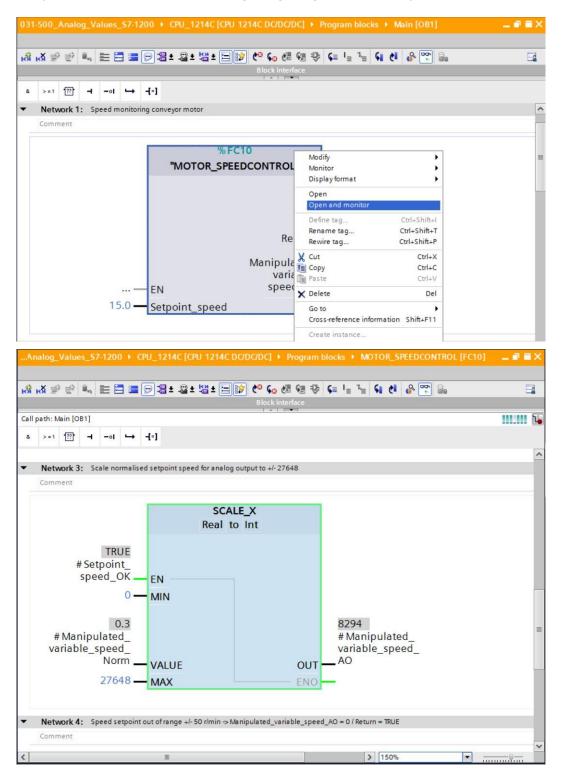
Siemens - C:\Users\mde\Documents\Automa	on\031-500_Analog_Values_S7-1200\031-500_Analog_Values_S7-1200	_ 0
Project Edit View Insert Online Options	and the second	Fotally Integrated Automation
	(* ± 🖥 🗓 🛐 🚆 🖓 Go online 🖉 Go offline 🎄 🖟 🖡 🗶 🖃 🚺 <rp>Search in project> 🙀</rp>	PORTAL
Project tree	031-500_Apalog_Values_S7-1200 CPU_1214C [CPU 1214C DC/DC/DC] Program blocks Mai	[OB1] _ I I X
Devices		
	成成 \$P\$ ● ■ ■ ■ ● 월 ± 월 ± 월 달 ● 6 6 6 6 6 9 9 6 1 1 1 6 6	· ••• 🔒 🔄
✓ 031-500_Analog_Values_S7-1200 ✓ Add new device	a >=1 [??]	
Devices & networks		
CPU_1214C [CPU 1214C DC/DC/DC]	 Block title: "Main Program Sweep (Cycle)" 	^
Device configuration	Comment	
Q. Online & diagnostics	Network 1: Speed monitoring conveyor motor	Ξ
 Program blocks 	Comment	
Add new block	Comment	
The Main [OB1]	0/ ====	
MOTOR_SPEEDCONTROL [FC	% FC10	
MOTOR_AUTO [FB1]	"MOTOR_SPEEDCONTROL"	
MOTOR_AUTO_DB [DB1]		
Technology objects	# Motor speed	L
External source files	monitoring R	[
PLC tags	Ret_Val — Val	-
LC data types		
Watch and force tables	Manipulated_	
< III >	variable_ %QW64	
✓ Details view	Speed AO	
Module		
P	Properties 1. Inf	Diagnostics
	General () Cross-references Compile Energy Suite Syntax	
Name Name Device configuration	Show all messages	
Online & diagnostics	Compiling finished (errors: 0; warnings: 0)	
Regram blocks	Path Description Go to ?	Errors Warnings Time
Technology objects	✓ CPU_1214C	0 0 1:27:45 PM
External source files	✓ ▼ Program blocks	0 0 1:27:45 PM
PLC tags	MOTOR_SPEEDCONTRO Block was successfully compiled.	1:27:45 PM
PLC data types	Main (OB1) Block was successfully compiled.	1:27:49 PM
Watch and force tables	Compiling finished (errors: 0; warnings: 0)	1:27:49 PM
🙀 Online backups		
Portal view Overview		ect 031-500_Analog_Values_S

7.8 Monitor program blocks

The desired block must be open for monitoring the downloaded program. The monitoring can
 now be activated/deactivated by clicking the icon.



 The "MOTOR_SPEEDCONTROL" [FC10] function called in the "Main [OB1]" organization block can be selected directly for "Open and monitor" after right-clicking and the program code in the function can thus be monitored.



(® "MOTOR_SPEEDCONTROL" [FC10] ® Open and monitor)

7.9 Archive the project

- As the final step, we want to archive the complete project. Select the
 'Archive ...' command in the
 'Project' menu. Select a folder where you want to archive your project and save it with the file type "TIA Portal project archive".
 - (® Project ® Archive ® TIA Portal project archive ® 031-500_Analog_Values_S7-1200....
 - ® Save)

🕌 Siemens - C:\Users\mdelDocuments\Automation\031-500_Analog_Values_\$7-1200\031-500_Analog_Values_\$7-1200 💶	×
Project Edit View Insert Online Options Totally Integrated Automation	
Migrate project Ctrl+O Ctrl+O Migrate project Ctrl+O O 31-500_Analog_Values_S7-1200 → CPU_1214C [CPU 1214C DC/DC/DC] → Program blocks → Main [OB1] _ II = II = X	•
🖬 Save Ctrl+S 🎰 🚓 KK # 관 🖬 듣 🚍 🚍 🥲 웹 ± 😫 ± 웹 ± 별할 산 ፍ₀ 셴 웹 및 🌾 🖬 🖓 🖓 🚇 📑	and Instructions
Save as Ctrl+Shift+S Block interface	truc
Delete project Ctrl+E	tion
Archive a >=1 (12) → -01 → -[•] Retrieve	S
Network 1: Speed monitoring conveyor mator	
Comment	E
Tead Reader/USB memory	⇒ Testing
Start basic integrity check	g
	-
CiUsersi1031-500_Analog_Values_57-1200	Tasks
C:UsersInd1031-300_IEC_Timers_Counters	sks
# Motor_speed_	
monitoring_Ret_	
C:UsersIndelU031-100_FC_Programming Ret_Val Val	ibr
C:lUserl031-300_IEC_Timers_Counters_V14	Libraries
C:UsersImdelDocume1011-101_CPU1214C Manipulated_ 8294	S
Exit Alt+F4 Variable %QW64	
V Details view EN Speed_AO -*-U1*	
15.0 Setpoint_speed ENO	
Name	
Add new device Vetwork 2: Control conveyor motor forwards in automatic mode	
Devices & networks Comment	
L GPU_1214C ☐ Ungrouped devices	
ing Common data	
To Documentation settings	
© Languages & resources % DB1 ✓	
Properties 🗓 Info 🗓 Diagnostics 🗖 🗆 🛆	
🖣 Portal view 🔚 Overview 🎂 Main (OB1)	

8 Checklist

No.	Description	Completed
1	Compiling successful and without error message	
2	Download successful and without error message	
3	Switch on station (-K0 = 1) Cylinder retracted / Feedback activated (-B1 = 1) EMERGENCY OFF (-A1 = 1) not activated AUTOMATIC mode (-S0 = 1) Pushbutton automatic stop not actuated (-S2 = 1) Briefly press the automatic start pushbutton (-S1 = 1) Sensor part at slide activated (-B4 = 1) then Conveyor motor -M1 variable speed (-Q3 = 1) switches on and stays on. The speed corresponds to the speed setpoint in the range +/- 50 rpm	
4	Sensor part at end of conveyor activated (-B7 = 1) \textcircled{B} -Q3 = 0 (after 2 seconds)	
5	Briefly press the automatic stop pushbutton (-S2 = 0) $\ensuremath{\mathbb{R}}$ -Q3 = 0	
6	Activate EMERGENCY OFF (-A1 = 0) ® -Q3 = 0	
7	Manual mode (-S0 = 0) ® -Q3 = 0	
8	Switch off station (-K0 = 0) $\ensuremath{\mathbb{B}}$ -Q3 = 0	
9	Cylinder not retracted (-B1 = 0) \circledast -Q3 = 0	
10	Project successfully archived	

9 Exercise

9.1 Task – Exercise

In this exercise a "MOTOR_SPEEDMONITORING" [FC11] function will be created additionally.

The actual value will be made available to B8 (Sensor actual value speed of the motor +/-10V corresponds to +/- 50 rpm) as an analog value and queried at an input of the "MOTOR_SPEEDMONITORING" [FC11] function. The data type is 16-bit integer (Int).

This actual speed value will first be normalized to the range +/- 1 as 32-bit floating-point number (Real) in the function.

The normalized actual speed value will then be scaled to revolutions per minute (range: +/- 50 rpm) as 32-bit floating-point number (Real) and made available at an output.

The following 4 limit values can be specified as 32-bit floating-point numbers (Real) at the block inputs in order to monitor them in the function:

Speed > Motor_speed_monitoring_error_max

Speed > Motor_speed_monitoring_warning_max

Speed < Motor_speed_monitoring_warning_min

Speed < Motor_speed_monitoring_error_min

If a limit value is exceeded or fallen below, the value TRUE (1) is assigned to the corresponding output bit.

If a fault is present, the protective tripping of the "MOTOR_AUTO" [FB1] function block will be tripped.

9.2 Technology diagram

Here you see the technology diagram for the task.

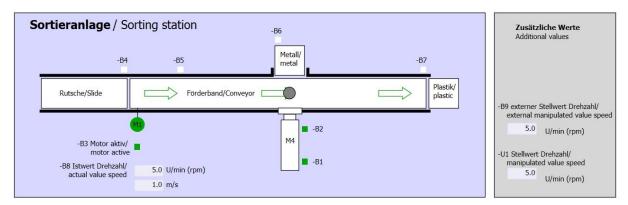


Figure 3: Technology diagram

Schalter der Sortieranlage Switches of sorting station	Automatikbetrieb Automatic mode	Handbetrieb / Manual mode -S3 Tippbetrieb -M1 vorwärts/ Manual -M1 forwards
-P1 enton -P4 aktiviet/vactive		 -S4 Tippbetrieb -M1 rückwärts/ Manual -M1 backwards -P7 ausgefahren/extended -S6 Zylinder -M4 ausfahren/ cylinder -M4 extend -S5 Zylinder -M4 einfahren/ cylinder -M4 retract

Figure 4: Control panel

9.3 Reference list

DI	Туре	Identifier	Function	NC/NO
I 0.0	BOOL	-A1	Return signal emergency stop OK	NC
I 0.1	BOOL	-K0	Main switch "ON"	NO
10.2	BOOL	-S0	Mode selector manual (0)/ automatic (1)	Manual = 0 Auto = 1
I 0.3	BOOL	-S1	Pushbutton automatic start	NO
I 0.4	BOOL	-S2	Pushbutton automatic stop	NC
I 0.5	BOOL	-B1	Sensor cylinder -M4 retracted	NO
l 1.0	BOOL	-B4	Sensor part at slide	NO
l 1.3	BOOL	-B7	Sensor part at end of conveyor	NO
IW64	BOOL	-B8	Sensor actual value speed of the motor +/-10V corresponds to +/- 50 rpm	

The following signals are required as global operands for this task.

DO	Туре	Identifier	Function	
Q 0.2	BOOL	-Q3	Conveyor motor -M1 variable speed	
QW 64	BOOL	-U1	Manipulated value speed of the motor in 2 directions +/- 10V corresponds to +/- 50 rpm	

Legend for reference list

- DI Digital Input DO Digital Output
- AI Analog Input AO Analog Output
- I Input Q Output
- NC Normally Closed
- NO Normally Open

9.4 Planning

Plan the implementation of the task on your own.

9.5 Checklist – Exercise

No.	Description	Completed
1	Compiling successful and without error message	
2	Download successful and without error message	
3	Switch on station (-K0 = 1) Cylinder retracted / Feedback activated (-B1 = 1) EMERGENCY OFF (-A1 = 1) not activated AUTOMATIC mode (-S0 = 1) Pushbutton automatic stop not actuated (-S2 = 1) Briefly press the automatic start pushbutton (-S1 = 1) Sensor part at slide activated (-B4 = 1) then Conveyor motor M1 variable speed (-Q3 = 1) switches on and stays on. The speed corresponds to the speed setpoint in the range +/- 50 rpm	
4	Sensor part at end of conveyor activated (-B7 = 1) $\ensuremath{\mathbb{B}}$ -Q3 = 0 (after 2 seconds)	
5	Briefly press the automatic stop pushbutton (-S2 = 0) \circledast -Q3 = 0	
6	Activate EMERGENCY OFF (-A1 = 0) ® -Q3 = 0	
7	Manual mode (-S0 = 0) ® -Q3 = 0	
8	Switch off station (-K0 = 0) \circledast -Q3 = 0	
9	Cylinder not retracted (-B1 = 0) \textcircled{B} -Q3 = 0	
10	Speed > Motor_speed_monitoring_error_max	
11	Speed < Motor_speed_monitoring_error_min	
12	Project successfully archived	

10 Additional information

More information for further practice and consolidation is available as orientation, for example: Getting Started, videos, tutorials, apps, manuals, programming guidelines and trial software / firmware, under the following link:

www.siemens.com/sce/s7-1200

Preview "Additional information"

- Getting Started, Videos, Tutorials, Apps, Manuals, Trial-SW/Firmware
 - ↗ TIA Portal Videos
 - ↗ TIA Portal Tutorial Center
 - > Getting Started
 - ↗ Programming Guideline
 - ↗ Easy Entry in SIMATIC S7-1200
 - > Download Trial Software/Firmware
 - ↗ Technical Documentation SIMATIC Controller
 - ↗ Industry Online Support App
 - TIA Portal, SIMATIC S7-1200/1500 Overview
 - ↗ TIA Portal Website
 - ↗ SIMATIC S7-1200 Website
 - ↗ SIMATIC S7-1500 Website

Further Information

Siemens Automation Cooperates with Education siemens.com/sce

SCE Learn-/Training Documents siemens.com/sce/documents

SCE Trainer Packages siemens.com/sce/tp

SCE Contact Partners siemens.com/sce/contact

Digital Enterprise siemens.com/digital-enterprise

Industrie 4.0 siemens.com/future-of-manufacturing

Totally Integrated Automation (TIA) siemens.com/tia

TIA Portal siemens.com/tia-portal

SIMATIC Controller siemens.com/controller

SIMATIC Technical Documentation siemens.com/simatic-docu

Industry Online Support support.industry.siemens.com

Product catalogue and online ordering system Industry Mall **mall.industry.siemens.com**

Siemens AG Digital Factory P.O. Box 4848 90026 Nuremberg Germany

Subject to change and errors © Siemens AG 2018

siemens.com/sce