SIEMENS

SIMATIC

SM331; AI 8 x 12 Bit

Getting Started
Part2: Voltage and PT100

Edition 04/2004

A5E00264161-01

Safety-related Guidelines

This manual contains notices intended to ensure personal safety, as well as to protect the products and connected equipment against damage. These notices are highlighted by the symbols shown below and graded according to severity by the following texts:

Danger
indicates that death, severe personal injury or substantial property damage will result if proper precautions are not taken.

Warning
indicates that death, severe personal injury, or substantial property damage can result if proper precautions are not taken.

Caution
indicates that minor personal injury or property damage can result if proper precautions are not taken.

Note
contains important information about the product, its operation or a part of the document to which special attention is drawn.

Qualified personnel

A device/system may only be commissioned or operated by **qualified personnel**. Qualified personnel as referred to in the safety guidelines in this document are persons authorized to energize, de-energize, clear, ground, and tag circuits, equipment and systems in accordance with established safety practice.

Correct Usage

Please observe the following:

Warning
The equipment/system or the system components may only be used for the applications described in the catalog or the technical description, and only in combination with the equipment, components, and devices of other manufacturers as far as this is recommended or permitted by Siemens.
The product will function correctly and safely only if it is transported, stored, set up, and installed as intended, and operated and maintained with care.

Registered Trademarks

SIMATIC®, SIMATIC HMI® and SIMATIC NET® are registered trademarks of Siemens AG.

Third parties using for their own purposes any other names in this document which refer to trademarks might infringe upon the rights of the trademark owners.

Siemens AG
Bereich Automation and Drives
Geschäftsgebiet Industrial Automation Systems
Postfach 4848, D- 90327 Nürnberg

Disclaimer of Liability

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcomed.

Table of contents:

1 Preface

Purpose of the Getting Started

The Getting Started gives you a complete overview of the commissioning of the analog module SM331. It assists you in the installation and parameterisation of the hardware of a voltage transducer and a resistance thermometer PT100. Furthermore, you will get an introduction to the configuration of the analog module with SIMATIC S7 Manager.

The target audience of the Getting Started is a beginner with limited experience in configuration, commissioning and servicing automation systems.

What to expect

Step by step the procedures from mounting the module to storing analog values in the STEP7 user program are explained in detail by means of an example. You will be guided through the following sections:

- Problem analysis
- Mechanical setup of the sample station
- Electrical connection of the sample station with conventional wiring
- Electrical wiring of the sample station with system wiring SIMATIC TOP connect
- Configuration with SIMATIC Manager
- Creating a small user program with STEP 7 which stores the read analog value in a data block
- Triggering and interpreting diagnostic and hardware interrupts.

2 Requirements

2.1 Required basic knowledge

No special knowledge in the area of automation technique is required to understand this description. As the configuration of the analog module is done with the software STEP 7, knowledge in how to use STEP 7 would be advantageous.

Further information on STEP 7 can be found in the electronic manuals that were delivered with STEP 7.

Knowledge of PC or similar computer devices (e.g. programming devices) using the operating system Windows 95/98/2000/NT or XP is assumed.

2.2 Required Hardware and Software

The scope of delivery of the analog module consists of two parts:

- Module SM331
- Front connector, which enables it to comfortably connect the power supply and the data connections.

Table 2-1 Components of the analog module

Quantity	Article	Order number
1	SM 331, OPTICALLY ISOLATED 8 AE, ALARM DIAGNOSTICS	6ES7331-7KF02-0AB0
1	20-pin front Connector with Spring contacts Alternatively: - 20-pin FRONT CONNECTOR WITH SCREW CONTACTS - FRONTSIDE PLUG-IN MODULE WITH CONNECTION FOR TWISTED RIBBON CABLE (System wiring TOP connect)	6ES7392-1BJ00-0AA0 6ES7392-1AJ00-0AA0 6ES7921-3AF00-0AA0
1	SIMATIC S7 SHIELD CONNECTING ELEMENT	6ES7390-5AA00-0AA0
2	SIMATIC S7, , TERMINAL ELEMENT F. 1 CABLE W. 4 TO 13MM IN DIA.	6ES7390-5CA00-0AA0

Additional general SIMATIC components required for the example are as follows:

Table 2-2 SIMATIC components of the sample station

Quantity	Article	Order number
1	POWER SUPPLY. PS 307 AC 120/230V, DC 24V, 5A (incl. power supply bridge)	6ES7307-1EA00-0AA0
1	CPU 315-2DP	6ES7315-2AG10-0AB0
1	MICRO MEMORY CARD, NFLASH, 128KBYTE	6ES7953-8LG00-0AA0
1	SIMATIC S7-300, Rail L=530MM	6ES7390-1AF30-0AA0
1	Programming device (PG) with MPI-interface and MPI cable PC with corresponding interface card	Depending on the configuration

If you want to set up the sample station with SIMATIC TOP connect, additional components as follows are required:

Table 2-3 SIMATIC Top connect components

Quantity	Article	Order number
1	FRONTSIDE PLUG-IN MODULE WITH CONNECTION FOR TWISTED RIBBON CABLE FOR ANALOG MODULES OF THE S7-300, POWER SUPPLY VIA SPRING TERMINALS	6ES7921-3AF00-0AA0
2	TERMINAL BLOCK TPA, 3 ROWS FOR ANALOG MODULES OF THE S7, CONN. WITH TWISTED RIBBON CABLE CONNECTION VIA SPRING TERMINALS 1 PCK = 1 PIECE	6ES7924-0CC00-0AB0
2	SHIELD SHEET FOR TERMINAL BLOCK ANALOG	6ES7928-1BA00-0AA0
4	CONNECTOR (FLAT SOCKET) ACC. TO DIN 41652, 16-PIN INSULATION PIERCING CONNECTING DEVICE	6ES7921-3BE10-0AA0
2	SIMATIC S7, TERMINAL ELEMENT F. 1 CABLE W. 4 TO 13MM IN DIA	6ES7390-5CA00-0AA0
2	SIMATIC S7, TERMINAL ELEMENT F. 2 CABLES W. 2 TO 6MM IN DIA	6ES7390-5AB00-0AA0
1	TWISTED RIBBON CABLE WITH 16 CORES 0.14 MM2 LENGTH: 30 M, SHIELDED	6ES7923-0CD00-0BA0

Table 2-4 Software STEP 7

Quantity	Article	Order number
1	STEP 7 Software version 5.2 or later installed on the programming device.	6ES7810-4CC06-0YX0

The following voltage transducers and resistance sensors can be used for the acquisition of analog signals:

Table 2-5 Resistance sensor and voltage transducer

Quantity	Article	Order number
1	±5V Voltage transducer	Depending on the manufacturer
3	PT100 Standard	Depending on the manufacturer

Note

This „Getting Started“ describes only the application of voltage transducers and resistance thermometers PT100 standard. If you want to use other transducers then you have to wire and parameterise the SM331 differently.

Furthermore, the following tools and materials are necessary:

Table 2-6 General tools and materials

Quantity	Article	Order number
Multiple	M6-bolts and nuts (Length depending on the mounting place)	standard
1	Screwdriver with blade width 3,5 mm	standard
1	Screwdriver with blade width 4,5 mm	standard
1	Wire cutting pliers and tools for stripping	standard
1	Tool to mount the cable end sleeve	standard
X m	Wire for grounding the rail, 10 mm^2 diameter. Ring terminal with 6,5 mm hole, length according to local conditions.	standard
X m	Flexible wire with 1mm^2 diameter with fitting wire end sleeves, Form A in 3 different colours – blue, red and green	standard
X m	3-wire power cord (AC 230/120V) with protective contact socket, length according to local condi-tions.	standard
1	Calibration device (Measuring instrument for commissioning, that can measure and supply current)	Depending on the manufacturer

3 Task

The Getting Started will guide you successfully through a sample application in which the following four sensors are installed:

- A pressure sensor which is connected to a voltage transducer (±5V).
- Three resistance thermometer type PT100

You will trigger failure diagnostics and hardware interrupts. You have the analog input module SM331, AI8x12 Bit (Order number 6ES7 331-7KF02-0AB0) available .

The module can trigger diagnostics alarms and hardware interrupts. It can process up to 8 analog inputs. Each module is configurable to different measuring modes (e.g. current measurement; voltage measurement; PT 100; thermo couple).

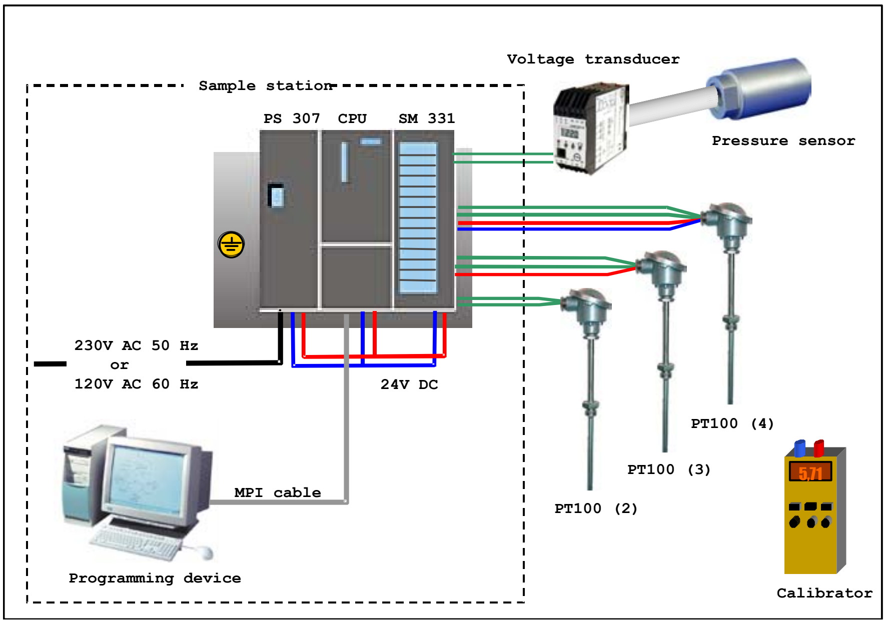


Figure 3-1 Components of the sample station

You will be guided through these steps

- Mechanical setup of the sample station (see chapter 4)
    - General mounting instructions for S7-300 modules
    - Configuration of the SM331 for the two selected measurement transducer types
- Electrical wiring of the sample station (see chapter 5)
    - Wiring of the power supply and the CPU
    - Setup of the analog module the conventional way
    - Setup of the analog module with system cabling SIMATIC TOP connect
- Configuration with SIMATIC Manager (see chapter 6)
    - Usage of project wizard
    - Completing the automatically generated hardware configuration
    - Integration of the supplied user program source
- User program testing (see chapter 7)
    - Interpreting the read values
    - Converting the measured values into readable analog values
- Utilizing the diagnostic capabilities of the SM331 module (see chapter 8)
    - Triggering a diagnostic interrupt
    - Analyzing the diagnostics
- Application of hardware interrupts (see chapter 9)
    - Parameterisation of hardware interrupts
    - Configuration and analysis of hardware interrupts

4 Mechanical setup of the sample station

The setup of the sample station is divided into two steps. First the setup of the power supply and the CPU is explained. After becoming acquainted with the analog module SM331, its mounting is described.

4.1 Mounting the sample station

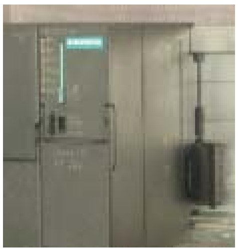
Before you can install the analog input module SM331, you need a basic setup of general SIMATIC S7-300 components.

The mounting takes place from left to right as follows:

- Power supply PS307
- CPU 315-2DP
- Analog module SM331

Table 4-1 Mounting the sample station (without SM331)

Step	Graphics	Description
1		Bolt together the rail to the base (screw size: M6) so that at least 40 mm space remains above and below the rail. If the base is a grounded sheet metal or a grounded mounting plate, ensure that the rail and the base are con-nected together with low resistance. Connect the rail with the protective ground wire. An M6 size bolt for grounding is available for this purpose.
2		Mounting of the power supply: • Hang the power supply to the top end of the rail
3		• and tighten it to the rail underneath

Step	Graphics	Description
4		Connect the bus connector (delivered with SM331) to the **left** connector on the back of the CPU
5		Mounting of the CPU: • Hang the CPU to the top end of the rail • Push it all the way left to the power supply • push it down • and tighten the screw to the rail underneath

4.2 Mounting the analog module

Before the mounting of the SM331 the measuring range modules should be plugged in (see chapter 4.2.4).

In this section you will learn:

- Which components you need
- What the properties of the analog input module are
- What a measuring range module is and how it is setup
- How to mount the already setup module

4.2.1 Components of SM331 with conventional connection plug

A functional analog module consists of the following components:

- Module SM331 (in our example 6ES7331-7KF02-0AB0)
- 20-pin front connector. There are two types of front connectors:
  - With spring contacts (Order number 6ES7392-1BJ00-0AA0)
  - With screw contacts (Order number 6ES7392-1AJ00-0AA0)

Figure 4-1 Components of SM331

4.2.2 SM331 with system wiring SIMATIC TOP connect

The system wiring SIMATIC TOP connect for the SM331 module consists of the following components:

- Front connector module (Order number 6ES7921-3AF00-0AA0)
- Terminal block TPA (Order number 6ES7924-0CC00-0AB0)
- Multiple small parts (see. Table 2-3)

Figure 4-2 Components of the SM331 with system wiring SIMATIC TOP connect

4.2.3 Properties of the analog module

The module is a universal analog module designed for the most commonly used applications.

The desired measuring mode should be set up directly on the module with the measuring range modules (see chapter 4.2.4).

- 8 inputs in 4 channel groups (each group with two inputs of the same type)
- Measurement resolution adjustable for each channel group
- User defined measuring mode per channel group:
  - Voltage
  - Current
  - Resistance
  - Temperature
- Configurable diagnostic interrupt
- Two channels with limit value interrupts (only channel 0 and channel 2 are configurable)
- Electrically isolated against back pane bus
- Electrically isolated against load voltage (exception: At least one module is set to position D)

Scope of delivery of the module SM331 (Order number 6ES7331-7KF02-0AB0):

Table 4-2 Scope of delivery of the module SM331

Components
Analog module SM331
Labelling strips
Bus connector
2 cable ties (not in the picture) to tie the external wiring

4.2.4 Measuring range modules

The module SM331 has four measuring range modules (one per channel group). The measuring range modules can be set to 4 different positions (A, B, C or D). With the set position you determine which transducer can be connected to the respective channel group.

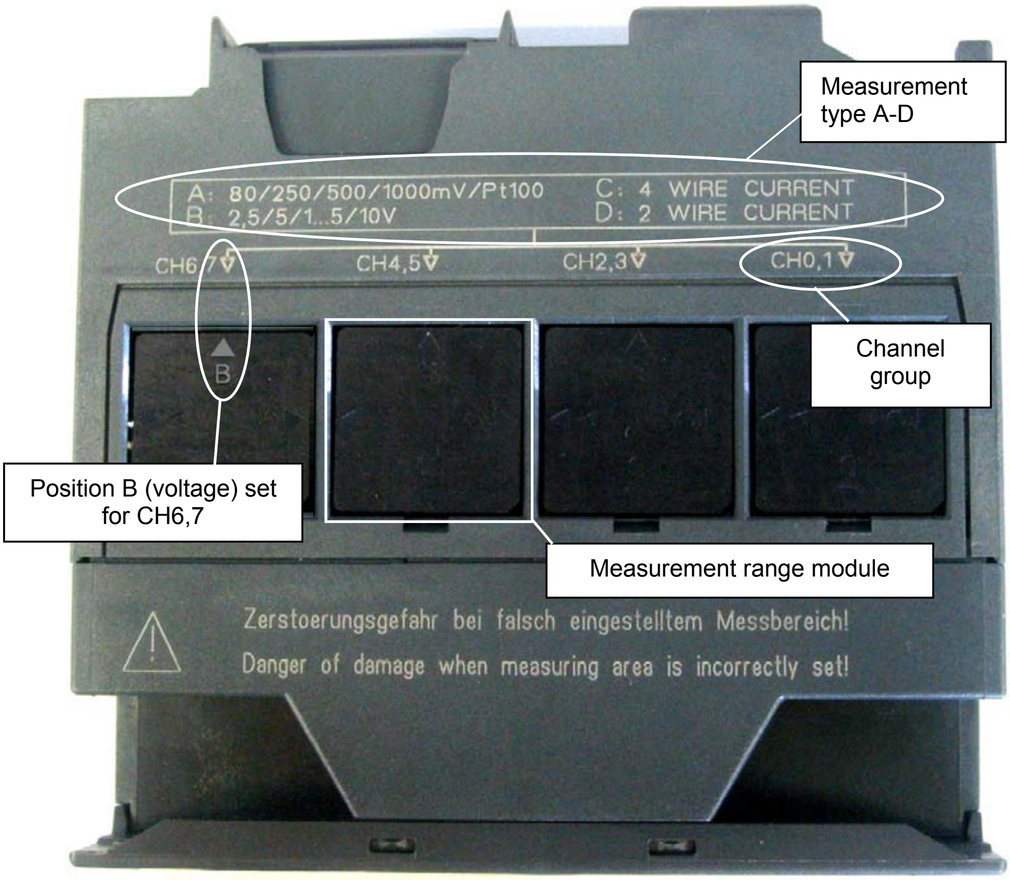


Figure 4-3 4 measuring range modules with default setting B (Voltage)

Table 4-3 Possible positions of the measuring range modules

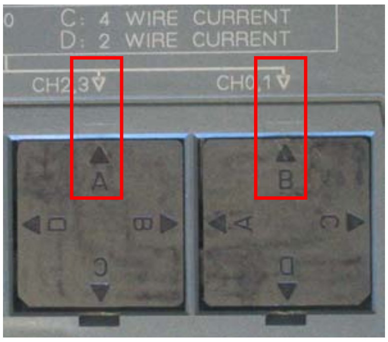
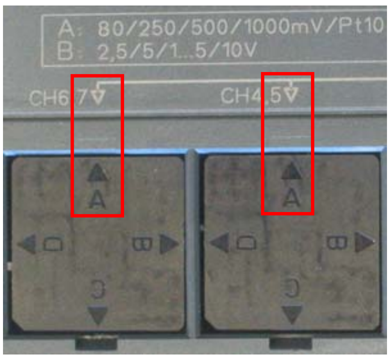
Position	Measurement type
A	Thermo couple / Resistance measurement
B	Voltage(default setting)
C	Current (4-wire transducer)
D	Current (2-wire transducer)

In our sample, input 0 of channel group CH0,1 is connected to a ±5V voltage transducer..

For connecting the three resistance thermometers type PT100 you need a complete channel group for each PT100 (CH2,3 / CH4,5 / CH 6,7).

The first measuring range module of the channel group CH0,1 should keep Position B (default setting) and the other modules should be changed to Position A.

Table 4-4 Positioning of the measuring range modules

Step	Graphics	Description
1		With a screwdriver, pull out the two measuring range modules.
2		Turn the measuring range modules to the desired position.
3		Plug the measuring range modules back into the module In our example, the module should have the following positions: CH0,1: B CH2,3: A CH4,5: A CH6,7: A

4.2.5 Mounting the SM331 module

After you have prepared the analog module accordingly, mount it to the rail as well.

Table 4-5 Mounting the SM331 module

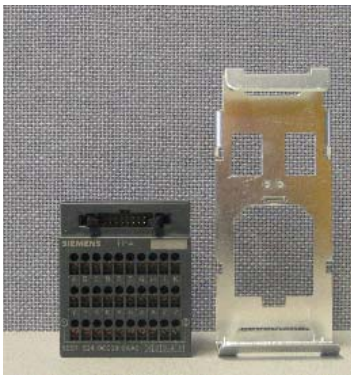
Step	Graphics	Description
1		Mounting of the SM331: • Insert the SM331 to the top part of the rail • Push it all the way left to the CPU • Push down • and tighten the screw at the bottom to the rail
2		Mounting of the front connector: • Press the upper release button of the front terminal block • Insert the front connector into the module until it snaps into the upper position
3		Mounting of the shroud Tighten the shroud on the lower side of the rail. Insert the two shielding terminals into the shroud.

.

4.2.6 Mounting of the TOP connect terminal blocks

The system wiring TOP connect requires a system specific terminal block

Table 4-6 Mounting of the TOP connect terminal block

Step	Graphics	Description
1		Insert the terminal block into the shielding support element
2		Snap the terminal block with the shielding support element onto a top hat rail
3		Mount the shielding terminals onto the shielding support element

Mechanically the sample station is now completely mounted.

5 Electrical connection

This chapter shows you how the various parts of the sample station are electrically wired from the power supply to the analog module.

Warning

You might get an electrical shock if the power supply PS307 is turned on or the power cord is connected to the line.

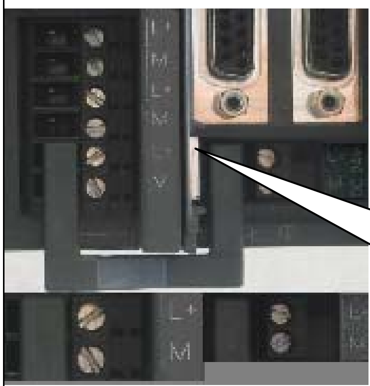
Wire the S7-300 only in power-off state.

5.1 Wiring of the power supply and the CPU

Figure 5-1 Wiring of the power supply and the CPU

The sample station requires a power supply. The wiring is done as follows:

Table 5-1 Wiring of the power supply and the CPU

Step	Graphics	Description
1		Open the front flaps of the power supply and the CPU
2		Unscrew the pull relief bracket of the power supply
3		Remove the insulation from the power cord, attach the cable end sleeves (for multi-wire cords) and connect it to the power supply
4		Tighten the pull relief bracket
5		Insert the power supply bridge between the power supply and the CPU and tighten the terminals. The grounding switch of the CPU does not have to be changed as the SM331 is already set up electrically isolated. Info on the grounding switch of the CPU: • Pressed: Electrically connected (factory default) • Pulled: Electrically isolated
6		Confirm that the voltage selector is set to your local power line voltage The power supply is set to a factory default voltage of AC 230V To modify the setting take the following measures: Remove the protective cover with a screw driver. Set the switch to your local power line voltage. Put the protective cover back.

5.2 Wiring of the analog module

The wiring of the analog module SM331 is dependant on the type of analog measurement transducer.

5.2.1 Shielded wiring for analog signals

For analog signals use shielded and twisted pair wires. That minimizes interference. The shielding of the analog wires should be grounded at both ends of the wire.

If there are any voltage differences between the ends of the wires, a compensation current might flow, which can interfere with the analog signals. In this case you should either ground the shield only at one end or install an appropriate compensation wire.

5.2.2 Voltage transducer wiring - principle

A voltage transducer should be wired as follows:

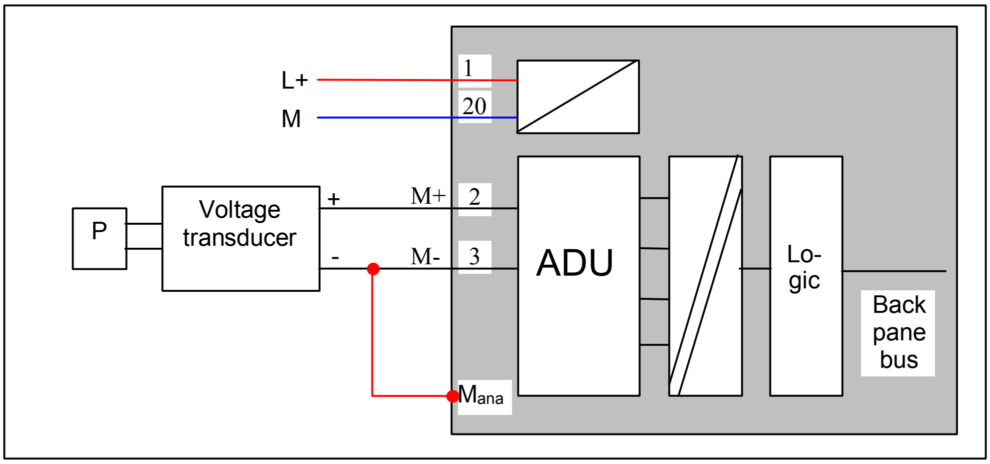


Figure 5-2 Wiring principle: Voltage transducer for electrically isolated SM331

If you use the SM331 module in a high electrically polluted environment (EMI) then you should connect M- to M_{ana}. Thus the potential difference between the inputs and the reference potential M_{ana} will not exceed the specified limit value.

5.2.3 Wiring principles of the resistance thermometer (PT100)

There are three possibilities to wire a resistance thermometer:

- 4-wire connection
- 3-wire connection
- 2-wire connection

For 4 wire and 3 wire connections the module delivers a constant current from its I_c+ and I_c- clamps which compensates the voltage drop of the measurement wires.

It is important that the connected wires with constant current are connected directly to the resistance thermometer.

Note

Because of the compensation, measurements with 4 or 3 wire connections deliver more accurate measurement results than a 2 wire connection.

---

4 wire connection of a resistance thermometer

The voltage at the resistance thermometer is measured at the M+ and M- terminals.

Make sure that the polarity of the connection I_c+ / M+ and I_c- / M- is correct and that the wires are directly connected to the resistance thermometer.

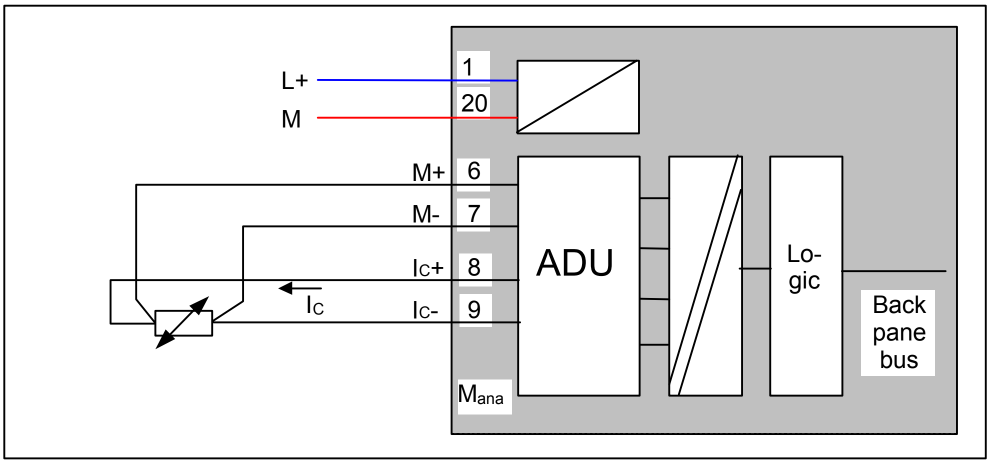


Figure 5-3 Wiring: 4 wire connection of a resistance thermometer

3 wire connection of a resistance thermometer

Normally on a 3 wire connector a bridge should be connected between M- und I_c- .

Make sure that the wires I_c+ and M+ are directly connected to the resistance thermometer.

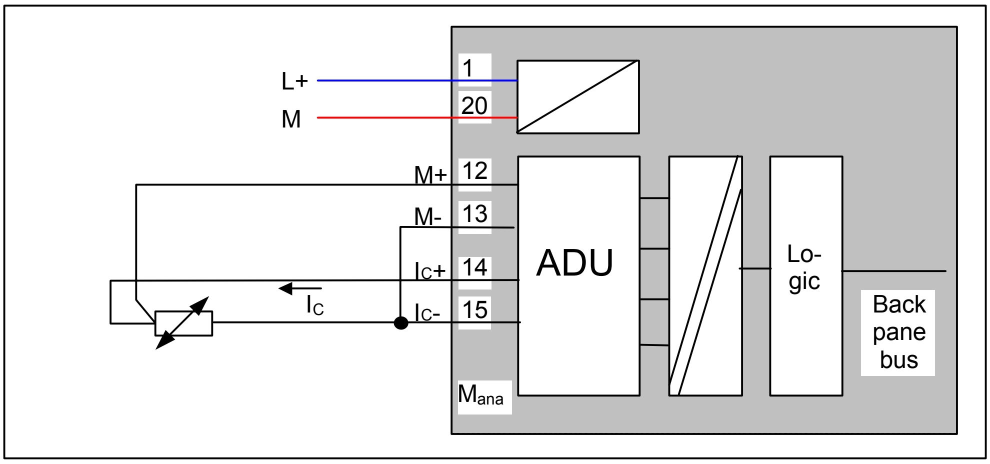


Figure 5-4 Wiring: 3 wire connection of a resistance thermometer

2 wire connection of a resistance thermometer

For a 2 wire connection, a bridge should be installed on the front connector of the module between clamps M+ and I_c+ and another bridge between clamps M- and I_c-.

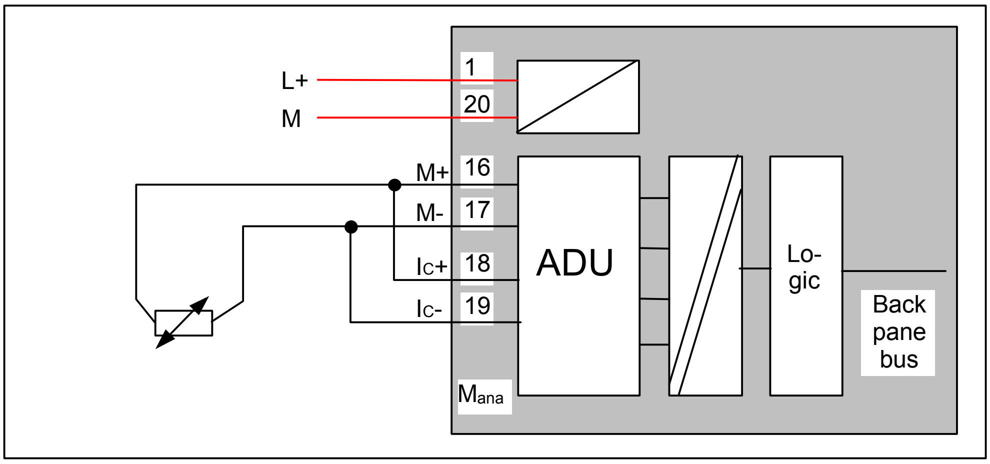


Figure 5-5 Wiring: 2 wire connection of a resistance thermometer

5.2.4 Conventional installation of an analog module

This chapter explains the conventional wiring of an analog module with single wires. The connection with system wiring TOP connect can be found in chapter 5.2.6.

The wiring of the analog module consists of the following tasks:

- Installation of the power supply (red cable)
- Installation of the voltage transducer (green cables)
- Connect unused channel of a channel group (see chapter 4.2.4) in parallel
- Wiring of the first PT100 with 4 wire connection (green cables)
- Wiring of the first PT100 with 3 wire connection (green cables)
- Wiring of the first PT100 with 3 wire connection (green cables)
- Wiring of the grounding (blue cables)

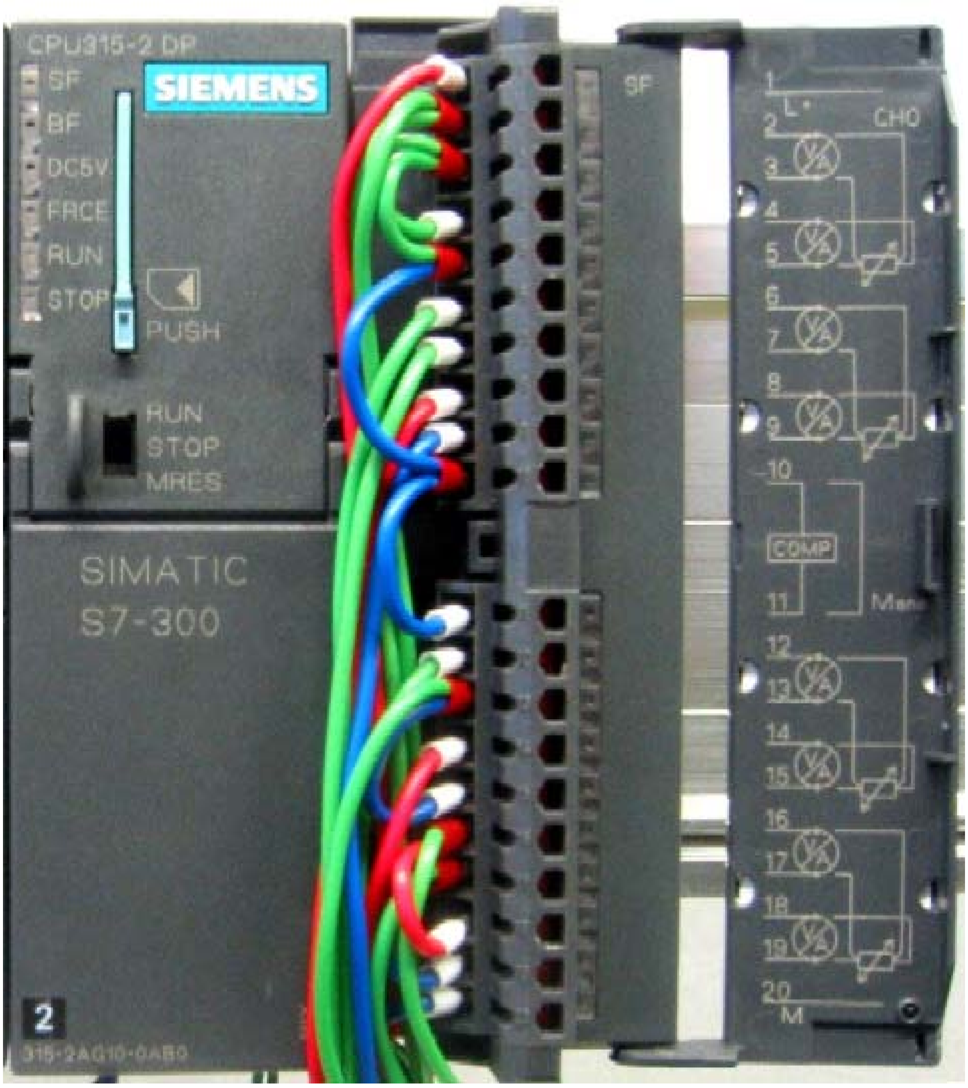
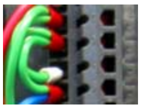
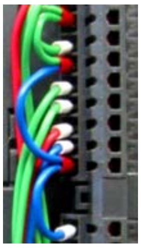


Figure 5-5 SM331 Front connector wiring

The tasks of the wiring are explained step by step below:

Table 5-2 SM331 Front connector wiring

Graphics	Wiring	Comments
	Open the front flap of the SM331	The terminals are pictured on the front flap
	Remove 6 mm of the insulation of the wire ends that you want to plug into the front connector and put the appropriate wire end sleeves.	
	Wire the front connector as follows: Terminal 1: L+	Power supply of the module
	Terminal 2: M+ Sensor 1 Terminal 3: M- Sensor 1 **Connect inputs in parallel:** Connect terminal 2 with 4 Connect terminal 3 with 5	Standard wiring for voltage transducer of electrically isolated modules In order to maintain the diagnostic functionalities of channel group 0, you should connect the second unused input in parallel to the first.
	Terminal 6: M+ PT100 (4 wire) Terminal 7: M- PT100 (4 wire) Terminal 8: Ic+ PT100 (4 wire) Terminal 9: Ic- PT100 (4 wire)	Standard wiring of a PT100 with 4 wire connection
	Connect terminal 10 (Comp) with M_{ana} Connect terminal 11 (M_{ana}) with terminal 3 and 5	Comp is not used for voltage measurement and PT100 Recommended for voltage trans-ducers
	Terminal 12: M+ PT100 (3 wire) Terminal 13: M- PT100 (3 wire) Terminal 14: PT100 (3 wire) Connect terminal 15 (Ic-) with 13 (M-)	Standard wiring of a PT100 with 3 wire connection
	Terminal 16: M+ PT100 (2 wire) Terminal 17: M-PT100 (2 wire) Connect terminal 18 (Ic+) with 16 M+ Connect terminal 19 (Ic-) with 17 (M-)	Standard wiring of a PT100 with 2 connection
	Terminal 20: M	Grounding

5.2.5 Wiring of the Connecting Terminals

In our example a terminal block replaces the connectors of the voltage transmitter or the resistance thermometer. The voltage is supplied with a calibrator and the resistance thermometer is simulated with a potentiometer.

Voltage measurement

In our example we simulate the voltage transducer through the following circuit:

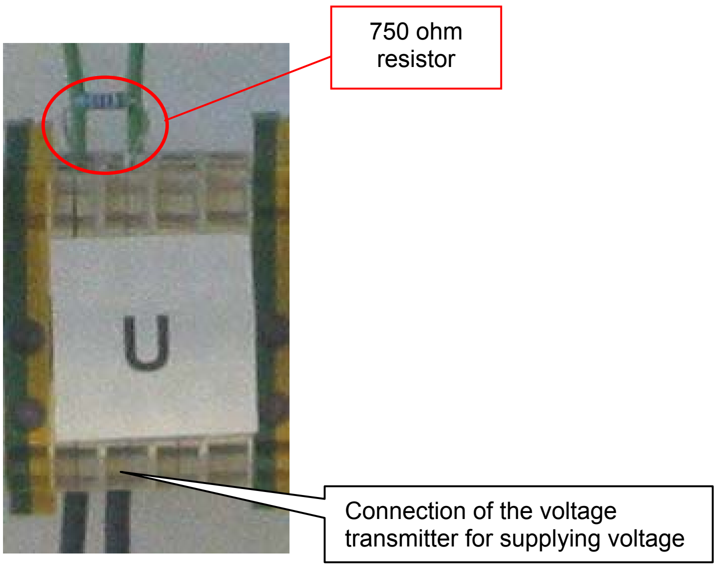


Figure 5-6 Terminal connection of the voltage transducer

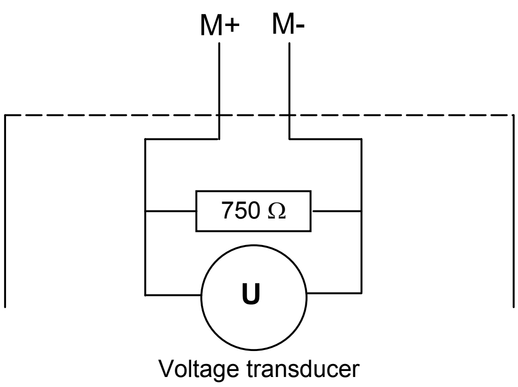


Figure 5-7 Basic wiring diagram of the voltage transducer

The circuit required for your voltage transducer can be found in your voltage transmitter manual.

Resistance Thermometer PT100

If you want to install a PT100, then wire the connecting terminal of the resistance thermometer as explained in chapter 5.2.3..

In our example a terminal block replaces the connecting terminal of the resistance thermometer. The desired resistance value is adjusted with a potentiometer.

To simulate the wires we use resistors. A 5 Ohm resistor simulates a copper connection with a cross sectional area of 0.6 mm^2 and a length of 171.4 m.

The length of the wiring is calculated based on the resistance using the following formula:

$$R = \frac{\rho * l}{q} \qquad l = \frac{R * q}{\rho}$$

R: Wire resistance

ρ: Specific resistance of the wire material (copper 0.0178Ω mm^2/m)

q: Cross sectional area of the wire

l: Length of the wire

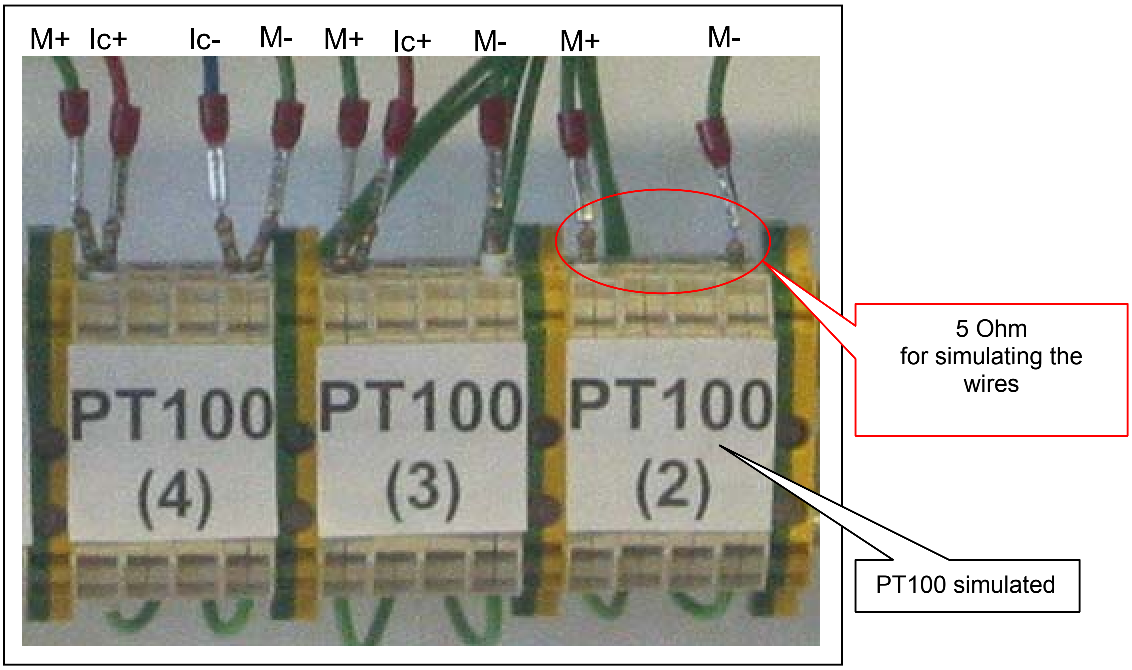


Figure 5-8 Terminal connection of PT100

5.2.6 Wiring of the analog module with system wiring TOP connect

With the system wiring SIMATIC TOP you connect the sensor specific wiring from the analog module to the TOP connect terminal block.

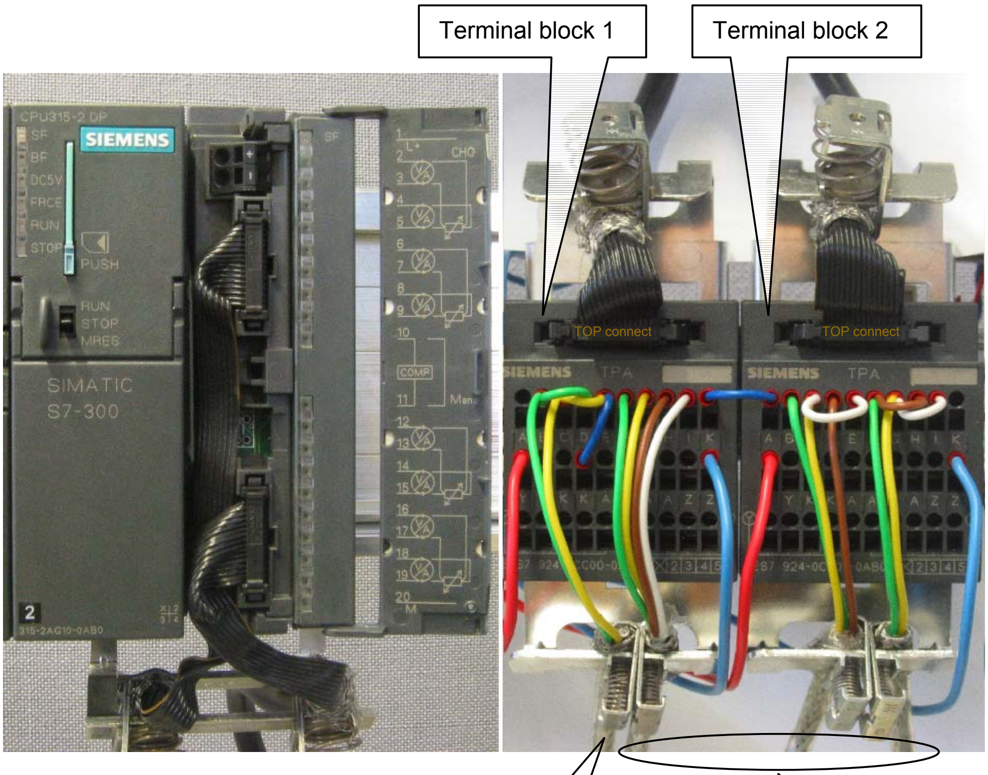


Figure 5-9 TOP connect wiring

The following table describes step by step the wiring tasks for the connection of terminal block 1. The connection of terminal block 2 follows the same procedures.

Table 5-3 SM331 Front connector wiring

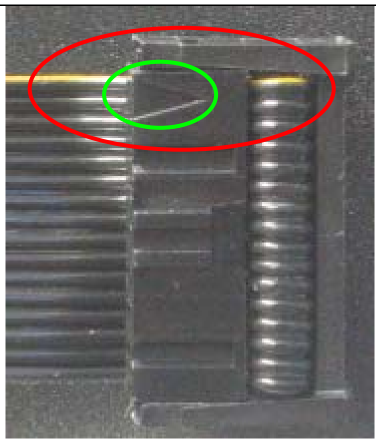
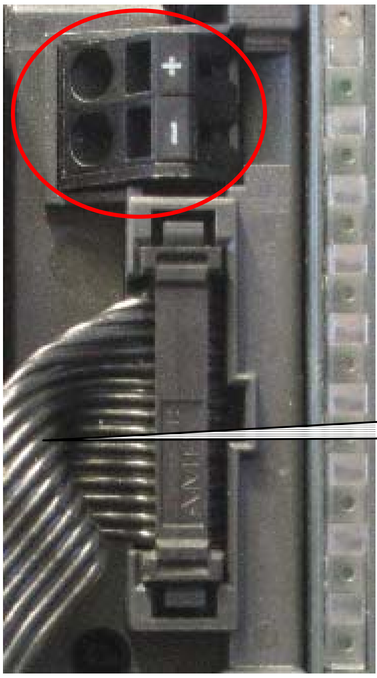
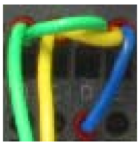
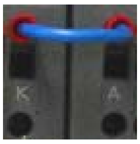
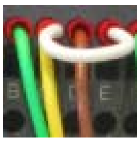
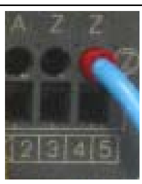
	Graphics	Wiring
1		Strip off the cable sheath from the top connect flat round cable at an appropriate length and lay the 16-wire flat cable open. Shorten the shielding to 15 mm and roll it back. Put the flat cable into the shielding terminal.
2		Put the flat cable into the connector and press it gently. Make sure that the triangular mark of the connector (green circle) and the yellow lead are on the same side.
3		Now connect the 16-pin flat connector to the front connector of the analog module If you need more than 4A current (in our example it is not needed), then you should connect a power supply to the front terminal of SM331 (see red circle). Wiring to terminal block 1
4		Put the other end of the flat round cable into the terminal block.

Table 5-4 SM331 Front connector wiring

	Graphics	Wiring	Comments
		**Terminal block 1 and 2:** Terminal Y: Power supply of the module	Up to a current consumption of 4A, the power supply can be connected through the terminal blocks. For higher currents, the power supply has to be connected directly to the front connector of the module.
4		**Terminal block 1:** Terminal B: M+ voltage transducer Terminal C: M- voltage transducer Connect terminal E and K <u>Connect inputs in parallel:</u> Connect terminal B with D Connect terminal C with E	Standard wiring for a voltage transducer at an electrically isolated module. To maintain the diagnostic functionalities of channel group 0, the second unused input has to be connected in parallel to the first.
5		**Terminal block 1:** Terminal F: M+ PT100 (4 wire) Terminal G: M- PT100 (4 wire) Terminal H: Ic+ PT100 (4 wire) Terminal I: Ic- PT100 (4 wire)	Standard wiring of a PT100 with 4 wire connection
6		**Terminal block 1:** Connect terminal K Comp with A M_{ana}	COMP is not used for voltage measuring and PT100. Recommended for voltage transducers
7		**Terminal block 2:** Terminal B: M+ PT100 (3 wire) Terminal C: M- PT100 (3 wire) Terminal D: Ic+ PT100 (3 wire) Connect terminal E: Ic- with C M-	Standard wiring of a PT100 with 3 wire connection
8		**Terminal block 2:** Terminal F: M+ PT100 (2 wire) Terminal G: M- PT100 (2 wire) Connect terminal H: Ic+ with F M+ Connect terminal I: Ic- with G M-	Standard wiring of a PT100 with 2 wire connection
9		**Terminal block 2:** Terminal Z: M	Grounding

Note

When an electrical isolation between the CPU and the analog module is required, then you should supply the analog module from a separate power supply.

5.2.7 Wiring of a PT100

The picture shows the connection of a PT100 with 4 wire connection. The connection of the wires takes place in PT100 by itself.

Figure 5-10 Wiring of PT100 with 4 wire connection

5.2.8 Switch on now

If you want to test the wiring, you may now switch the power supply on.

Do not forget to set the CPU to STOP mode (see the red circle).

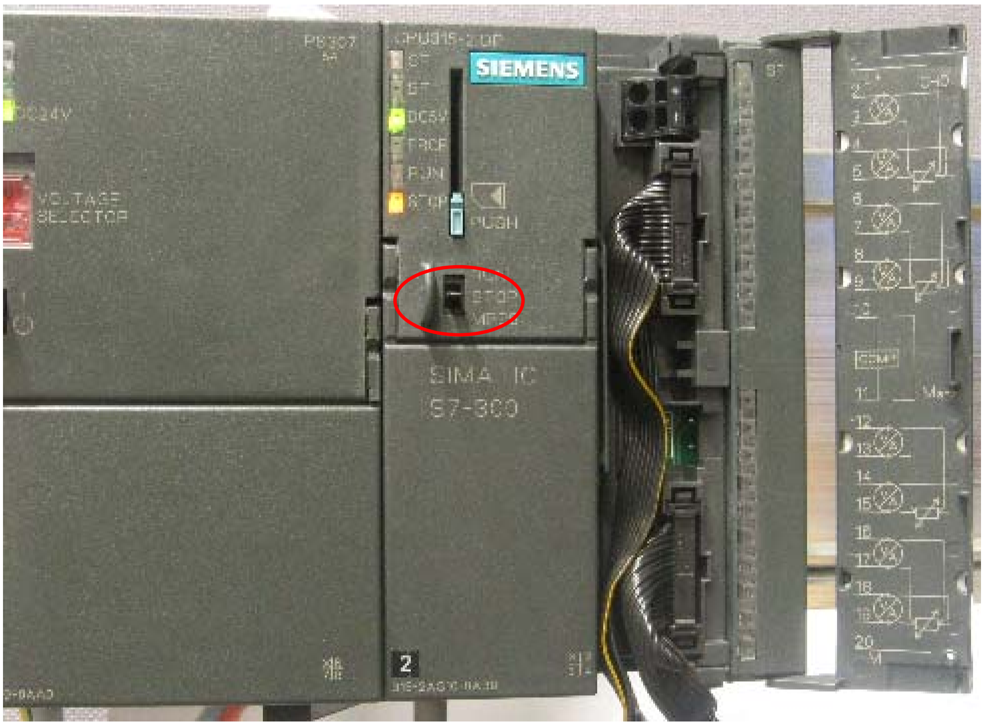


Figure 5-11 Successful wiring, CPU in position STOP

If a red LED is lit, then there is an error in the wiring. Verify your wiring.

6 Configuration with SIMATIC Manager

In this chapter you will be guided through the following steps:

- Creating a new STEP 7 project
- Parameterisation of the hardware configuration

6.1 Create a new STEP 7 Project

Use STEP7 V5.2 or later version for configuring the new CPU 315-2 DP.

Start SIMATIC Manager by clicking the symbol „SIMATIC Manager“ on your windows desktop and create a new project with the STEP 7 wizard „New Project“.

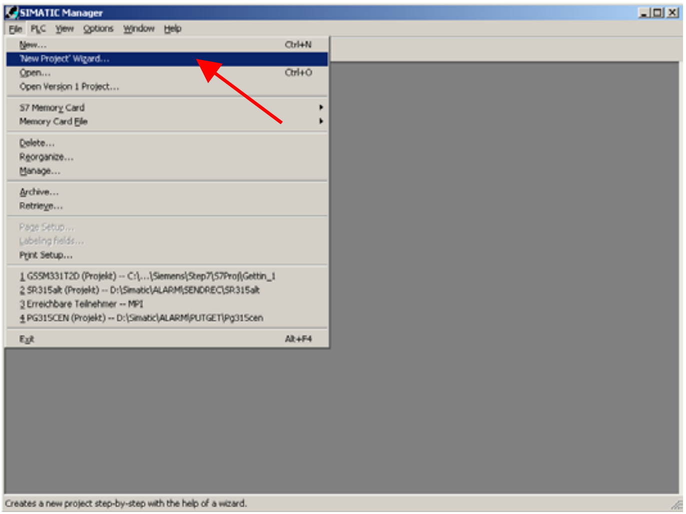


Figure 6-1 Execute STEP 7 wizard „New Project“

An introduction window pops up. The wizard will guide you through the creation of a new project.

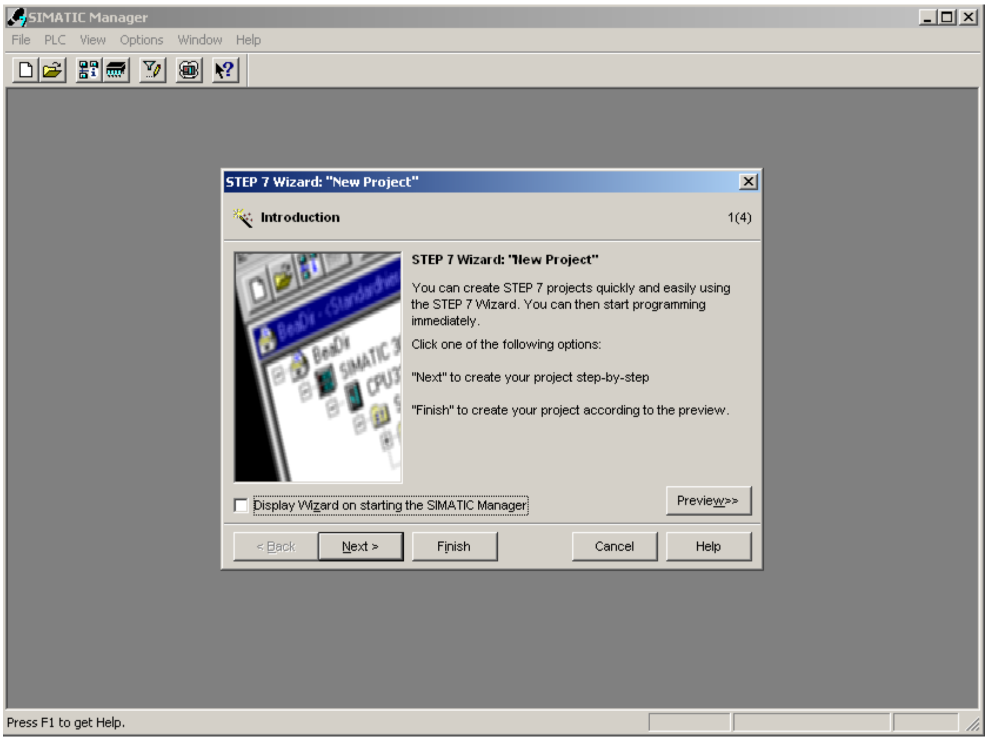


Figure 6-2 STEP 7 wizard „New Project“: Start

During the creation the following inputs are necessary:

- Selection of the CPU
- Define the basic user program
- Selection of organization blocks
- Project name

Click „Next“

6.1.1 CPU Selection

Choose the CPU 315-2DP for the sample project. (You can also use our example for a different CPU). Then choose your CPU.

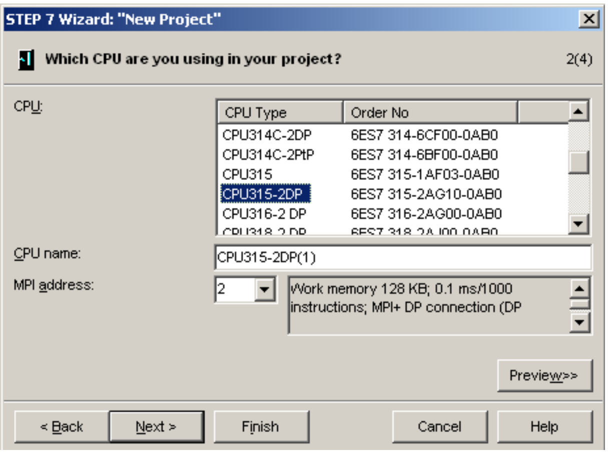


Figure 6-3 STEP 7 wizard „New Project“: CPU selection

Click „Next“

6.1.2 Define the basic user program

Choose the SIMATIC language STL and select the following organization blocks (OBs):

- OB1 Program Cycle Organization block
- OB40 Hardware interrupt
- OB82 Diagnostic interrupt

OB1 is required in every project and is called cyclically.
OB40 is called when a hardware interrupt occurs.
OB 82 is called when a diagnostic interrupt occurs.

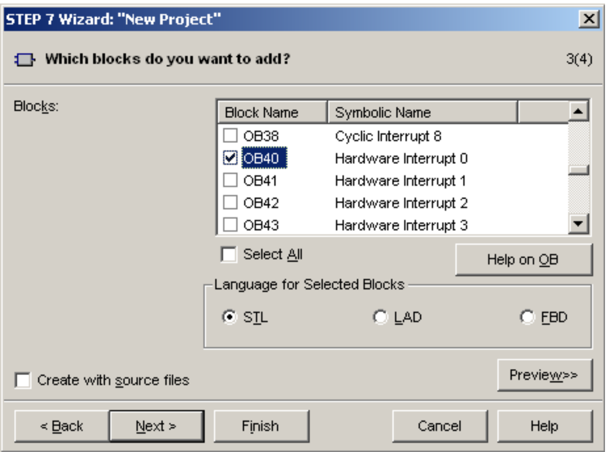


Figure 6-4 STEP 7 wizard „New Project“: Insert organization blocks

Click „Next“

6.1.3 Specify the project name

Select the edit field “Project name” and overwrite the name in it with “Getting Started S7 SM331”

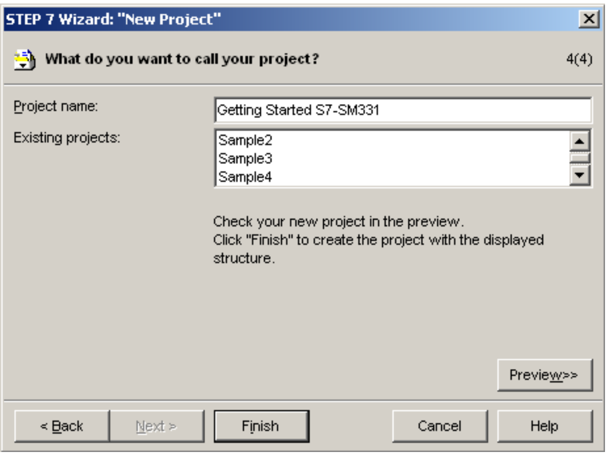


Figure 6-5 STEP 7 wizard „New Project“: Specify project name

Click „Finish“. The basic STEP 7 project is now created automatically.

6.1.4 Resulting S7 project is created

The wizard has created the project “Getting Started S7-SM331”. In the right pane you can find the inserted organization blocks.

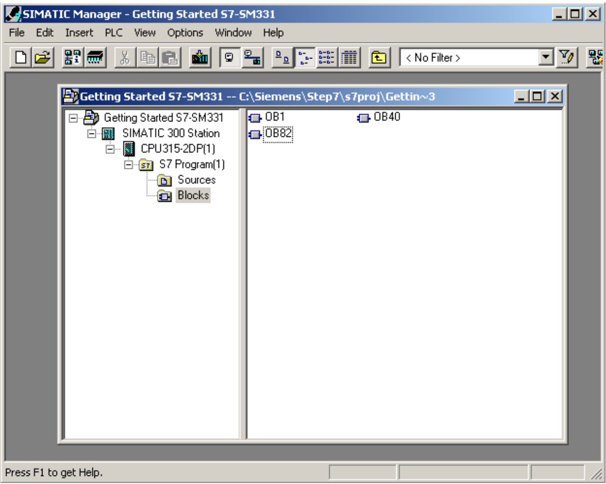


Figure 6-6 STEP 7 wizard „New Project“: Result

6.2 Configuration of the hardware configuration

The STEP 7 wizard has created a basic S7 project. You also need a complete hardware configuration in order to create the system data for the CPU.

6.2.1 Create the hardware configuration

You create the hardware configuration of the sample station with SIMATIC Manager.

In order to do this, select the folder „SIMATIC 300 Station“ on the left hand pane. Start the hardware configuration by double clicking the icon “Hardware” on the right hand pane.

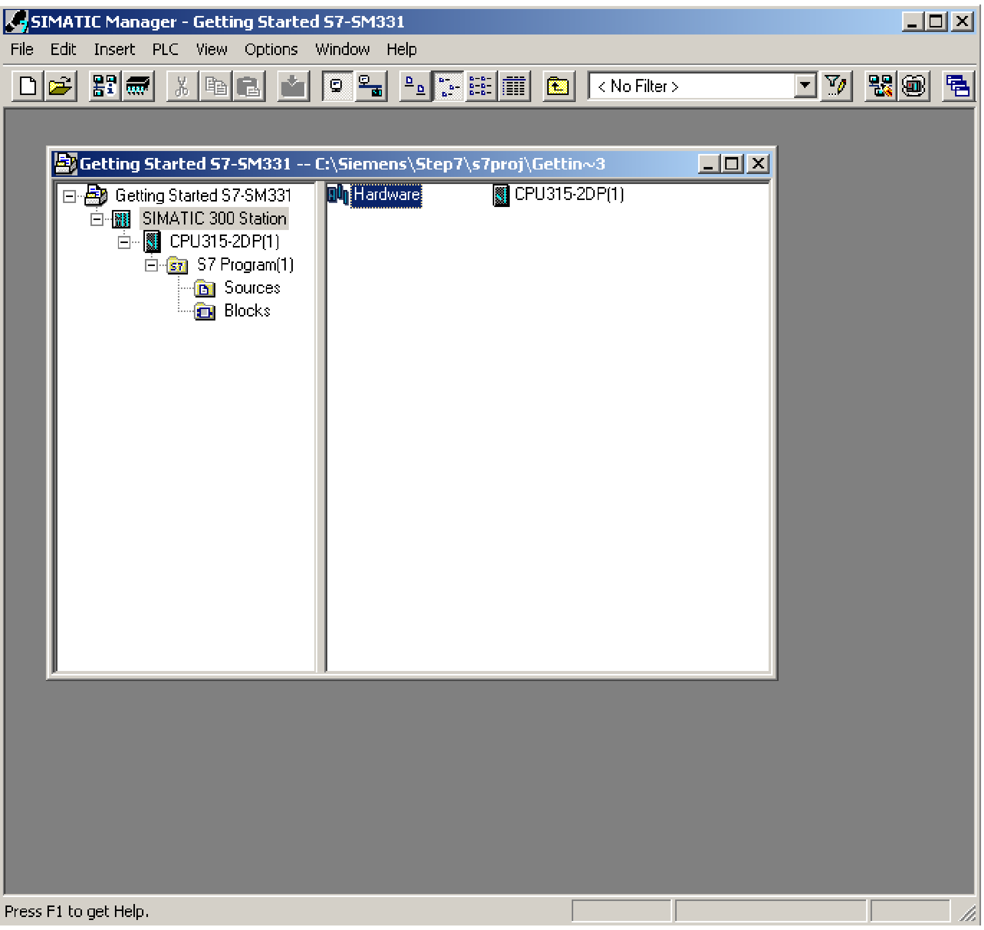


Figure 6-7 Starting the hardware configuration

6.2.2 Insert SIMATIC components

First select a power supply module from the hardware catalogue.

If the hardware catalogue is not visible, open it with the shortcut key Ctrl+K or by clicking the catalogue symbol (blue arrow).

In the hardware catalogue you can browse through the folder SIMATIC 300 to the folder PS-300.

From the right window pane, select the PS307 5A and drag it into slot 1 (see red arrow) of your rack.

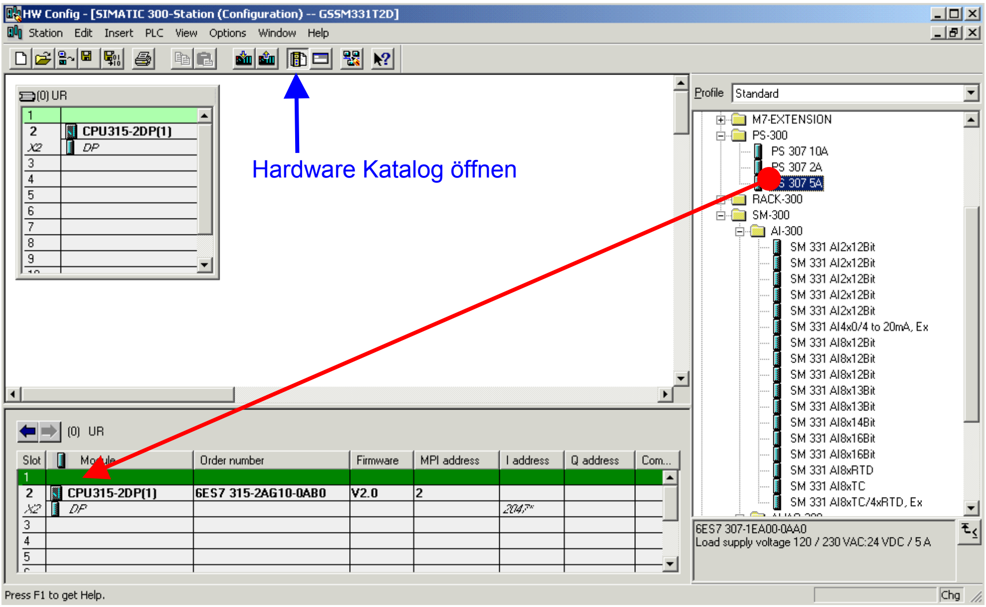


Figure 6-8 Hardware configuration: Basic configuration

Result: PS 307 5A appears in the configuration of your rack.

Insert analog module

There are many SM331 analog modules. For this project we use an SM331, AI8x12 Bit with the order number 6ES7 331-7KF02-0AB0.

The order number is displayed at the bottom of the hardware catalogue (see blue arrow).

Click on SM331 AI8x12Bit on the right window pane and drag it into the first available field on slot 4 of your configuration table (see red arrow).

You have inserted all the modules into the hardware configuration. In the next step you parameterise the modules.

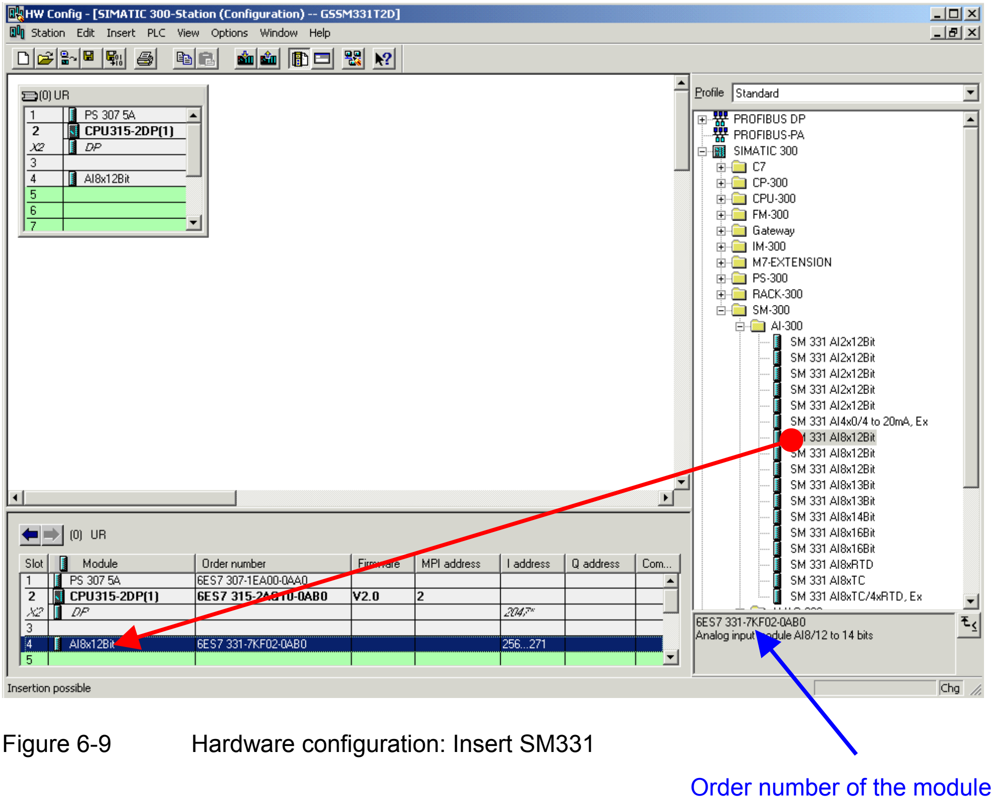


Figure 6-9 Hardware configuration: Insert SM331

Result: You can now parameterise SM331.

6.2.3 Parameterisation of the analog module

SIMATIC Manager inserts the analog module with its standard settings. You can modify the parameters to change the sensor types, diagnostics and interrupt capabilities.

Functionality of the sample station

The table below shows the parameters to be set for our sample station

Table 6-1 SM331 Functionality of the sample station

Function	Description
Process reactions	• Diagnosis - active • Hardware interrupt when limit value exceed - active
Sensor 1	• Voltage transducer • Group diagnostics • Measuring range ±5V • Limit value -3 Volt and +3 Volt
Sensor 2	• Resistance thermometer PT100 • Group diagnostics • Wire break detection • Limit value -20 °C and +50 °C
Sensor 3	• Resistance thermometer PT100 • Group diagnostics • Wire break detection
Sensor 4	• Resistance thermometer PT100 • Group diagnostics • Wire break detection

Open the parameterisation

Double click slot 4 that has the SM331 in it.

Select the tab inputs.

Parameterise as follows:

- Diagnostic interrupt - checked
- Hardware interrupt - checked
- Input 0-1:
    - Measuring type: U
    - Measurement range ±5
    - Group diagnostics - checked
- Input 2-3, 4-5 and 6-7
    - Measuring type: RT
    - Measuring range PT100 standard
    - Group diagnostics - checked
    - Wire break - checked
- Interference frequency
    - Select your local power line frequency (50 Hz or 60 Hz)
- Trigger for hardware interrupt channel 0
    - High limit +3 V
    - Low limit -3V
- Trigger for hardware interrupt channel 2
    - High limit +50 °C
    - Low limit -20 °C

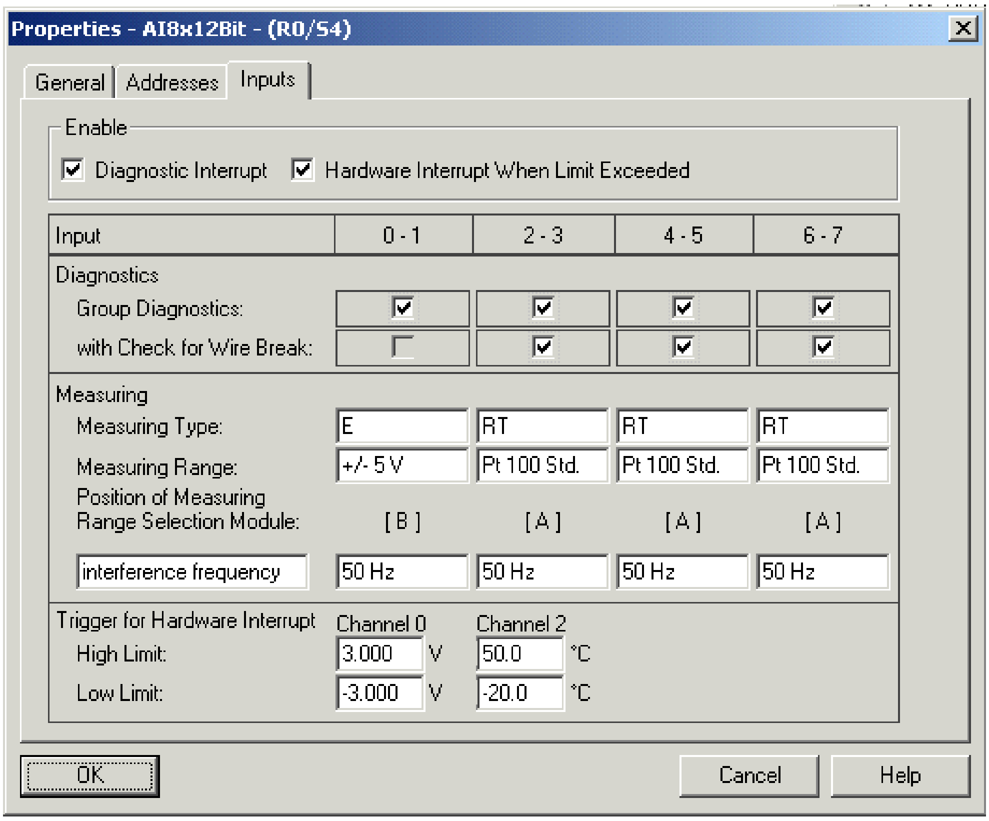


Figure 6-10 SM331: Parameterisation

Explanation of the individual settings for SM331

Diagnostic Interrupt

Activating the ”Diagnostic Interrupt” configures OB86 to be called when the grounding or the supply voltage is missing..

Hardware interrupt

Activating the parameter „Hardware Interrupt When Limit Exceeded “ configures the hardware interrupt OB40 to be called when the set limit values exceed or fall below.
The limit values can be parameterised in the same window under „Trigger for hardware interrupt“.

Group diagnostics

By selecting Group Diagnostics, channel bound diagnostic interrupts are activated (see chapter 8.3). When a diagnostic event occurs, OB86 is called.

Wire break detection

By activating the Check for Wire Break, wire breaks are diagnosed. The diagnostic OB86 is called.

Measuring type

U stands for voltage

RT stands for resistance (Resistance, Temperature).

Measuring range

Specification of the measuring range of the voltage transducer and the type of PT100.

Position of the measuring range module

The position of the measuring range modules is shown (chapter 4.2.4).

Interference frequency (interface frequency suppression)

Set the interference frequency to your local power line frequency.

Trigger for hardware interrupt

If the parameter „Hardware Interrupt When Limit Exceeded " is selected, you can specify the limit values. If the specified value exceed or fall below, the hardware interrupt OB40 is called.

Only channels (inputs) 0 and 2 are hardware interrupt capable.

Completion of the hardware configuration

Close the parameter window.

Compile and save the project via Station -> Save and Compile (Ctrl+S)

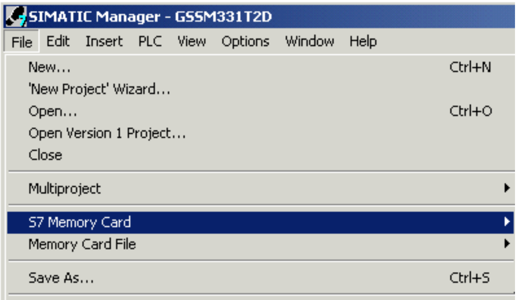
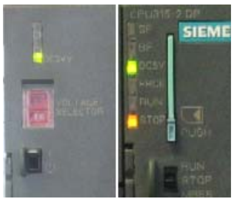
With this the hardware configuration of the project is completed.

6.2.4 Power up test

For testing, do a power up test and download the system data.

Power up

Table 6-2 Power up

#	Graphics	Description
1		Erase your Micro Memory Card with a Power PG or a PC with external programming device: In SIMATIC Manager click “File → S7 Memory Card → Delete …” The MMC is deleted.
2		Turn off the CPU’s power supply. Insert the MMC into the CPU. Turn on the power supply.
3		If the CPU is in RUN mode, set it to STOP mode.
4		Turn on the power supply again. If the STOP LED blinks, the CPU requests for a reset. Acknowledge this by turning the mode switch to MRES for a quick second.
5		Connect the CPU to the PG with an MPI cable. To do this, connect the MPI cable with the CPU’s MPI port. Connect the other end to the PG interface of your programming device.

Download hardware configuration

Download the hardware configuration into the CPU with HW Config.

Click the symbol „Load to module“ (shown in the red circle).

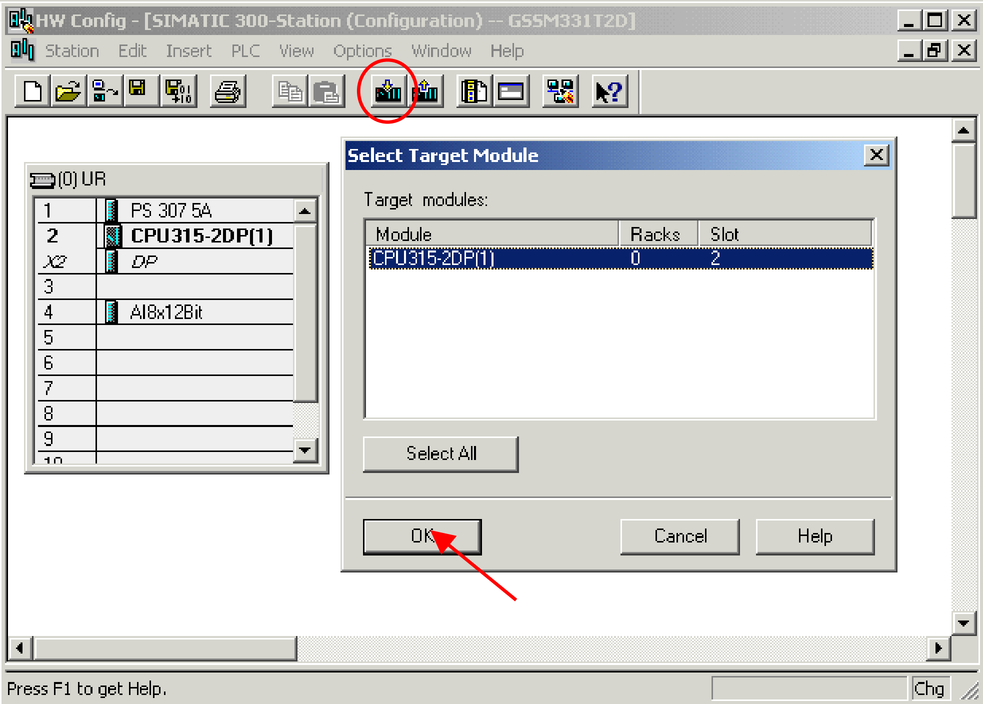


Figure 6-11 Download the CPU’s hardware configuration (1)

When the dialog window „Select Target Module“ pops up, click “OK” (see red arrow).

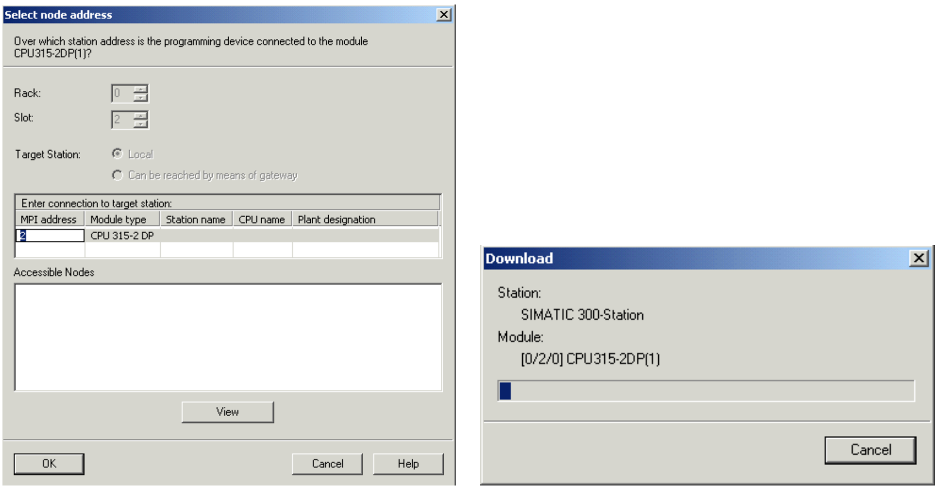


Figure 6-12 Download the CPU’s hardware configuration (2)

The dialog window „Select node address” is shown. The system data are now transferred into the CPU.

Start CPU

Set the CPU to RUN mode

If the hardware configuration is done right, two green LEDs (RUN and DC5V) should be lit on the CPU

Figure 6-13 CPU in error free state

If the RUN LED light is not lit, then there is an error.

Read the diagnostic buffer with a PG in order to localize the error. Possible causes are:

- The wiring is connected wrong
- The coding device is inserted wrong
- You have entered the wrong parameters for SM331

6.3 STEP 7 User Program

6.3.1 Function of the user program

The sample user program:

- Stores the sensor values in a data block
- saves the status information of the hardware interrupts in a memory word.

The status information is acknowledged by means of a bit. Furthermore the channel values (values of the input words) are stored in another data block.

In the user program the following tasks have to be performed:

1. Cyclical storage of the analog input values in a data block (DB1)
2. Cyclic conversion of the analog input values to floating point values (FC1) and storage in a data block (DB2)
3. Acknowledgement of the hardware interrupt status when the acknowledge marker (M200.0) is TRUE.
4. Store the status in a marker word (MW100) when a hardware interrupt occurs.

Table 6-3 Structure of the user program

Execution mode	Responsible organization block	Programming task	Used block or marker
Cyclic execution	OB1	Store analog input values	DB1
		Convert and store the sensor signals	FC1, DB2
		Acknowledge hardware interrupt	M200.0
Execution triggered by hardware interrupt	OB40	Store status	MW100
Execution triggered by diagnostic interrupt	OB82	Has to be implemented because a module with diagnostic capabilities is used	---

Diagnostic interrupt OB82

OB82 is used for modules with diagnostic capabilities in the STEP 7 program.

If the module detects a failure (coming and going events), the module requests the CPU for diagnostics. As a reaction to this the operating system calls OB82.

In our example we only use OB82 in order to prevent the CPU from changing to STOP mode. You can program OB82 to react on hardware interrupts.

6.3.2 Create User Program

There are two ways to create a user program:

- o If you know how to program STEP 7 STL, then you can create and program the necessary blocks and function blocks in the Blocks folder of STEP 7.
- o You can insert the user program from an STL source into the project. In this “Getting started” we describe this way.

Creating a user program in STEP 7 requires three steps:

1. Download of the source file directly from the web page
2. Import source file
3. Compile source file

1. Downloading the source file

You can download the source file directly from the web page from which you loaded this “Getting Started”. Click on “Info”. The download window will open.

- Note the name of the source file
- Save the source file to your hard drive.

2. Import source file

You can import the source file into SIMATIC Manager as follows:

- Right click the folder „Sources“
- Select „Insert new Object“ → External Source...

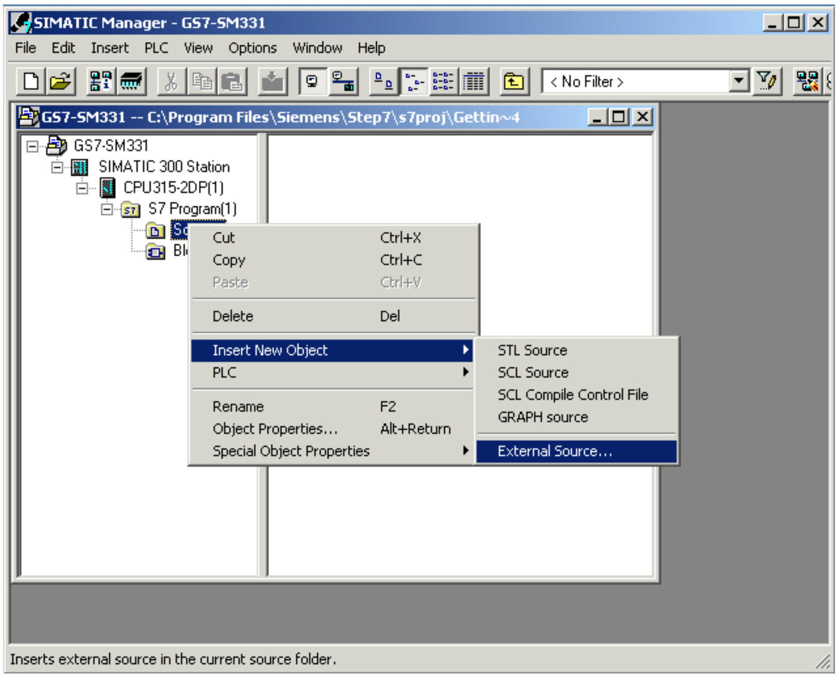


Figure 6-14 Import external source

In the dialog window „Insert external source“ browse for the source file that you have already downloaded and saved on your hard drive.

Select the source file GSSM331T2DE.AWL (see red arrow).

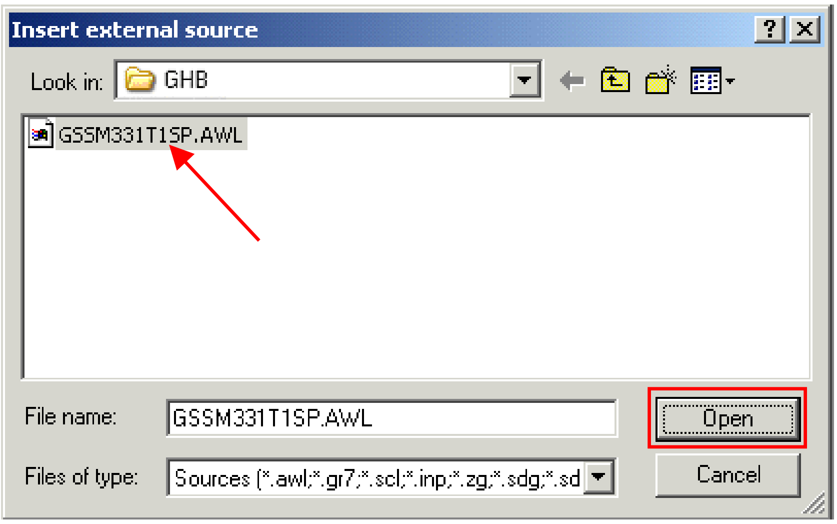


Figure 6-15 Import external source

Click „Open“

SIMATIC Manager has opened the source file. On the right pane you can see the source file.

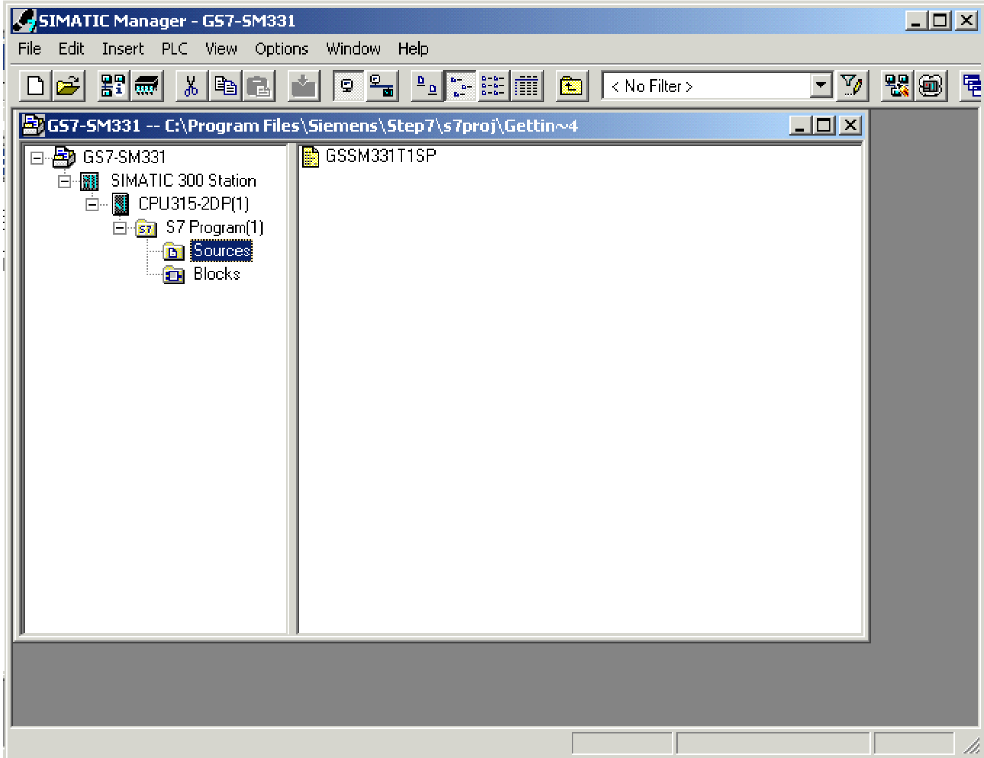


Figure 6-16 Storing the source file

3. Compile Source Code

In order to create an executable STEP 7 program, the STL source has to be compiled.

Double click the source file in the Sources folder. The source code editor is called.

In the window of the source code editor you can view the source code (Code from chapter 10).

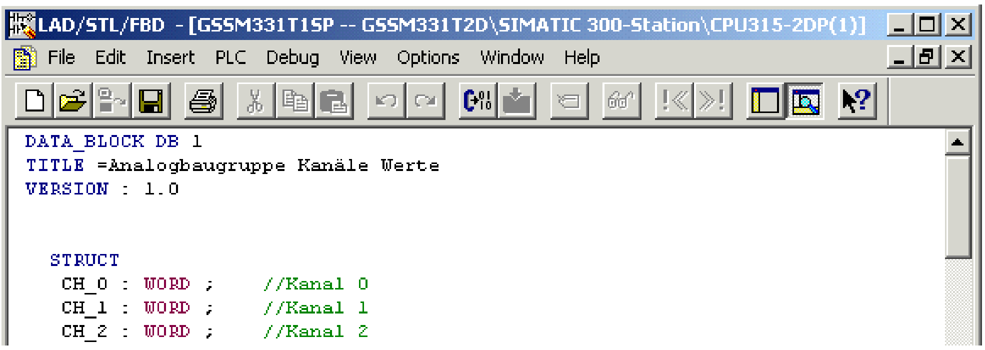


Figure 6-17 Source code editor

After the source code is loaded, start the compilation.

Press the shortcut key Ctrl+B or select File → Compile. The compilation starts immediately.

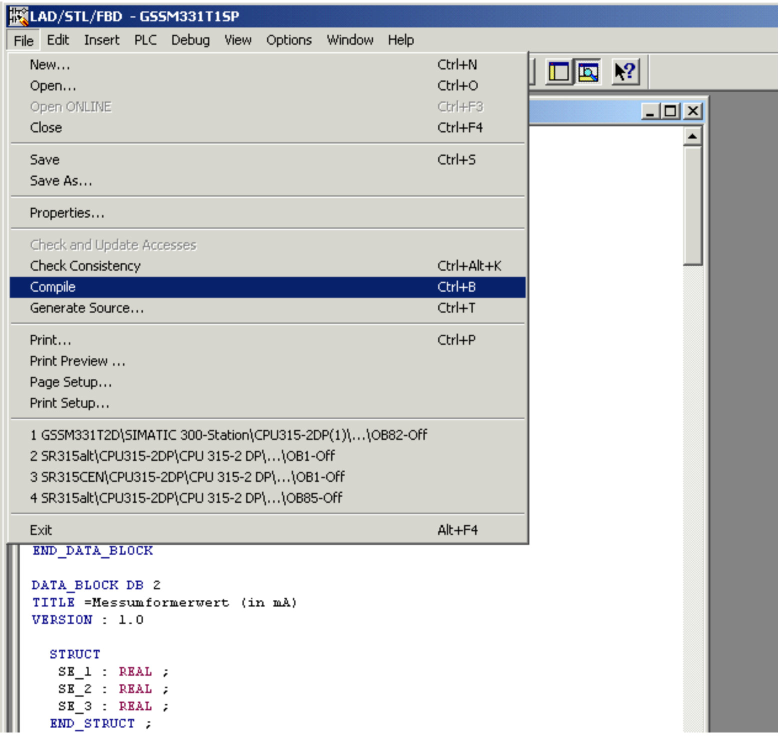


Figure 6-18 Translation of the STL source

In case of warning or error messages, check the source code.

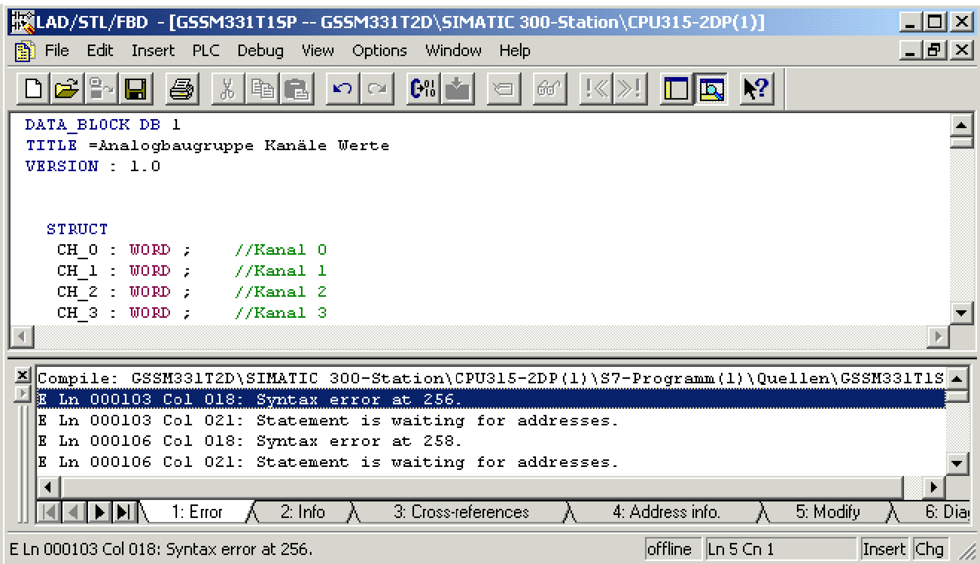


Figure 6-19 Source code editor, messages after compilation

Close the source code editor.

After compiling the STL source without errors the following blocks should appear in the Blocks folder:

OB1, OB40, OB82, FC1, DB1 and DB2

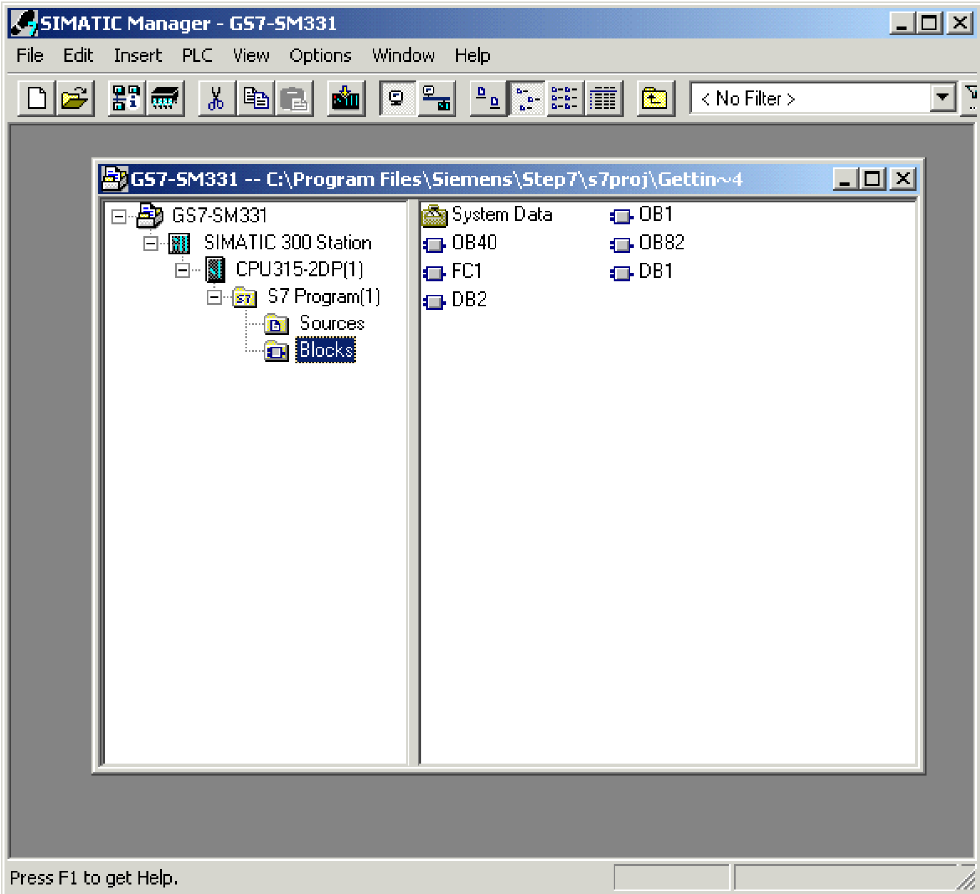


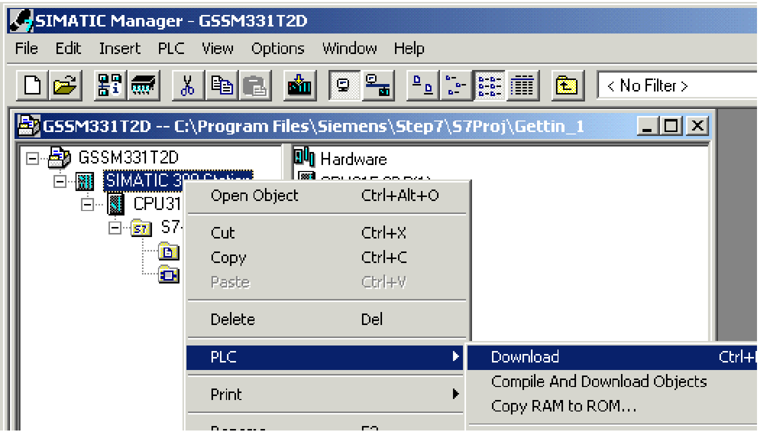
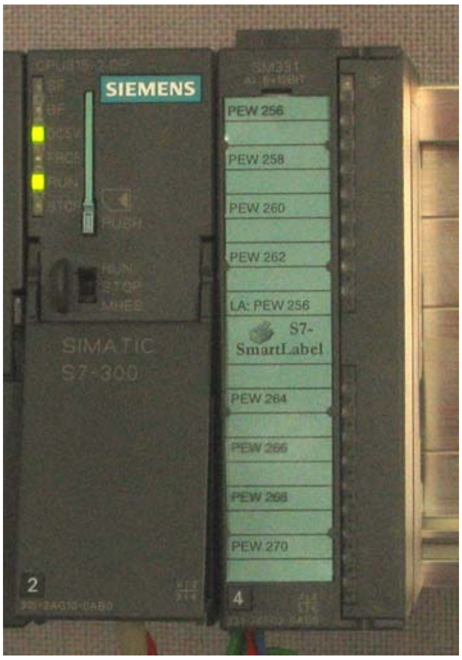
Figure 6-20 Generated blocks

7 Testing the User Program

7.1 Download system data and user program

Hardware and software are ready now. The next step is to download the system data and the user program into the automation system. To do this, execute the following steps:

Table 7-1 Download user program and system data

Step	Description
1	Download the system data (containing hardware configuration and the user program) into the CPU using SIMATIC Manager. 
2	 Follow the instructions on the screen. If all sensors are properly con-nected, the CPU and the SM331 do not show any red error lights. The error free state of the CPU is displayed by the green „RUN“ light.

Smart Label

The labelling strips for the modules were created with Siemens S7-SmartLabel (Order number.: 2XV9 450-1SL01-0YX0)

The original size of the labelling strips is displayed in Figure 7-1

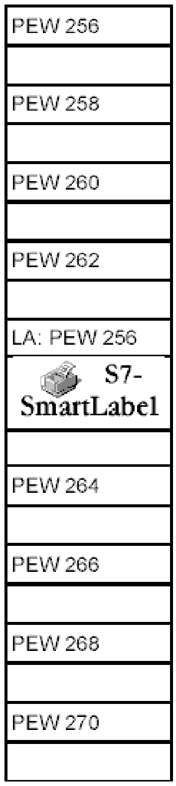


Figure 7-1 S7-SmartLabel labelling strip for the example

7.2 Visualisation of the sensor signals

In order to visualise the sensor signals, insert a variable table as follows into the project. To do this, select from the context menu of the Blocks folder:

Insert new object -> Variable table

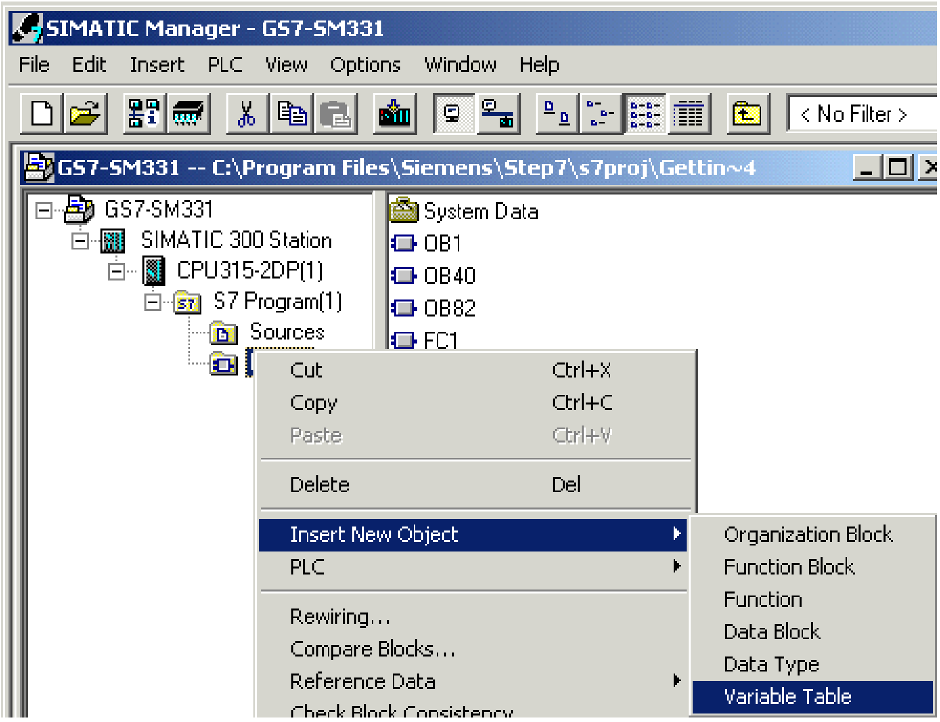
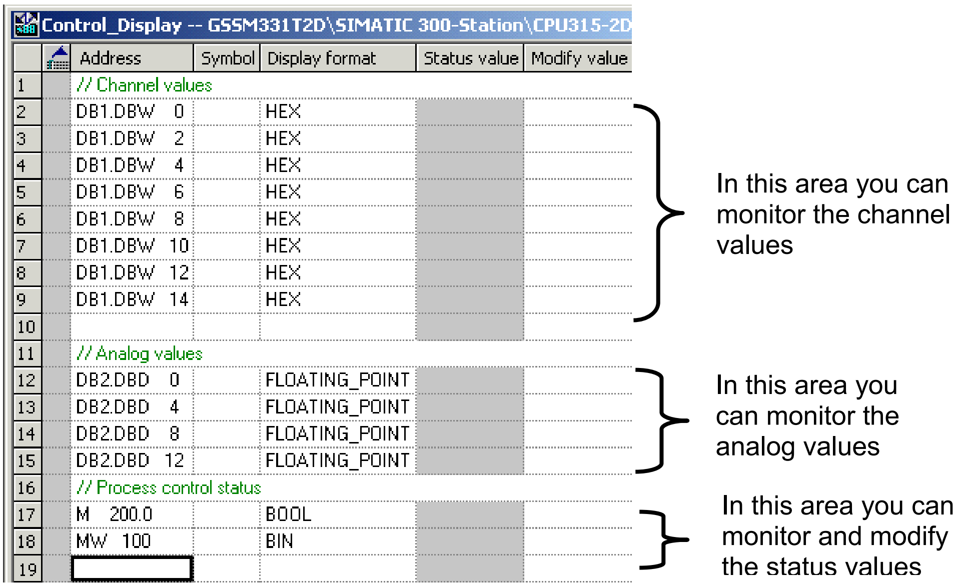


Figure 7-2 Insert Variable Table

Fill in the new variable table as follows:


Control_Display -- GSSM331T2D\SIMATIC 300-Station\CPU315-2D

	Address	Symbol	Display format	Status value	Modify value
1	// Channel values				
2	DB1.DBW 0		HEX		
3	DB1.DBW 2		HEX		
4	DB1.DBW 4		HEX		
5	DB1.DBW 6		HEX		
6	DB1.DBW 8		HEX		
7	DB1.DBW 10		HEX		
8	DB1.DBW 12		HEX		
9	DB1.DBW 14		HEX		
10					
11	// Analog values				
12	DB2.DBD 0		FLOATING_POINT		
13	DB2.DBD 4		FLOATING_POINT		
14	DB2.DBD 8		FLOATING_POINT		
15	DB2.DBD 12		FLOATING_POINT		
16	// Process control status				
17	M 200.0		BOOL		
18	MW 100		BIN		
19					

Figure 7-3 Variable table Control_Display

Monitoring of variables

In order to monitor variables, open the online view of the controller by clicking the Eye Glasses symbol. Now you can monitor the values in the data blocks and markers.

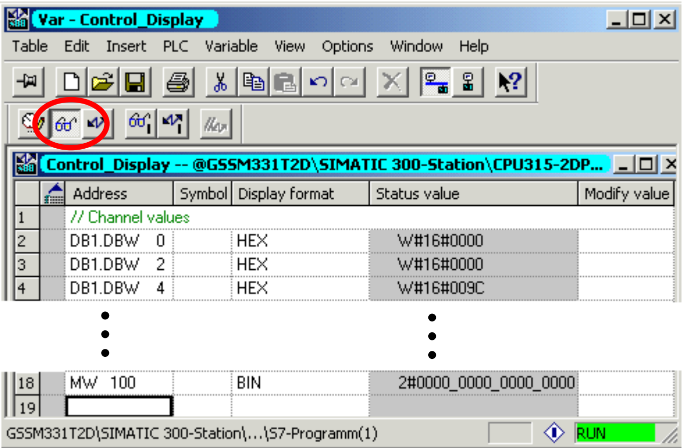


Figure 7-4 Online view of the variable table

Specifics for monitoring the variables

While monitoring the values you will notice that the channel values are different from the analog values. The reason for this is that the analog module only supports the binary format "Word" (16 bits). Therefore the values of the analog module have to be converted.

Modifying variables

For controlling the Process Control Acknowledgement enter the desired value (TRUE or FALSE) into the column „Modify Value". The value depends on whether you want to activate or deactivate the acknowledgement. Click the symbol with the two arrows.

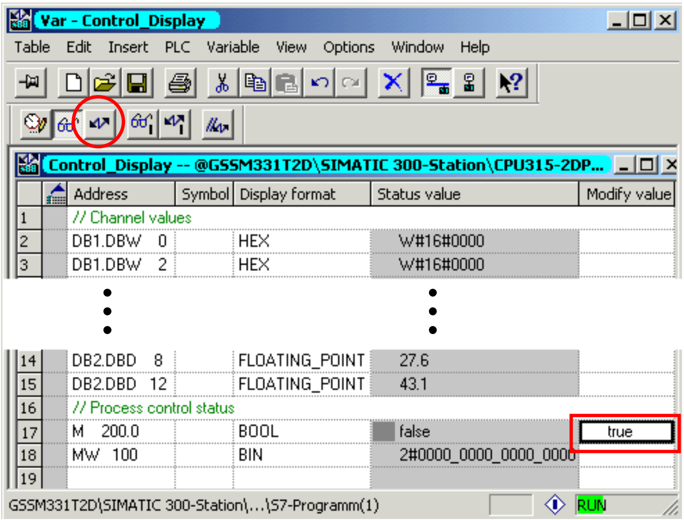


Figure 7-5 Modifying variables

7.3 Analog value display

Analog input modules convert the analog process signal into a digital format (16 bit word).

If you want to display the analog process variables, you should convert the digital values of the module to decimal values. In our example program, the process variable is displayed in a readable format in Volt (V) or for PT100 in °C. The conversion of the digital values to decimal values is implemented in the programmed function FC1.

Five ranges have to be taken into account when converting from digital to analog values. The following table describes these ranges.

7.3.1 Analog value display of a ± 5V voltage transducer

Table 7-2 Analog value display for a ± 5V voltage range

Analog value display		Voltage range	Scope	Comment
Decimal	Hex decimal			
32767	7FFF	5,926V	Overflow	From hex value 16#7F00 on, the sensor value is above the over steering range and is no more valid.
32512	7F00			
32511	7EFF	5,879V	Over steering range	This range corresponds to a tolerance band before the overflow range is reached. Within this range the resolution is not optimal.
27644	6C01			
27648	6C00	5 V	Nominal range	The nominal range is the normal range for acquiring measurement values. This range guarantees optimal resolution.
20736	5100	3,75V		
1	1	180,8µV		
0	0	0V		
-1	FFFF			
-20736	AF00	-3,75V		
-27648	9400	-5V		
-27649	93FF		Under steering range	Range corresponding to the over steering range but for low values.
-32512	8100	-5,879V		
-32513	80FF		Underflow	From hex value 16#80FF on, the sensor signal is below the configured measurement value range and is no more valid.
-32768	8000	-5,926V		

With the aid of a voltage source (calibrator) you can now compare the given values with the values of the analog value display in the table. The values will be identical.

7.3.2 Analog value display of a ± 10V voltage transducer

Table 7-3 Analog value display for a ±10V voltage range

Analog value display		Voltage range	Scope	Comment
Decimal	Hex decimal			
32767	7FFF	11,851V	Overflow	From hex value 16#7F00 on, the sensor value is above the over steering range and is no more valid.
32512	7F00			
32511	7EFF	11,759V	Over steering range	This range corresponds to a tolerance band before the overflow range is reached. Within this range the resolution is not optimal.
27644	6C01			
27648	6C00	10V	Nominal range	The nominal range is the normal range for recording measurement values. This range guarantees optimal resolution.
20736	5100	7,5V		
1	1	361,7µV		
0	0	0V		
-1	FFFF			
-20736	AF00	-7,5V		
-27648	9400	-10 V		
-27649	93FF		Under steering range	Range corresponding to the over steering range but for low values
-32512	8100	-11,759 V		
-32513	80FF		Underflow	From hex value 16#80FF on, the sensor signal is below the configured measurement value range and is no more valid.
-32768	8000	-11,851V		

7.3.3 Analog value display of a 0 - 10V voltage transducer

Table 7-4 Analog value display for a 0-10V voltage range

Analog value display		Voltage range	Scope	Comment
Decimal	Hex decimal			
32767	7FFF	11,851V	Overflow	From hex value 16#7F00 on, the sensor value is above the over steering range and is no more valid.
32512	7F00			
32511	7EFF	11,759V	Over steering range	This range corresponds to a tolerance band before the overflow range is reached. Within this range the resolution is not optimal.
27644	6C01			
27648	6C00	10V	Nominal range	The nominal range is the normal range for recording measurement values. This range guarantees optimal resolution.
20736	5100	7,5V		
1	1	361,7µV		
0	0	0V		
			Under steering range	A negative value is not possible

7.3.4 Analog value display of PT100 Standard

Table 7-5 Analog value display for resistance thermometer PT100 standard

Analog value display		Voltage range	Scope	Comment
Decimal	**Hex decimal**			
32.767	7FFF	> 1000°C	Overflow	From hex value 16#2711 on, the sensor value is above the over steering range and is no more valid.
10.000	2710	1000 °C	Over steering range	This range corresponds to a tolerance band before the overflow range is reached. Within this range the resolution is not optimal.
	...	...		
8.501	2135	850,1 °C		
8.500	2134	850 °C	Nominal range	The nominal range is the normal range for recording measurement values. This range guarantees optimal resolution.
	…	…		
	…	…		
-2.000	F830	-200 °C		
-2.001	F82F	-200,1 °C	Under steering range	Range corresponding to the over steering range but for low values.
-2.430	F682	-243 °C		
-2431	F681		Underflow	From hex value 16#F681 on, the sensor signal is below the configured measurement value range and is no more valid.
-32.768	8000	< -243 °C		

7.4 Influence of PT100 wiring on analog value display

The wiring of the PT100 has a great influence on measurement value acquisition.

The connecting wires from the SM331 module to the PT100 thermometer have a resistance that is dependant on the wire's material, the length and the cross section of the wires.

If you want to compensate the wiring resistance, then you should choose 3 or 4 wire connection.

Measure the environmental temperature with the three PT100 with a wiring resistance of 5 Ohm (170m copper wire 0,6mm^2). You get the following measurement values:

Table 7-6 Influence of the wiring on temperature measurement

PT100 Connection type	Environmental temperature	Analog value display	Measured temperature	Absolute error
4 wire	17,0 °C	00AA Hex.	17,0 °C	0 °C
3 wire	17,0 °C	013C Hex.	31,6 °C	14,6 °C
2 wire	17,0 °C	01BD Hex.	44,5 °C	27,5 °C

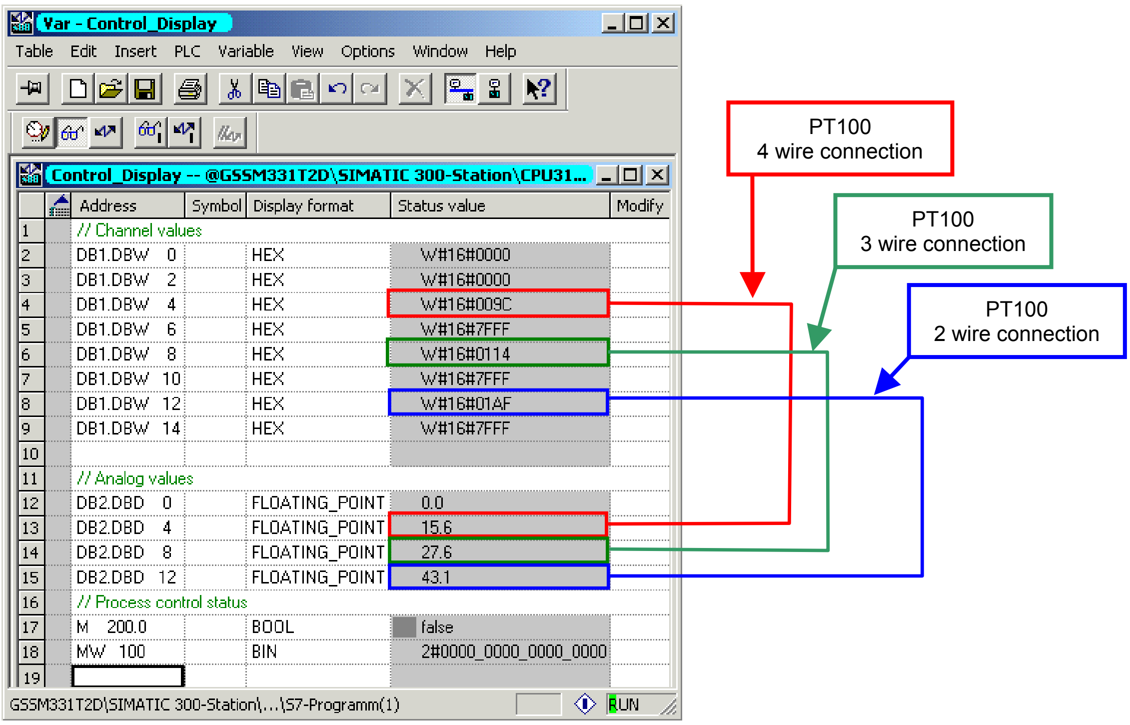


Figure 7-6 Direct comparison of the three connection types

Note

The wire resistance does not increase with the temperature. It remains constant. If you measure high temperatures, the relative failure is smaller.

8 Diagnostic interrupt

Diagnostic interrupts enable the user program to react on hardware failures.

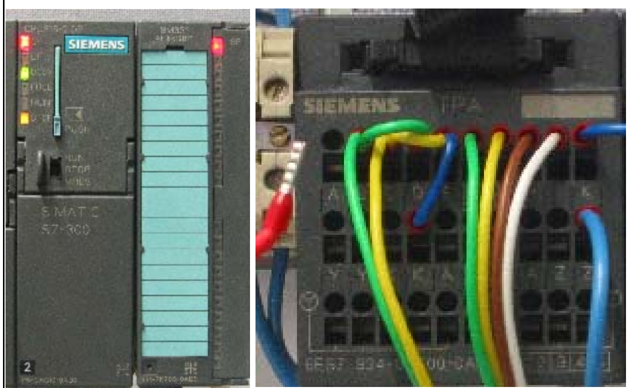
Modules must have diagnostic capabilities in order to be able to generate diagnostic interrupts.

In OB82 you program the reaction on diagnostic interrupts.

8.1 Triggering Diagnostic Interrupt

The analog input module SM331 AI8x12 has diagnostic capabilities. Diagnostic interrupts that occur are signalled by the red „SF“ LED on the SM331 and on the CPU.

Table 8-1 Simulation of a hardware failure

Graphic	Description
	Release the voltage supply from terminal 1 on the front connector of the module or from terminal Y of TOP connect terminal block Result: A diagnostic interrupt is triggered, the red LED “SF” is lit.

The cause for the error can be found “Online” through a query of the module information.

In order to see the module information "Online", do the following:

- Select the SM331 in the hardware configuration
- Click the menu item PLC -→ Module Information... in order to perform a hardware diagnosis.

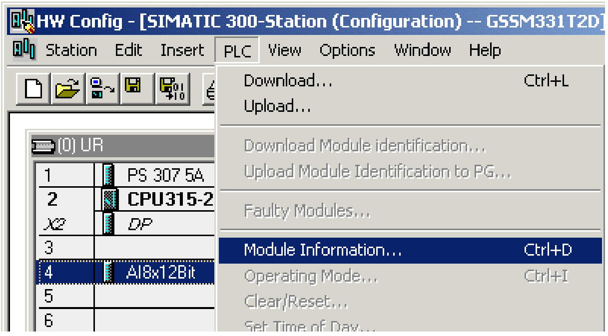


Figure 8-1 Module information

8.2 General Diagnostic Messages

You will find information on the reported error in the Diagnostic Interrupt tab. The interrupts are not channel dependent and affect the whole module.

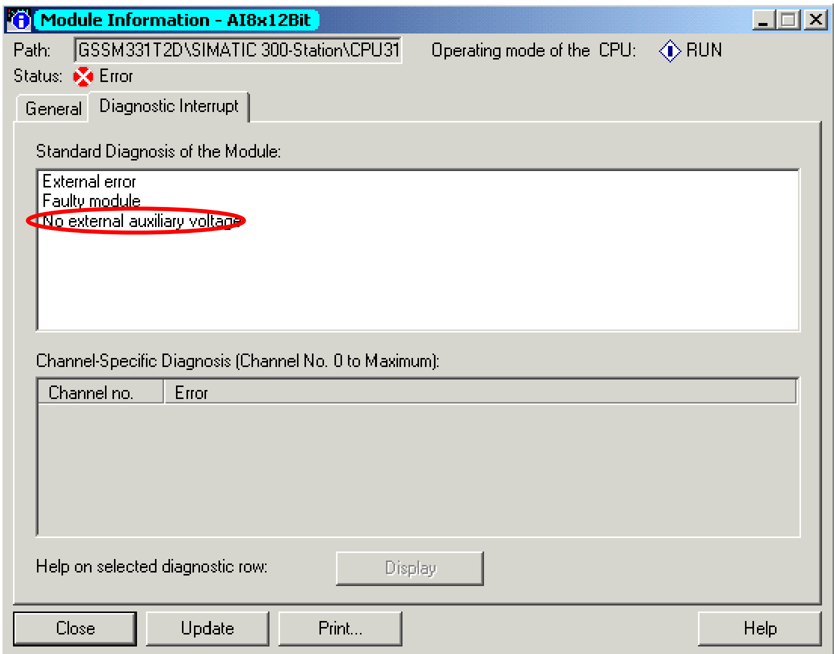


Figure 8-2 Diagnostics for SM331

8.3 Channel dependant diagnostic messages

There are five channel dependant diagnostic messages:

- Configuration or parameterisation errors
- Common mode errors
- Wire break (not for the voltage measurement type)
- Underflow
- Overflow

Note

Here we only show you the channel specific diagnostics for the measuring mode PT100 and voltage. Other measuring modes are similar but not described here.

8.3.1 Configuration / Parameterisation errors

The position of the measuring range module does not match the measuring mode set in the hardware configuration.

8.3.2 Common mode errors

The voltage difference U_{cm} between the inputs (M-) and the common voltage potential of the measuring circuit (M_{ana}) is too high.

In our example this failure cannot occur, because for a voltage transducer M_{ana} is connected to M-.

8.3.3 Wire break (only for measurement type PT100)

If the measurement type is set to voltage, there is no way to check for a wire break. You cannot choose this in the SIMATIC Manager either (see Figure 6-10, input 0-1). For the measurement type PT100 a wire break is detected and reported.

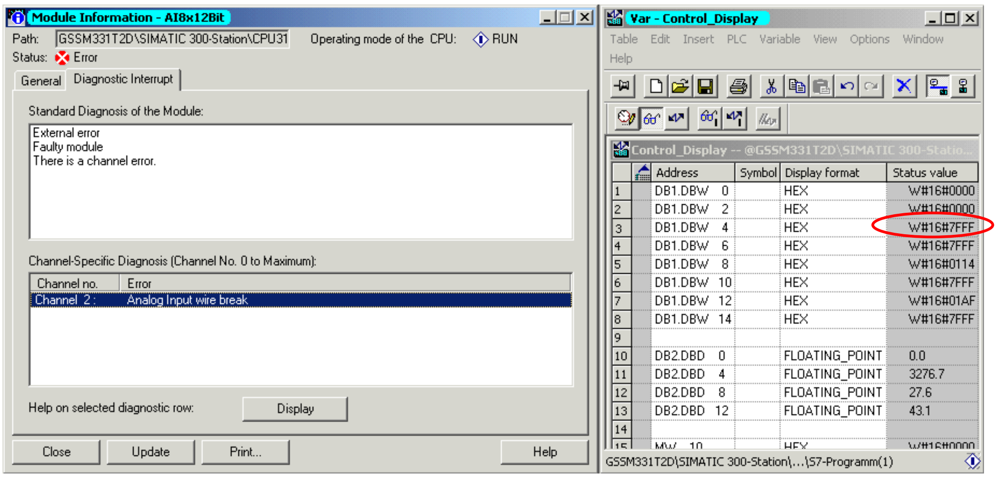


Figure 8-3 Left: Diagnostic message for wire break / Right: Variable table

The analog value display immediately shows overflow (HEX 7FFF) because an infinite resistance is measured.

8.3.4 Underflow

The two measurement types voltage and PT100 can trigger the diagnostic interrupt „Analog input measuring range / Low limit exceeded".

Voltage

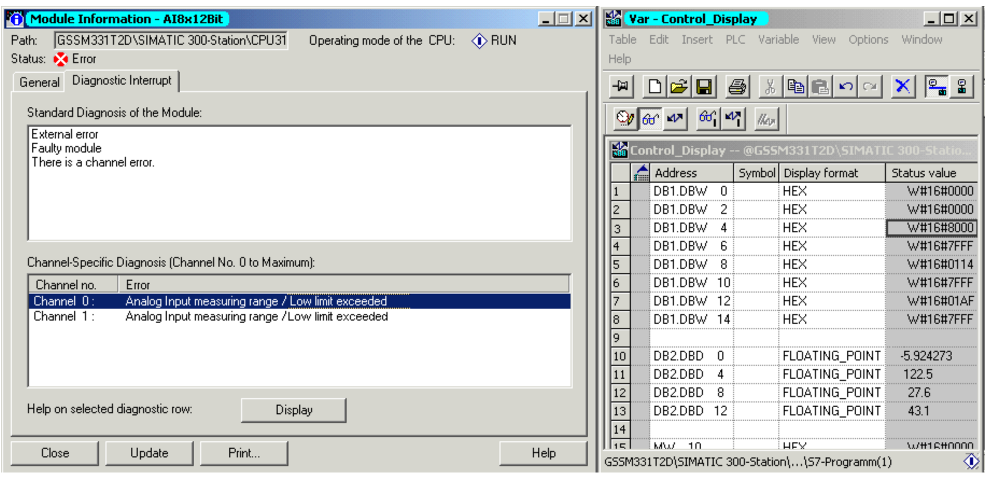


Figure 8-4 Left: Diagnostic message with underflow / Right: Variable table

We have connected the two channels in parallel to maintain the diagnostic capabilities of the channel group. Consequently we get the diagnostic message for the second channel as well.

If you get this message during commissioning, then check if the measurement range of the transducer and the parameterisation concur.

PT100

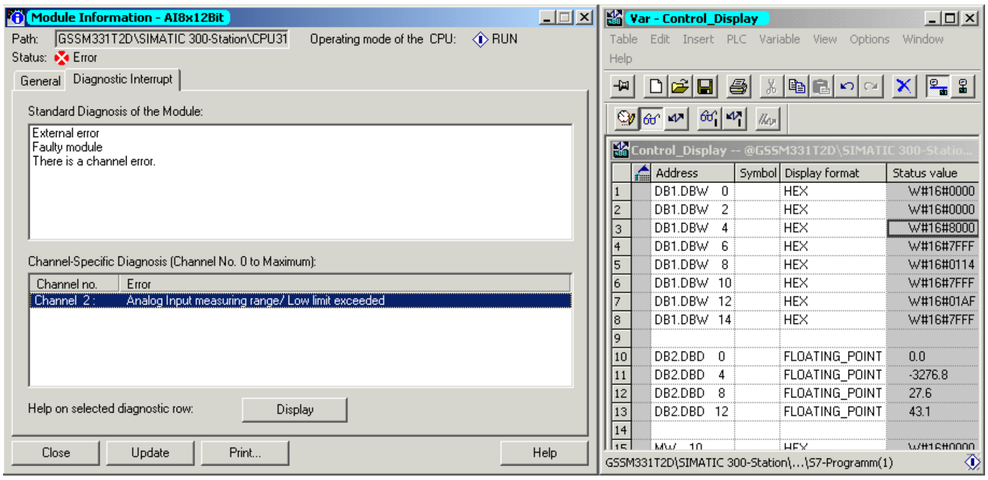


Figure 8-5 Left: Diagnostic message with underflow / Right: Variable table

This message occurs when the temperature is below -243 °C or PT100 has a very low resistance. Most likely the PT100 connection has a short circuit or it is defective.

8.3.5 Overflow

In the two measurement types voltage and PT100 the diagnostic interrupt „Analog input measuring range / High limit exceeded” can be triggered.

Measurement type voltage

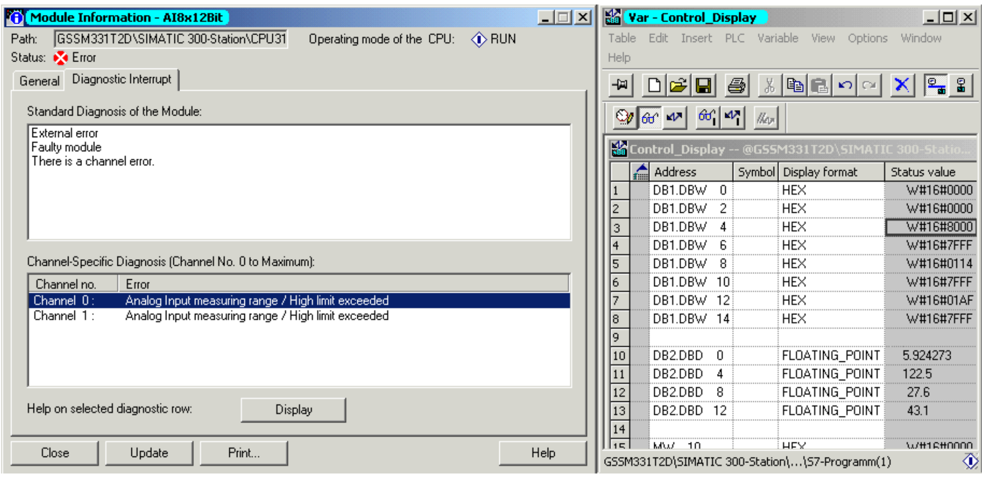


Figure 8-6 Left: Diagnostic message with overflow / Right: Variable table

Measurement type PT100

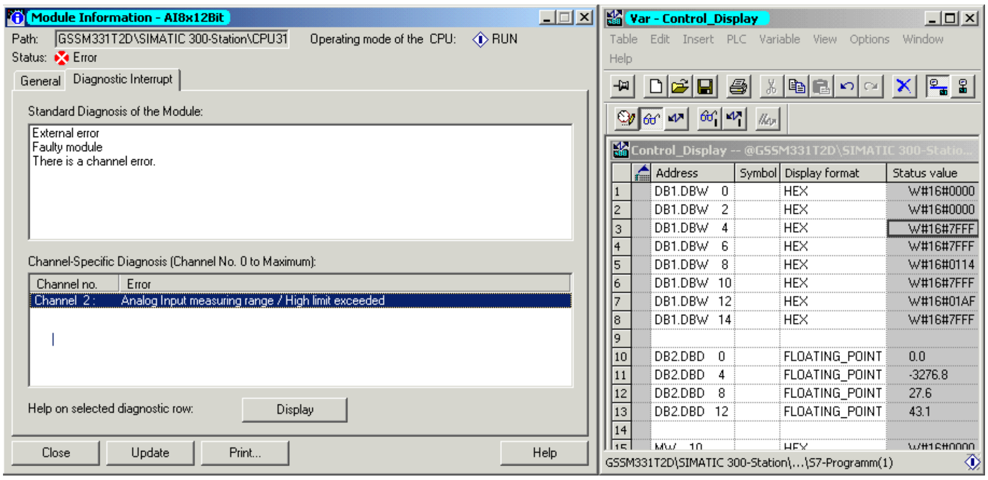


Figure 8-7 Left: Diagnostic message with overflow / Right: Variable table

9 Hardware Interrupt

A special feature of the SM331 AI8x12bit is its capability to trigger hardware interrupts. Two channels (0 and 2) can be configured that way.

Defining the limit values for hardware interrupts:

The limit values for resistance thermometer PT100 have to be defined in °C and not in °F or K.

The limit values of the voltage transducers have to be defined in Volts (V) and not in the unit of the installed sensor.

Example:
You have a pressure sensor with a physical unit of Pascal (Pa). The limit values are not entered in Pascal but in Volts of the voltage transducer.

Features of the hardware interrupt triggering

In order to trigger a hardware interrupt the limit values have to be in the nominal range of the measurement type.

Example:
You use a voltage transducer (±5V) with a nominal value of -5V and +5V. When you enter the lower limit value of -6V, this value is accepted by the system, but the hardware interrupt will never be triggered because the diagnostic interrupt (underflow of the nominal range) is always activated first.

In our example we have configured channel 0 (voltage transducer) with the following limit values:

- Lower limit value: -3V
- Upper limit value: +3V

When these values within the nominal values exceed, then the hardware interrupt OB40 is triggered.

Hardware interrupt OB40

In general a hardware interrupt calls an organisation block of the CPU. In our example OB40 is called.

In STEP 7 programs OB40 is used for hardware interrupts. Depending on the CPU, additional hardware interrupts can be configured.

If a hardware interrupt occurs, OB40 is called. In the user program of OB40 you can program the reaction of the automation system on hardware interrupts.

In the sample user program, OB40 reads the cause of the hardware interrupt. This can be found in the temporary variable structure OB40_POINT_ADDR (Local bytes 8 to 11).

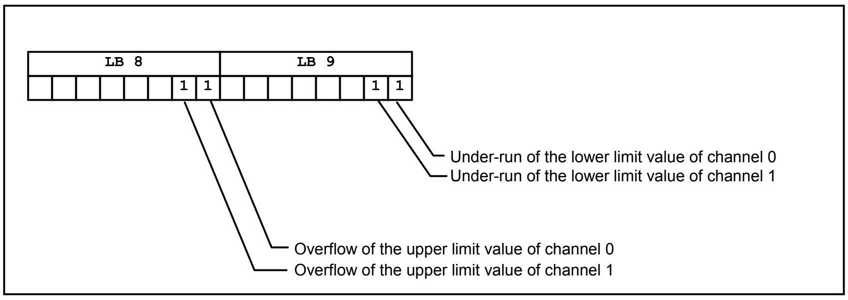


Figure 9-1 Start-up information of OB40: Which event has triggered the hardware interrupt for which limit value

In the example, OB40 only transfers the local data variables LB8 and LB9 into a memory word (MW100). The marker word is monitored in the existing variable table.

You can acknowledge the memory word in OB1 by setting marker bit M200.0 or by setting the marker to TRUE in the variable table.

Simulation of a hardware interrupts

When you supply channel 0 with 4V using a calibrator, you get the binary value 0000 0001 0000 0000 for MW100 in the variable table. This means OB40 was called and an overflow of the upper limit value of >4V has occurred in channel 0.

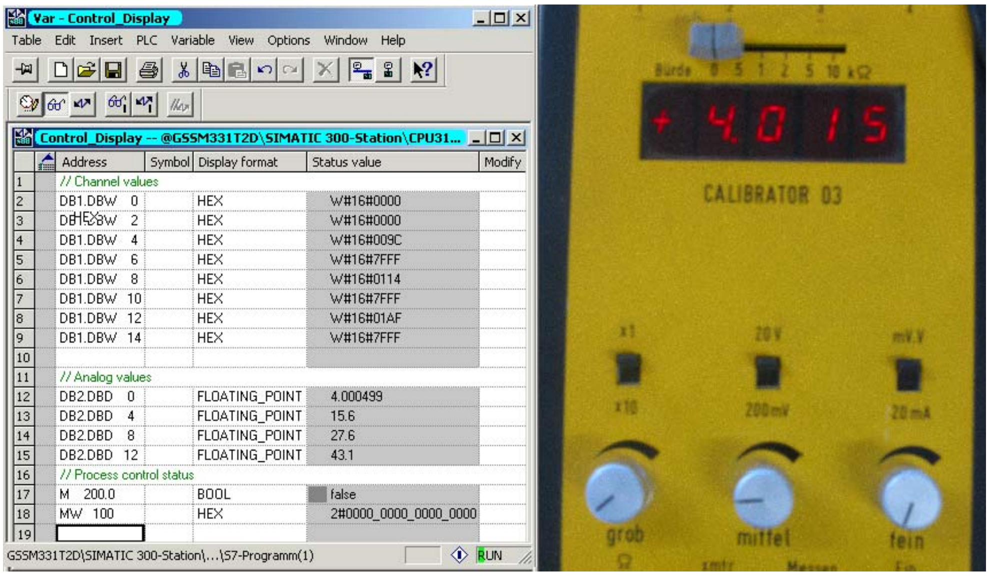


Figure 9-2 Hardware interrupt: Lower limit value underflow in Channel 0

10 Source code of the user program

In this chapter you will be provided with a quick overview of the user program's functions of the sample station. A flow chart will show you a general program structure. The complete program is listed in detail in the STL source code.

You can also download the source code as an STL file directly from the HTML page from which you have loaded this "Getting started".

Flow chart

The text marked in red is the source code of the user program.

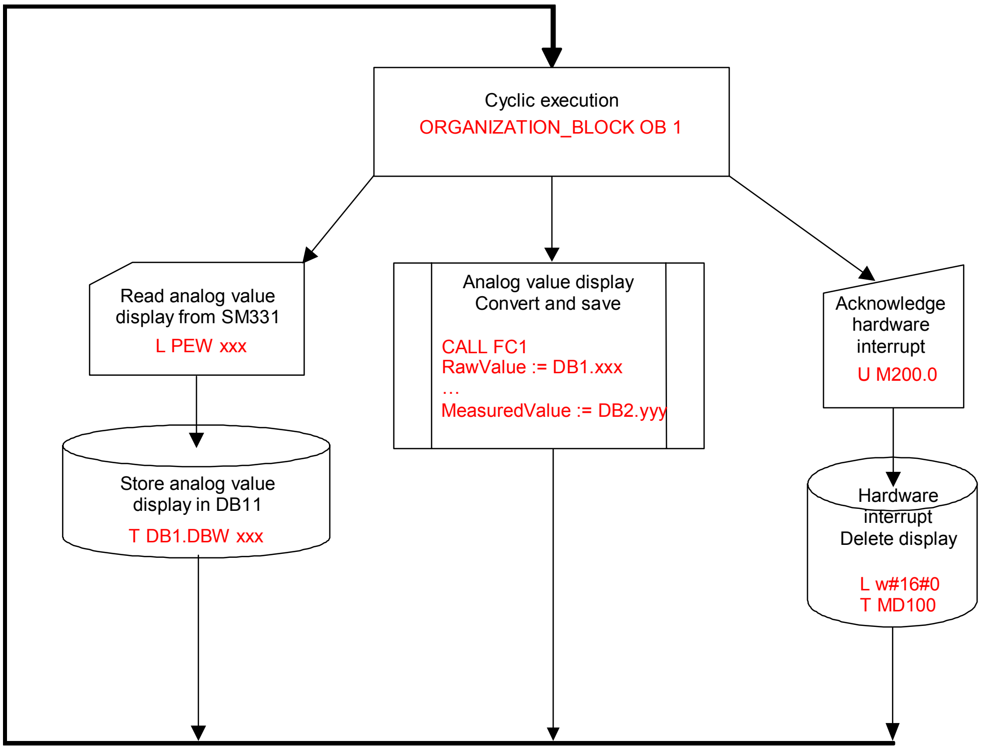


Figure 10-1 OB1 Flow chart

Variable description

Table 10-1 Variable description

Variable	Description
DB1.DBW 0	Channel 0 Analog value display
DB1.DBW 2	Channel 1 Analog value display
DB1.DBW 4	Channel 2 Analog value display
DB1.DBW 6	Channel 3 Analog value display
DB1.DBW 8	Channel 4 Analog value display
DB1.DBW 10	Channel 5 Analog value display
DB1.DBW 12	Channel 6 Analog value display
DB1.DBW 14	Channel 7 Analog value display
DB2.DBD 0	Transducer1 voltage (V)
DB2.DBD 4	PT100 with 4 wire connection (°C)
DB2.DBD 8	PT100 with 3 wire connection (°C)
DB2.DBD 12	PT100 with 2 wire connection (°C)
M200.0	Acknowledge hardware interrupt
MW 100	Hardware interrupt status

STL source code

```
DATA_BLOCK DB 1
TITLE =Analog value display
VERSION : 0.1


  STRUCT
   CH_0 : INT ; //Channel 0
   CH_1 : INT ; //Channel 1
   CH_2 : INT ; //Channel 2
   CH_3 : INT ; //Channel 3
   CH_4 : INT ; //Channel 4
   CH_5 : INT ; //Channel 5
   CH_6 : INT ; //Channel 6
   CH_7 : INT ; //Channel 7
  END_STRUCT ;
BEGIN
   CH_0 := 0;
   CH_1 := 0;
   CH_2 := 0;
   CH_3 := 0;
   CH_4 := 0;
   CH_5 := 0;
   CH_6 := 0;
   CH_7 := 0;
END_DATA_BLOCK

DATA_BLOCK DB 2
TITLE =Process values
VERSION : 0.1

  STRUCT
   SE_1 : REAL ;     //Voltage Transducer
   SE_2 : REAL ;     //PT100 (4)
   SE_3 : REAL ;     //PT100 (3)
   SE_4 : REAL ;     //PT100 (2)
```

```
  END_STRUCT ;
BEGIN
   SE_1 := 0.000000e+000;
   SE_2 := 0.000000e+000;
   SE_3 := 0.000000e+000;
   SE_4 := 0.000000e+000;
END_DATA_BLOCK

FUNCTION FC 1 : VOID
TITLE =Convert analog value display to process values
VERSION : 0.1


VAR_INPUT
  RawValue : INT ;
  Factor : REAL ;
  Offset : REAL ;
  OverFlow : INT ;
  OverRange : INT ;
  UnderRange : INT ;
  UnderFlow : INT ;
END_VAR
VAR_OUTPUT
  MeasuredValue : REAL ;
  Status : WORD ;
END_VAR
VAR_TEMP
  TInt : INT ;
  TDoubleInt : DINT ;
  TFactor : REAL ;
  TOffset : REAL ;
  TFactor1 : DINT ;
  TFactor2 : REAL ;
END_VAR
BEGIN
NETWORK
TITLE =Conversion

      L     #RawValue;
      ITD   ;
      DTR   ;
      L     #Factor;
      *R    ;
      L     #Offset;
      +R    ;
      T     #MeasuredValue;

NETWORK
TITLE =Analog value display monitoring

      L     W#16#0;
      T     #Status;


      L     #RawValue;
      L     #OverFlow;
      >=I   ;
      SPB   m_of;

      L     #RawValue;
      L     #OverRange;
      >=I   ;
      SPB   m_or;

      L     #RawValue;
      L     #UnderFlow;
      <=I   ;
      SPB   m_uf;

      L     #RawValue;
      L     #UnderRange;
      <=I   ;
```

```
      SPB   m_ur;

      SPA   end;

m_of: L     W#16#800;
      T     #Status;
      SPA   end;

m_or: L     W#16#400;
      T     #Status;
      SPA   end;

m_uf: L     W#16#200;
      T     #Status;
      SPA   end;

m_ur: L     W#16#100;
      T     #Status;
      SPA   end;

end:  NOP   0;
END_FUNCTION


ORGANIZATION_BLOCK OB 1
TITLE = "Main Program Sweep (Cycle)"
VERSION : 0.1


VAR_TEMP
  OB1_EV_CLASS : BYTE ; //Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event
class 1)
  OB1_SCAN_1 : BYTE ;   //1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of
OB 1)
  OB1_PRIORITY : BYTE ; //Priority of OB Execution
  OB1_OB_NUMBR : BYTE ; //1 (Organization block 1, OB1)
  OB1_RESERVED_1 : BYTE ;   //Reserved for system
  OB1_RESERVED_2 : BYTE ;   //Reserved for system
  OB1_PREV_CYCLE : INT ;    //Cycle time of previous OB1 scan (millisec-
onds)
  OB1_MIN_CYCLE : INT ; //Minimum cycle time of OB1 (milliseconds)
  OB1_MAX_CYCLE : INT ; //Maximum cycle time of OB1 (milliseconds)
  OB1_DATE_TIME : DATE_AND_TIME ;   //Date and time OB1 started
END_VAR
BEGIN
NETWORK
TITLE =Transfer of channel values into data block DB1

// Channel 0 -> Data block
      L     PEW  256;
      T     DB1.DBW    0;

// Channel 1 -> Data block
      L     PEW  258;
      T     DB1.DBW    2;

// Channel 2 -> Data block
      L     PEW  260;
      T     DB1.DBW    4;

// Channel 3 -> Data block
      L     PEW  262;
      T     DB1.DBW    6;

// Channel 4 -> Data block
      L     PEW  264;
      T     DB1.DBW    8;

// Channel 5 -> Data block
      L     PEW  266;
      T     DB1.DBW   10;
```

```
// Channel 6 -> Data block
      L     PEW  268;
      T     DB1.DBW   12;

// Channel 7 -> Data block
      L     PEW  270;
      T     DB1.DBW   14;


NETWORK
TITLE =Converting analog value display -> Measurement value

// Channel 1 : Voltage transducer 1 to 5V

      CALL FC     1 (
           RawValue                 := DB1.DBW    0,
           Factor                   := 1.447000e-004,
           Offset                   := 1.000000e+000,
           OverFlow                 := 32512,
           OverRange                := 27649,
           UnderRange               := -1,
           UnderFlow                := -4865,
           MeasuredValue            := DB2.DBD    0,
           Status                   := MW     10);

// Channel 2 : PT100

      CALL FC     1 (
           RawValue                 := DB1.DBW    4,
           Factor                   := 1.000000e-001,
           Offset                   := 0.000000e+000,
           OverFlow                 := 10001,
           OverRange                := 8501,
           UnderRange               := -2001,
           UnderFlow                := -2431,
           MeasuredValue            := DB2.DBD    4,
           Status                   := MW     20);

// Channel 3 : PT100

      CALL FC     1 (
           RawValue                 := DB1.DBW    8,
           Factor                   := 1.000000e-001,
           Offset                   := 0.000000e+000,
           OverFlow                 := 10001,
           OverRange                := 8501,
           UnderRange               := -2001,
           UnderFlow                := -2431,
           MeasuredValue            := DB2.DBD    8,
           Status                   := MW     30);

// Channel 4: PT 100

      CALL FC     1 (
           RawValue                 := DB1.DBW   12,
           Factor                   := 1.000000e-001,
           Offset                   := 0.000000e+000,
           OverFlow                 := 10001,
           OverRange                := 8501,
           UnderRange               := -2001,
           UnderFlow                := -2431,
           MeasuredValue            := DB2.DBD   12,
           Status                   := MW     40);

NETWORK
TITLE = Acknowledge hardware interrupt

      U     M     200.0;
      FP    M     200.1;
      SPBN  m001;
      L     0;
```

```
      T     MD   100;
      T     MW   104;
      T     MW   106;
      R     M    200.0;
m001: NOP   0;

END_ORGANIZATION_BLOCK


ORGANIZATION_BLOCK OB 40
TITLE = "Hardware Interrupt"
VERSION : 0.1


VAR_TEMP
  OB40_EV_CLASS : BYTE ;    //Bits 0-3 = 1 (Coming event), Bits 4-7 = 1
(Event class 1)
  OB40_STRT_INF : BYTE ;    //16#41 (OB 40 has started)
  OB40_PRIORITY : BYTE ;    //Priority of OB Execution
  OB40_OB_NUMBR : BYTE ;    //40 (Organization block 40, OB40)
  OB40_RESERVED_1 : BYTE ;  //Reserved for system
  OB40_IO_FLAG : BYTE ; //16#54 (input module), 16#55 (output module)
  OB40_MDL_ADDR : WORD ;    //Base address of module initiating interrupt
  OB40_POINT_ADDR : DWORD ; //Interrupt status of the module
  OB40_DATE_TIME : DATE_AND_TIME ;  //Date and time OB40 started
END_VAR
BEGIN
NETWORK
TITLE =

      L     #OB40_IO_FLAG; // OB40_IO_FLAG    : 16#54 = Input module
      T     MB   104; //                 : 16#55 = Output module

      L     #OB40_MDL_ADDR; // OB40_MDL_ADDR   : Start address of the
      T     MW   106; //                   triggering module

      L     #OB40_POINT_ADDR; // OB40_POINT_ADDR : LB8 = Exceeding high
                              // limit value
      T     MD   100; //

      NOP   0; // OB40_POINT_ADDR : LB9 = Under-run of low limit value
      NOP   0; //

END_ORGANIZATION_BLOCK

ORGANIZATION_BLOCK OB 82
TITLE = "I/O Point Fault"
VERSION : 0.1


VAR_TEMP
  OB82_EV_CLASS : BYTE ;    //16#39, Event class 3, Entering event state,
                            // Internal fault event
  OB82_FLT_ID : BYTE ;  //16#XX, Fault identifcation code
  OB82_PRIORITY : BYTE ;    //Priority of OB Execution
  OB82_OB_NUMBR : BYTE ;    //82 (Organization block 82, OB82)
  OB82_RESERVED_1 : BYTE ;  //Reserved for system
  OB82_IO_FLAG : BYTE ; //Input (01010100), Output (01010101)
  OB82_MDL_ADDR : WORD ;    //Base address of module with fault
  OB82_MDL_DEFECT : BOOL ;  //Module defective
  OB82_INT_FAULT : BOOL ;   //Internal fault
  OB82_EXT_FAULT : BOOL ;   //External fault
  OB82_PNT_INFO : BOOL ;    //Point information
  OB82_EXT_VOLTAGE : BOOL ; //External voltage low
  OB82_FLD_CONNCTR : BOOL ; //Field wiring connector missing
  OB82_NO_CONFIG : BOOL ;   //Module has no configuration data
  OB82_CONFIG_ERR : BOOL ;  //Module has configuration error
  OB82_MDL_TYPE : BYTE ;    //Type of module
  OB82_SUB_MDL_ERR : BOOL ; //Sub-Module is missing or has error
  OB82_COMM_FAULT : BOOL ;  //Communication fault
  OB82_MDL_STOP : BOOL ;    //Module is stopped
  OB82_WTCH_DOG_FLT : BOOL ;    //Watch dog timer stopped module
```

```
  OB82_INT_PS_FLT : BOOL ;  //Internal power supply fault
  OB82_PRIM_BATT_FLT : BOOL ;   //Primary battery is in fault
  OB82_BCKUP_BATT_FLT : BOOL ;  //Backup battery is in fault
  OB82_RESERVED_2 : BOOL ;  //Reserved for system
  OB82_RACK_FLT : BOOL ;    //Rack fault, only for bus interface module
  OB82_PROC_FLT : BOOL ;    //Processor fault
  OB82_EPROM_FLT : BOOL ;   //EPROM fault
  OB82_RAM_FLT : BOOL ; //RAM fault
  OB82_ADU_FLT : BOOL ; //ADU fault
  OB82_FUSE_FLT : BOOL ;    //Fuse fault
  OB82_HW_INTR_FLT : BOOL ; //Hardware interupt input in fault
  OB82_RESERVED_3 : BOOL ;  //Reserved for system
  OB82_DATE_TIME : DATE_AND_TIME ;  //Date and time OB82 started
END_VAR
BEGIN
END_ORGANIZATION_BLOCK
```