SIMATIC

S7-300
SM331; AI 8x12 Bit Getting Started
part 1: 4 -20mA

Getting Started
Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

| **DANGER** | indicates that death or severe personal injury will result if proper precautions are not taken. |
| **WARNING** | indicates that death or severe personal injury may result if proper precautions are not taken. |
| **CAUTION** | with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken. |
| **CAUTION** | without a safety alert symbol, indicates that property damage can result if proper precautions are not taken. |
| **NOTICE** | indicates that an unintended result or situation can occur if the corresponding information is not taken into account. |

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The device/system may only be set up and used in conjunction with this documentation. Commissioning and operation of a device/system may only be performed by qualified personnel. Within the context of the safety notes in this documentation qualified persons are defined as persons who are authorized to commission, ground and label devices, systems and circuits in accordance with established safety practices and standards.

Proper use of Siemens products

Note the following:

| **WARNING** | Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be adhered to. The information in the relevant documentation must be observed. |

Trademarks

All names identified by © are registered trademarks of the Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.
# Table of contents

1 Preface ................................................................................................................................................................ 5
   1.1 General .................................................................................................................................................. 5

2 Requirements .................................................................................................................................................... 7
   2.1 Basics.................................................................................................................................................. 7

3 Introduction .................................................................................................................................................. 9
   3.1 Example of an application .................................................................................................................... 9

4 Mechanical setup of the example station .................................................................................................... 11
   4.1 Mounting the example station ............................................................................................................. 11
   4.2 Mounting of analog module components ............................................................................................ 13
      4.2.1 General .................................................................................................................................... 13
      4.2.2 Components of the SM331 ....................................................................................................... 14
      4.2.3 Features of the analog modules ............................................................................................... 15
      4.2.4 Measuring range modules ....................................................................................................... 16
      4.2.5 Mounting the SM331 module .................................................................................................. 18

5 Electrical connection .................................................................................................................................. 19
   5.1 Overview .......................................................................................................................................... 19
   5.2 Wiring the power supply module and the CPU .................................................................................... 20
   5.3 Wiring of the analog module ................................................................................................................ 22
      5.3.1 Requirement ............................................................................................................................ 22
      5.3.2 Current transducer wiring - principle ....................................................................................... 22
      5.3.3 Wiring of the analog module .................................................................................................. 23
      5.3.4 Test ......................................................................................................................................... 26

6 Configuration of the SIMATIC Manager ....................................................................................................... 27
   6.1 Creating a new STEP7 project .............................................................................................................. 27
      6.1.1 Creating a new project .............................................................................................................. 27
      6.1.2 CPU selection .......................................................................................................................... 29
      6.1.3 Defining the basic user program ............................................................................................... 30
      6.1.4 Assigning the project name ...................................................................................................... 31
      6.1.5 Result S7 project is created .................................................................................................... 32
   6.2 Hardware configuration .......................................................................................................................... 33
      6.2.1 Creating the hardware configuration ......................................................................................... 33
      6.2.2 Adding SIMATIC components ............................................................................................... 34
      6.2.3 Configuring the analog module ............................................................................................... 36
      6.2.4 Test ......................................................................................................................................... 39
   6.3 STEP 7 user program ............................................................................................................................ 43
      6.3.1 Tasks of the user program ........................................................................................................ 43
      6.3.2 Creating a user program ......................................................................................................... 44
Table of contents

7 Testing the user program ................................................................. 51
  7.1 Downloading system data and user program .................................. 51
  7.2 Visualization of the sensor values ............................................... 53
  7.3 Analog value representation ....................................................... 57
8 Diagnostic interrupt ....................................................................... 59
  8.1 Reading diagnostic information from a PG .................................. 59
  8.2 General diagnostics .................................................................... 61
  8.3 Channel dependent diagnostic interrupts ...................................... 62
    8.3.1 There are five channel dependent diagnostic interrupts ........ 62
    8.3.2 Configuration / programming error ..................................... 62
    8.3.3 Common mode error ......................................................... 62
    8.3.4 Wire break ........................................................................ 63
    8.3.5 Underflow ......................................................................... 63
    8.3.6 Overflow .......................................................................... 64
9 Hardware interrupt ......................................................................... 65
  9.1 Hardware interrupt ...................................................................... 65
A Appendix ......................................................................................... 67
  A.1 Source of the user program ....................................................... 67
Index ................................................................................................. 73
1

Preface

1.1 General

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 configuration of the hardware of a 4-20mA sensor and the configuration with SIMATIC S7 Manager.

The intended readership of Getting Started is a novice with only basic experience in configuration, commissioning and servicing of automation systems.

What to expect

The procedures, from mounting the module to storing analog values in the STEP7 user program, are explained step-by-step and in detail based on an example. In the following sections you will be introduced to:

- Problem analysis
- Mechanical setup of the example station
- Electrical connection of the example station
- Configure hardware with SIMATIC Manager
- Creating a small user program with STEP7 which stores the read analog values in a data block
- Triggering and interpreting diagnostic and hardware interrupts
Preface

1.1 General
Requirements

2.1 Basics

Basic Knowledge Required

No special knowledge of the field of automation technology is required in order to understand the Getting Started guide. As the configuration of the analog module is done with the software STEP7, proficiency in STEP7 would be advantageous.

Further information on STEP7 can be found in the electronic manuals that are supplied with STEP7.

You will also need to know how to use computers or PC-like equipment (such as programming devices) under Windows 95/98/2000/NT or XP.

Required hardware and software

The scope of delivery of the analog module consists of 2 parts:

- The module itself
- A front connector, which enables you to conveniently connect the power supply and the data cables.

Analog module components

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Article</th>
<th>Order number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SM 331, Electrically ISOLATED 8 AI, ALARM DIAGNOSTICS</td>
<td>6ES7331-7KF02-0AB0</td>
</tr>
<tr>
<td>1</td>
<td>20-pin FRONT CONNECTOR with spring contacts</td>
<td>6ES7392-1BJ00-0AA0</td>
</tr>
</tbody>
</table>

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

SIMATIC components of the example station

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Article</th>
<th>Order number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PS 307 Power Supply AC 120/230V, DC 24V, 5A</td>
<td>6ES7307-1EA00-0AA0</td>
</tr>
<tr>
<td>1</td>
<td>CPU 315-2 DP</td>
<td>6ES7315-2AG10-0AB0</td>
</tr>
<tr>
<td>1</td>
<td>MICRO MEMORY CARD, NFLASH, 4 MB</td>
<td>6ES7953-8LM00-0AA0</td>
</tr>
<tr>
<td>1</td>
<td>SIMATIC S7-300, RAIL L=530MM</td>
<td>6ES7390-1AF30-0AA0</td>
</tr>
<tr>
<td>1</td>
<td>Programming device (PD) with MPI interface and MPI cable PC</td>
<td>depending on the configuration</td>
</tr>
<tr>
<td></td>
<td>with corresponding interface card</td>
<td></td>
</tr>
</tbody>
</table>
Software STEP7

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Article</th>
<th>Order number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>STEP7 Software version 5.2 or later, installed on the programming device.</td>
<td>6ES7810-4CC06-0YX0</td>
</tr>
</tbody>
</table>

The following current transducers can be used for the acquisition of analog signals:

Current transducers

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Article</th>
<th>Order number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2-Wire current transducer</td>
<td>depending on the manufacturer</td>
</tr>
<tr>
<td>1</td>
<td>4-Wire current transducer</td>
<td>depending on the manufacturer</td>
</tr>
</tbody>
</table>

Note

This "Getting Started" describes only the application of 4 – 20 mA current transducers in the 2-Wire or 4-Wire model. If you wish to use other transducers, you will need to wire and configure the SM331 differently.

General tools and materials:

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Article</th>
<th>Order number</th>
</tr>
</thead>
<tbody>
<tr>
<td>various</td>
<td>M6-bolts and nuts (Length depending on the mounting position)</td>
<td>commonly available</td>
</tr>
<tr>
<td>1</td>
<td>Screwdriver with 3,5 mm blade</td>
<td>commonly available</td>
</tr>
<tr>
<td>1</td>
<td>Screwdriver with 4.5 mm blade</td>
<td>commonly available</td>
</tr>
<tr>
<td>1</td>
<td>Side cutters and wire stripping tools</td>
<td>commonly available</td>
</tr>
<tr>
<td>1</td>
<td>Tool for crimping wire-end ferrules</td>
<td>commonly available</td>
</tr>
<tr>
<td>X m</td>
<td>Cable for grounding the mounting rail with 10 mm² cross-section, ring terminal with 6.5 mm hole, length appropriate for local requirements.</td>
<td>commonly available</td>
</tr>
<tr>
<td>X m</td>
<td>Flexible wire with 1mm² diameter with fitting wire end sleeves, form A in 3 different colors – blue, red and green</td>
<td>commonly available</td>
</tr>
<tr>
<td>X m</td>
<td>3-wire power cord (AC 230/120V) with protective contact socket, length according to local conditions.</td>
<td>commonly available</td>
</tr>
<tr>
<td>1</td>
<td>Calibration device (measuring instrument for commissioning, that can measure and supply current)</td>
<td>depends on the manufacturer</td>
</tr>
</tbody>
</table>
Introduction

3.1 Example of an application

Overview

You want to connect three analog inputs to your station. One of them should have a 2-wire current transducer and the other two should share a 4-wire current transducer.

You need failure diagnostic capabilities and want two sensors to be able to trigger hardware interrupts.

You have the analog input module SM331, AI8x12 Bit (order number 6ES7 331-7KF02-0AB0) available. The module is diagnostic and hardware interrupt capable and can process up to 8 analog inputs. The module is diagnostic and hardware interrupt capable and can process up to 8 analog inputs (e.g. 4-20 mA; PT 100; thermocouple).

Figure 3-1 Sample station components
Introduction
3.1 Example of an application

In the following sections you will be introduced to:

- Mechanical setup of the example station
  - General mounting instructions for S7-300 modules
  - Configuration of the SM331 for the two selected measurement transducer types
- Electrical connection of the example station
  - Wiring the power supply module and the CPU
  - Wiring of the analog module
  - Standard pin assignment of two measurement transducer types
  - Wiring of unused inputs
- Configuring the SIMATIC Manager
  - Using the project wizard
  - Completing the automatically generated hardware configuration
  - Integrating the supplied user program source
- User program testing
  - Interpreting the read values
  - Converting the measured values into readable analog values
- Utilizing the diagnostic capabilities of the SM331 module
  - Triggering a diagnostic interrupt
  - Evaluating the diagnostics:
- Application of hardware interrupts
  - Configuration of hardware interrupts
  - Configuration and analysis of hardware interrupts
Mechanical setup of the example station

4.1 Mounting the example station

Overview

The setup of the example 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, the mounting of it is described.

Requirements

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

The order of the mounting takes place from left to right:

- Power supply PS307
- CPU 315-2DP
- SM331
## Mechanical setup of the example station

### 4.1 Mounting the example station

### Instructions (without SM331)

<table>
<thead>
<tr>
<th>step</th>
<th>Graphic controller</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Screw on the mounting rail (screw size: M6) so that at least 40 mm space remains above and below the rail. When mounting it on a grounded steel panel or on a grounded device mounting panel made of steel sheet, make sure you have a low impedance connection between the mounting rail and the mounting surface.</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Connect the mounting rail with the protective conductor. An M6 protective conductor screw is provided on the mounting rail for this purpose.</td>
</tr>
</tbody>
</table>
| 3    |                    | Mounting the power supply:  
  - Hang the power supply on to the top end of the rail  
  - Screw it tight to the rail underneath |
| 4    |                    | Connect the bus connector (delivered with the SM331) to the left connector on the back of the CPU |
| 5    |                    | Mounting the CPU:  
  - Hang the CPU on to the top end of the rail  
  - Push it all the way left to the power supply  
  - Push it down  
  - Screw it tight to the rail underneath |
4.2 Mounting of analog module components

4.2.1 General

Overview

Before the actual mounting of the SM331 the module has to be completed with a front connector and the desired measurement mode of the inputs is set. In this section, you will learn about:

- The components you need
- The properties of the analog input module
- What a measuring range module is and how it is configured
- Mounting a configured module
4.2.2 Components of the SM331

Overview

A functional analog module consists of the following components:

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

![Components of the SM331](image)

The scope of delivery of SM331

<table>
<thead>
<tr>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
</tr>
<tr>
<td>Labeling strips</td>
</tr>
<tr>
<td>Bus connectors</td>
</tr>
<tr>
<td>2 cable ties (not in the picture) to tie the external wiring</td>
</tr>
</tbody>
</table>
4.2.3 Features of the analog modules

Characteristics

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

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.
4.2.4 Measuring range modules

Terminal

The module SM331 has 4 measuring range modules (one per channel group). The measuring range modules can be set to 4 different positions (A, B, C or D).

Figure 4-2 4 measuring range modules with default setting B (Voltage)
Positions of the measuring range modules

The position enables you to specify the transducer to be connected to the respective channel group.

<table>
<thead>
<tr>
<th>Position</th>
<th>Type of measurement</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Thermocouple / resistance measurement</td>
</tr>
<tr>
<td>B</td>
<td>Voltage (factory setting)</td>
</tr>
<tr>
<td>C</td>
<td>Current (4-wire transducer)</td>
</tr>
<tr>
<td>D</td>
<td>Current (2-wire transducer)</td>
</tr>
</tbody>
</table>

In our example, a sensor with a 4 to 20mA 2-wire transducer is connected to channel group 1 at input 0.

A 4-wire transducer is connected to channel group 2 at inputs 2 and 3.

Therefore, the first measuring range module should have position D and the second should have position C.

Positioning of the measuring range modules

<table>
<thead>
<tr>
<th>step</th>
<th>Graphic controller</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>With a screwdriver, pull out the two measuring range modules</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Turn the measuring range module to the desired position:</td>
</tr>
</tbody>
</table>
Mechanical setup of the example station

4.2 Mounting of analog module components

<table>
<thead>
<tr>
<th>step</th>
<th>Graphic controller</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td><img src="image1.png" alt="Image" /></td>
<td>Plug the measuring range module back into the module. In our example, the module should have the following positions: CH0,1: D CH2,3: C</td>
</tr>
</tbody>
</table>

Note
When you use a 2-wire transducer, the electrical isolation against the load voltage is lost for all the channels in the module (at least one measuring range module is set to position D)

4.2.5 Mounting the SM331 module

Proceed as follows

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

<table>
<thead>
<tr>
<th>step</th>
<th>Graphic controller</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | ![Image](image2.png) | Mounting the SM331:
  - Hang the SM311 to the top end of the rail
  - Push it all the way to the left up to the CPU
  - Push it down
  - Screw it tight to the rail underneath |

| 2    | ![Image](image3.png) | Mounting the front connector:
  - Press the upper release button of the front terminal block
  - Insert the front connector into the module until it snaps in |

The example station is now mechanically mounted.
5.1 Overview

Overview

This chapter shows you how the various parts of the example 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.

Always switch off power before you start wiring the S7-300.
5.2 Wiring the power supply module and the CPU

Overview

Figure 5-1 Wiring the power supply module and the CPU
The example station requires a power supply. The wiring is done as follows:

<table>
<thead>
<tr>
<th>Step</th>
<th>Graphic controller</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Open the front panel covers of the power supply module and CPU.</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Unscrew the cable grip on the power supply</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>Remove the insulation from the power cord, attach the cable end sleeves (for stranded conductors) and connect it to the power supply</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>Screw down the clamp of the cable grip.</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>Insert two connecting cables between the power supply and the CPU and tighten them</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>Verify that the setting of the selector switch matches your mains voltage. The default line voltage setting for the power supply module is 230 VAC. To change this setting, proceed as follows: Remove the protective cap with a screwdriver, set the selector switch to match your line voltage, then insert the protective cap again.</td>
</tr>
</tbody>
</table>
5.3 Wiring of the analog module

5.3.1 Requirement

General

The wiring of an analog measurement transducer is depends on its type and not on the SM331 module.

5.3.2 Current transducer wiring - principle

Options

Depending on the current transducer you use, you have to modify the wiring of the power supply. We differentiate between the wiring of a 2-wire current transducer and a 4-wire current transducer.

Wiring principles of a 2-wire current transducer

This transducer type is supplied with power from the analog input module.

![Wiring: 2-Wire current transducer](https://fb.com/tailieudientucntt)
Wiring principles of a 4-wire current transducer

Unlike a 2-wire transducer, this transducer has its own power supply.

Tasks

- Connecting the power supply (red cable)
- Connecting the 2-wire current transducer (green cables)
- Terminating unused channels with a resistor
- Connecting the 4-wire current transducer (green cables)
- Connecting the 4-wire current transducer (green cables)
- Connecting to ground and short-circuiting the other unused channels (blue wires)

Figure 5-3  Wiring: 4-Wire current transducer
5.3 Wiring of the analog module

SM331 Front connector wiring

![Image of SM331 Front connector wiring](image)

**NOTICE**

Possible destruction of the module!
If you connect a defective 4-wire current transducer to an input, which is configured for a 2-wire transducer, the module may be destroyed.
The required wiring tasks are explained below step-by-step:

### Proceed as follows

<table>
<thead>
<tr>
<th>Step</th>
<th>Graphic controller</th>
<th>Connecting-up</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Open the front door of the SM331</td>
<td>The connection diagram is printed on the front flap</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Remove 6 mm of the insulation from the ends of the wires that go into the front connector. Attach cable end sleeves to these ends.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>Wire the front connector as follows: Terminal 1: L+</td>
<td>Power supply of the module</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>Terminal 2: M+ sensor 1 Terminal 3: M- sensor 1</td>
<td>Standard wiring for 2-wire current transducer</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>Connect terminal 4 and 5 with a 1.5 to 3.3 kΩ resistor</td>
<td>In order to maintain the diagnostic capability of channel group 0, the second unused input must be connected to a resistor.</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>Terminal 6: M+ sensor 2 Terminal 7: M- sensor 2</td>
<td>Standard wiring for 4-wire current transducer</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>Terminal 8: M+ sensor 3 Terminal 9: M- sensor 3</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>terminal 10 (Comp) and connect terminal 11 (M_{ana}) to M</td>
<td>For measuring current comp is not used Mandatory for 2-wire current transducers Unused channel groups should be short-circuited with M_{ana} in order to achieve a maximum interference resistance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Short-circuit terminals 12 to 19 and connect with M_{ana}</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Terminal 20: M</td>
<td></td>
</tr>
</tbody>
</table>
5.3 Wiring of the analog module

5.3.4 Test

Proceed as follows

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

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

![Successful wiring, CPU in position STOP](image)

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

6.1 Creating a new STEP7 project

6.1.1 Creating a new project

"New Project" wizard

Use STEP 7 V5.2 or later for configuring the new CPU 315-2DP.

Start SIMATIC Manager by clicking the "SIMATIC Manager" icon on your Windows Desktop and create a new project with the "New Project" wizard.

![Image of the "New Project" wizard]

Figure 6-1 Starting the "New Project" wizard
A project wizard introduction window appears. The wizard guides you through the procedure for creating a new project.

![New Project wizard start](image)

**Figure 6-2  “New Project” wizard start**

The following must be specified during the creation procedure:

- The CPU type
- The basic user program
- The organization blocks
- Project name

Click "Next".
6.1.2  CPU selection

Proceed as follows

Choose the CPU 315-2DP for the example project. (You can also use our example for a different CPU. Select the appropriate CPU in this case.)

Click "Next."
6.1.3 Defining the basic user program

Proceed as follows

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

- OB1 cyclically executed 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.

If you use a module with diagnostic capabilities and OB82 is not inserted, the CPU changes to STOP mode when a diagnostic alarm occurs.

![Figure 6-4 “New Project” wizard: Inserting organization blocks](image)

Click "Next."
6.1.4 Assigning the project name

Proceed as follows

Select the "Project name" text box and overwrite the name in it with "Getting Started S7 SM331".

![Image of New Project wizard]

Figure 6-5 "New Project" wizard: Assigning the project name

Click "Finish". The basic STEP7 project is created automatically.
6.1.5 Result S7 project is created

Result

The wizard has created the project “Getting Started S7-SM331”. You can see the inserted organization blocks in the right window.

Figure 6-6 "New Project" wizard results
6.2 Hardware configuration

6.2.1 Creating the hardware configuration

Requirements

The STEP7 wizard has created a basic S7 project. You also need a complete hardware configuration in order to create the system data for the CPU.

Proceed as follows

You can create the hardware configuration of the example station with SIMATIC Manager. To do this, select the folder "SIMATIC 300 Station" in the left window. Start the hardware configuration by double clicking the folder "Hardware" in the right window.
6.2.2 Adding SIMATIC components

Proceed as follows:

First select a power supply module from the hardware catalog.

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

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

Select the PS307 5A and drag it into slot 1 (see red arrow).

![Hardware configuration: Basic configuration](image)

Result: PS 307 5A appears in the configuration of your rack.
Inserting an 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 catalog (see blue arrow).

Drag the module into the first available field at slot 4 of your rack (see red arrow).

You have inserted all the modules into the hardware configuration. In the next step, you configure the modules.
6.2.3 Configuring the analog module

Overview

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

Mounting the example station

The table shows, which parameters have to be set for our example station.

SM331 functions of the example station

<table>
<thead>
<tr>
<th>Functions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process reactions</td>
<td>• Diagnostics – enabled</td>
</tr>
<tr>
<td></td>
<td>• Hardware interrupt when limit exceeded - enabled</td>
</tr>
<tr>
<td>Encoder 1</td>
<td>• 2-Wire current transducer</td>
</tr>
<tr>
<td></td>
<td>• Group diagnostics</td>
</tr>
<tr>
<td></td>
<td>• Check for wire break</td>
</tr>
<tr>
<td></td>
<td>• Measuring range 6 mA and 18 mA</td>
</tr>
<tr>
<td>Encoder 2 &amp; 3</td>
<td>• 4-Wire current transducer</td>
</tr>
<tr>
<td></td>
<td>• Group diagnostics</td>
</tr>
<tr>
<td></td>
<td>• Wire break monitoring</td>
</tr>
<tr>
<td></td>
<td>• Limit values 6 mA and 18 mA</td>
</tr>
</tbody>
</table>

Opening the configuration

Double-click on slot 4 that has the SM331 in it.
Select the "Inputs" tab.
Configure the following functions:
• Diagnostic interrupt enabled
• Hardware interrupt enabled
• Input 0-1:
  – Type of measurement: 2DMU
  – Group diagnostics enabled
  – Wire break enabled
• Input 2-3:
  – Type of measurement: 4DMU
  – Group diagnostics enabled
  – Wire break enabled
6.2 Hardware configuration

- Input 4-5 and 6-7
  - Type of measurement: Disabled (- - -)
- Interference frequency
  - Select your power frequency (50 Hz or 60 Hz)
- Hardware interrupt trigger
  - Upper limit value 18 mA
  - Lower limit value 6 mA

![Figure 6-10 SM331: Configuration](https://fb.com/tailieudientucntt)
Explanation of the individual settings

Measuring type:
2DMU and 4DMU stand for 2-wire and 4-wire current transducers
- - - means that the channels are deactivated. If you deactivate channels, the remaining channels are processed faster.

Measuring range modules
The required setting of the measuring range module is displayed.

Interference frequency (Interference frequency suppression)
The frequency of your AC power system can interfere with the measured value, particularly when measuring in low voltage ranges and using thermocouple elements. With this parameter you specify the frequency of your power supply on site.
This parameter also influences the granularity, integration time and the basic execution period of the channel group.

- Resolution (accuracy)
The analog value is stored in a 16-bit word.

- Integration time
The module requires a certain amount of time to measure the analog signal. This time is called integration time. The higher the required accuracy is, the longer the module needs for measuring the voltage.

- Basic processing time
Besides the integration time, the module also needs a certain amount of time to provide the binary value.

Relationship between accuracy, interference frequency and integration period

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Interference frequency</th>
<th>Integration time</th>
<th>Basic processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>9 bits</td>
<td>400 Hz</td>
<td>2.5 ms</td>
<td>24 ms</td>
</tr>
<tr>
<td>12 bits</td>
<td>60 Hz</td>
<td>16.6 ms</td>
<td>136 ms</td>
</tr>
<tr>
<td>12 bits</td>
<td>20 Hz</td>
<td>20 ms</td>
<td>176 ms</td>
</tr>
<tr>
<td>14 bits</td>
<td>10 Hz</td>
<td>100 ms</td>
<td>816 ms</td>
</tr>
</tbody>
</table>

Hardware interrupt:
Only the channels 0 and 2 have hardware interrupt capabilities. You can use hardware interrupts to trigger an alarm when the analog signal exceeds its high or low limit.

Finish the hardware configuration:
Close the window with the configuration.
Compile and save the project with the command Station > Save and Compile (Ctrl+S)
This completes your hardware configuration for the project.
6.2.4 Test

Proceed as follows

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

<table>
<thead>
<tr>
<th>Step</th>
<th>Graphic controller</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><img src="image1.png" alt="Image" /></td>
<td>Erase your Micro Memory Card with a Power PG or a PC with external programming device: In SIMATIC Manager click &quot;File -&gt; S7 Memory Card &gt; Delete ...&quot;. The MCC will be deleted.</td>
</tr>
<tr>
<td>2</td>
<td><img src="image2.png" alt="Image" /></td>
<td>Switch off the power supply to the CPU. Insert the MMC into the CPU. Switch on the power supply.</td>
</tr>
<tr>
<td>3</td>
<td><img src="image3.png" alt="Image" /></td>
<td>If the CPU is in RUN mode, set it to STOP mode.</td>
</tr>
<tr>
<td>4</td>
<td><img src="image4.png" alt="Image" /></td>
<td>Switch the power supply on again. If the STOP LED blinks, the CPU requests a reset. Acknowledge this by turning the mode switch to MRES for a moment.</td>
</tr>
<tr>
<td>5</td>
<td><img src="image5.png" alt="Image" /></td>
<td>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.</td>
</tr>
</tbody>
</table>
6.2 Hardware configuration

Downloading hardware configuration

Download the hardware configuration into the CPU with HW Config.

![Image of HW Config tool]

Figure 6-11  Download CPU hardware configuration (1)

Click the "Load to module" icon (shown in the red circle).

When the dialog window "Select target module" appears, click OK.
The dialog window "Select target address" is shown. Click "OK." The system data will now be transferred to the CPU.
Starting the CPU

Switch the CPU to RUN.

If the hardware configuration was undertaken correctly, two red LEDs (RUN and DC5V) should be lit on the CPU.

Figure 6-13  CPU in error free state
6.3 STEP 7 user program

6.3.1 Tasks of the user program

Overview

In our example, the sensor values are stored in a data block. Also, the hardware interrupt status should be stored in a marker word. It should be possible to acknowledge the status information by means of a bit.

Furthermore, the channel values (values of the input words) should be stored in another data block.

The following tasks have to be performed in the user program:

- Cyclical storage of the analog input values in a data block (DB1)
- Cyclical conversion of the analog input values in floating point values (FC1) and storage in a data block (DB2)
- Acknowledgement of the hardware interrupt status when the acknowledge marker (M200.0) is TRUE.
- Save the status in a marker word (MW100) when a hardware interrupt occurs.

The structure of the user program is depicted in the following table:

<table>
<thead>
<tr>
<th>Call type</th>
<th>Responsible organization block</th>
<th>Task to be programmed</th>
<th>Used block or marker</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cyclic execution</td>
<td>OB1</td>
<td>Save analog input values</td>
<td>DB1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Convert and store the sensor values</td>
<td>FC1, DB2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Acknowledge hardware interrupt</td>
<td>M200.0</td>
</tr>
<tr>
<td>Hardware interrupt</td>
<td>OB40</td>
<td>Save status</td>
<td>MW100</td>
</tr>
<tr>
<td>triggered call</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diagnostic interrupt</td>
<td>OB82</td>
<td>Has to be implemented because a module with</td>
<td></td>
</tr>
<tr>
<td>triggered call</td>
<td></td>
<td>diagnostic capabilities is used</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>---</td>
</tr>
</tbody>
</table>

About OB82

OB82 is used for modules with diagnostic capabilities. If the diagnostic alarm is enabled for such modules, OB82 requests for diagnoses when a failure is detected (incoming and outgoing events). The operating system then calls OB 82.

In our example, we use OB82 to prevent the CPU from changing to STOP mode. You can program reactions to diagnostic interrupts in OB82.
6.3 STEP 7 user program

6.3.2 Creating a user program

Proceed as follows

There are two ways to create a user program.

- If you know how to program STEP7 SCL, then you can create and program the necessary blocks and the function blocks in the Blocks folder of STEP7.
- You can insert the user program from an SCL source into the project. In this “Getting started” we describe this method.

Creating a user program in STEP7 requires three steps:
1. Downloading the source file directly from the HTML page
2. Importing a source file
3. Compiling the source

Downloading the source file

You can download the source file directly from the HTML page from which you loaded this “Getting Started”.

The German version of the source file has the name "GSSM331T1DE.AWL".

Save the source file to your hard drive.
Importing a 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 Importing an external source
In the "Insert external source" dialog browse for the source file GSSM331T1DE.AWL, which you have already downloaded and saved on your hard disk.

Select the source file GSSM331T1DE.AWL (red arrow).

![Insert external source dialog](image)

Figure 6-15  Importing an external source

Click "Open".

---

SM331; AI 8x12 Bit Getting Started part 1:  4-20mA
Getting Started, 05/2008, A5E00253410-04
SIMATIC Manager has opened the source file. In the right window you can see the source file inserted.

![SIMATIC Manager Figure 6-16 Compiling the source code](image)

Figure 6-16  Compiling the source code
Compiling the source code

In order to create an executable STEP7 program, the STL source has to be compiled. Double-click the source file GSSM331T1DE in the Sources folder. The source code editor opens.

In the window of the source code editor you can view the source code.

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 Compiling 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
Testing the user program

7.1 Downloading system data and user program

Proceed as follows

The hardware and software are now ready. The next step is to download the system data and the user program into the automation system. To do this, proceed as follows:

**Downloading the system data and user program**

<table>
<thead>
<tr>
<th>Step</th>
<th>Graphic controller</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><img src="image" alt="SIMATIC Manager" /></td>
<td>Using the SIMATIC Manager, download the user program and the system data (containing the hardware configuration) into the CPU.</td>
</tr>
<tr>
<td>2</td>
<td><img src="image" alt="SIMATIC S7-300" /></td>
<td>Follow the instructions displayed on the screen. If all sensors are properly connected, the CPU and the SM331 do not display an error light. The status of the CPU is displayed by the green &quot;RUN&quot; light.</td>
</tr>
</tbody>
</table>
Smart Label

The labeling strips for the modules were created with Siemens S7 Smart Label (order no: 2XV9 450-1SL01-0YX0).

A labeling strip in its actual size:

Figure 7-1   S7-SmartLabel labeling strip for the example
7.2 Visualization of the sensor values

Proceed as follows

In order to visualize the sensor values, 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
Fill the new variable table as follows:

<table>
<thead>
<tr>
<th>Variables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB1.DBW 0</td>
<td>Channel 0 Display of analog value</td>
</tr>
<tr>
<td>DB1.DBW 2</td>
<td>Channel 1 Display of analog value</td>
</tr>
<tr>
<td>DB1.DBW 4</td>
<td>Channel 2 Display of analog value</td>
</tr>
<tr>
<td>DB1.DBW 6</td>
<td>Channel 3 Display of analog value</td>
</tr>
<tr>
<td>DB1.DBW 8</td>
<td>Channel 4 Display of analog value</td>
</tr>
<tr>
<td>DB1.DBW 10</td>
<td>Channel 5 Display of analog value</td>
</tr>
<tr>
<td>DB1.DBW 12</td>
<td>Channel 6 Display of analog value</td>
</tr>
<tr>
<td>DB1.DBW 14</td>
<td>Channel 7 Display of analog value</td>
</tr>
<tr>
<td>DB2.DBD 0</td>
<td>Transducer1 current (mA)</td>
</tr>
<tr>
<td>DB2.DBD 4</td>
<td>Transducer2 current (mA)</td>
</tr>
<tr>
<td>DB2.DBD 8</td>
<td>Transducer3 current (mA)</td>
</tr>
<tr>
<td>MW 100</td>
<td>Status hardware interrupt</td>
</tr>
<tr>
<td>MW 200.0</td>
<td>Acknowledge hardware interrupt</td>
</tr>
<tr>
<td>M101.0</td>
<td>Channel 0 exceeded low limit</td>
</tr>
<tr>
<td>M101.1</td>
<td>Channel 0 exceeded high limit</td>
</tr>
<tr>
<td>M101.2</td>
<td>Channel 2 exceeded low limit</td>
</tr>
<tr>
<td>M101.3</td>
<td>Channel 0 exceeded high limit</td>
</tr>
</tbody>
</table>

(1) In this area you can monitor the channel values
(2) In this area you can see the analog values
(3) In this area you can monitor and control the status signals
Monitoring values

In order to monitor values, 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.

(1) Channel values in hex format
(2) Converted analog value
(3) Status information

Figure 7-3   Online view of the variable table
Testing the user program

7.2 Visualization of the sensor values

Controlling values

To control the process acknowledgement, enter the desired value (TRUE or FALSE, depending on whether you want to activate or deactivate acknowledgement) into the column "Control Value" and click the icon with the two arrows.

Controlling variables

<table>
<thead>
<tr>
<th>Address</th>
<th>Symbol</th>
<th>Display format</th>
<th>Status value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IB1.DW 0</td>
<td>HEX</td>
<td>W41580000</td>
</tr>
<tr>
<td>2</td>
<td>IB1.DW 2</td>
<td>HEX</td>
<td>W41580010</td>
</tr>
<tr>
<td>3</td>
<td>DB1.DW 4</td>
<td>HEX</td>
<td>W4158003C</td>
</tr>
<tr>
<td>4</td>
<td>DB1.DW 6</td>
<td>HEX</td>
<td>W415807FF</td>
</tr>
<tr>
<td>5</td>
<td>DB1.DW 8</td>
<td>HEX</td>
<td>W41560114</td>
</tr>
<tr>
<td>6</td>
<td>DB1.DW 10</td>
<td>HEX</td>
<td>W415817FF</td>
</tr>
<tr>
<td>7</td>
<td>DB1.DW 12</td>
<td>HEX</td>
<td>W415601AF</td>
</tr>
<tr>
<td>8</td>
<td>DB1.DW 14</td>
<td>HEX</td>
<td>W415817FF</td>
</tr>
</tbody>
</table>

Controlling values

1. Channel value
2. Analog value
3. Status

Peculiarity in monitoring the values

While monitoring the values you will surely 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.
7.3 Analog value representation

Proceed as follows:

The analog values are only processed by the CPU in binary form. Analog input modules convert the analog process signal into a digital format (16 bit word).

Five ranges have to be taken into account when converting from digital to analog values:

Representation of analog values in current measuring ranges 4 to 20 mA

<table>
<thead>
<tr>
<th>Hex value</th>
<th>Current range</th>
<th>Comment</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7FFF</td>
<td>22.96 mA</td>
<td>Overflow</td>
<td>From hex value 16#F700 on, the sensor value is above the configured measurement value range and is no more valid.</td>
</tr>
<tr>
<td>7F00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7EFF</td>
<td>22.81 mA</td>
<td>Overload</td>
<td>This range corresponds to a tolerance band before the overflow range is reached. Within this range the resolution is not optimal though.</td>
</tr>
<tr>
<td>6C01</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6C00</td>
<td>20 mA</td>
<td>Rated range</td>
<td>The nominal range is the normal range for recording measurement values. This range guarantees optimal resolution.</td>
</tr>
<tr>
<td>5100</td>
<td>15 mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4 mA + 578.7 nA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>4 mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FFFF</td>
<td></td>
<td>Underload</td>
<td>Range corresponding to the overload range but for low values.</td>
</tr>
<tr>
<td>ED00</td>
<td>1.185 mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ECFF</td>
<td></td>
<td>Underflow</td>
<td>From hex value 16#ECFF on, the sensor value is below the configured measurement value range and is no more valid.</td>
</tr>
<tr>
<td>8000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

It is necessary to convert the binary format of the values in order to display analog process values. In our example, mA are displayed. This is done by converting the display of analog values in mA in a programmed function (FC1).

Note

In our example, we look at the values from the output of the transducer.

Using an amperemeter, you can now compare the values on the meter with the values of the analog values display. The values will be identical.
Testing the user program

7.3 Analog value representation
8.1 Reading diagnostic information from a PG

Overview

Diagnostic interrupts enable the user program to react to hardware errors. Modules must have diagnostic capabilities in order to generate diagnostic interrupts. In OB82 you program the reaction to diagnostic interrupts.

Display

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

Figure 8-1 Hardware error
The cause of the error can be determined "online" by requesting the hardware status.
In order to determine the state of module "online", proceed as follows:
Select the SM331 in the hardware configuration. Click the menu command CPU -> Module Information... in order to perform a hardware diagnostics.

Figure 8-2 Module status
8.2 General diagnostics

Diagnostic interrupt tab

On the Diagnostic Interrupt tab you will find information for the reported error. The interrupts are not channel dependent and apply to the entire module.

Figure 8-3 Diagnostics for SM331
8.3 Channel dependent diagnostic interrupts

8.3.1 There are five channel dependent diagnostic interrupts

There are five channel dependent diagnostic interrupts:

- Configuration / programming error
- Common mode error
- Wire break
- Underflow
- Overflow

Note
Here we show you only the channel specific diagnostics for the measuring modes 2 or 4-wire current transducers. Other measuring modes are similar but not described here.

8.3.2 Configuration / programming error

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

8.3.3 Common mode error

Meaning
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 error cannot occur because $M_{ana}$ is connected to M for a 2-wire transducer (fixed potential).
8.3.4 Wire break

Meaning
If wire break detection is enabled for 2-wire transducers, there will be no direct check for a wire break. The diagnostics instead reacts on the shortfall of the low limit current value.

With 4 to 20 mA current transducer, the diagnostic message “Analog input wire break” is shown in the module diagnostics when the current goes below 3.6 mA.

![Figure 8-4](image)

The display of the analog values shows an underflow (Hex 8000) immediately even if the current measured is clearly above 1.1185 mA.

Underloading 3.6 mA is only possible if wire break detection has been disabled.

8.3.5 Underflow

Meaning
The display of the analog values shows an underflow immediately even if the current measured is clearly above 1.1185 mA.
8.3.6 Overflow

Meaning

If the current exceeds 22.81 mA, an overflow message stating "Analog input measuring range / High limit exceeded" is displayed.

The display of the analog value (HEX 7FFF) is in the overflow range.

![Diagram]

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

Note

Disabled channels also have 7FFF hex as the analog display value.
Hardware interrupt

9.1 Hardware interrupt

Overview

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

Hardware interrupts generally trigger alarm organization blocks in the CPU. In our example, OB40 is called.

The limit values for hardware interrupts have to be specified in mA.

Example:

You have connected a pressure sensor with a 4-20mA transducer to channel 0. Here the limit values should be specified in mA and not in Pascal (Pa).

limit values

In order to trigger a hardware interrupt, the limit values have to be within the nominal values of the measuring mode.

Example:

If wire break detection (3.6 mA) is enabled, and you choose 3.5 mA for the low limit value, this setting is accepted by the system. A hardware interrupt will not be triggered because the diagnostic alarm is always triggered first.

In our example, 2 channels (sensor 1 and 2) are configured with the following limits:

- Lower limit value 6 mA
- Upper limit value 18 mA
Hardware interrupt

9.1 Hardware interrupt

Determining functions

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

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

<table>
<thead>
<tr>
<th>LB 8</th>
<th>LB 9</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Violation of the Lo limit at channel 0
Violation of the Lo limit at channel 1
Upper limit value violated channel 0
Upper limit value violated channel 1

Figure 9-1 OB40 start information: In the example, and triggered a hardware interrupt

In the example, OB40 only transfers LD8 and LD9 into a marker word (MW100). The marker word is monitored in the existing variable table. You can acknowledge the marker word in OB1 by setting marker bit M200.0 or by setting it to TRUE in the variable table.

If you supply 5.71 mA with a calibration device to channel 0, you will get the value 0001 hex for MW100 in the variable table. This means that OB40 was called and channel 0 exceeded its low limit value (6 mA).

Figure 9-2 Hardware interrupt: Channel 0 exceeded low limit value
Appendix

A.1 Source of the user program

STL source code

In this section you find the source code of the user program from the example.

You can download the source file directly from the HTML page from which you loaded this "Getting Started".

DATA_BLOCK DB 1
TITLE =Analog module channel values
VERSION : 1.0
STRUCT
  CH_0 : WORD ; //Channel 0
  CH_1 : WORD ; //Channel 1
  CH_2 : WORD ; //Channel 2
  CH_3 : WORD ; //Channel 3
  CH_4 : WORD ; //Channel 4
  CH_5 : WORD ; //Channel 5
  CH_6 : WORD ; //Channel 6
  CH_7 : WORD ; //Channel 7
END_STRUCT ;
BEGIN
  CH_0 := W#16#0;
  CH_1 := W#16#0;
  CH_2 := W#16#0;
  CH_3 := W#16#0;
  CH_4 := W#16#0;
  CH_5 := W#16#0;
  CH_6 := W#16#0;
  CH_7 := W#16#0;
END_DATA_BLOCK

DATA_BLOCK DB 2
TITLE =Current transducer (in mA)
VERSION : 1.0
STRUCT
  SE_1 : REAL ; //Sensor 1 current value (mA)
  SE_2 : REAL ; //Sensor 2 current value (mA)
  SE_3 : REAL ; //Sensor 3 current value (mA)
END_STRUCT ;
A.1 Source of the user program

BEGIN
  .SE_1 := 0.000000e+000;
  SE_2 := 0.000000e+000;
  SE_3 := 0.000000e+000;
END_DATA_BLOCK

FUNCTION FC 1 : VOID
TITLE = Conversion of a channel’s raw values in mA
VERSION : 1.0

VAR_INPUT
  Raw : WORD ;  // Analog value display
END_VAR

VAR_OUTPUT
  Current : REAL ;  // Current in mA
END_VAR

VAR_TEMP
  TDoubleInt : DINT ;
  TInt : INT;
END_VAR

BEGIN
  NETWORK
    TITLE = Conversion of raw values in mA
    L  #Raw;
    T  #TInt;
    // Only long integers can be converted into REAL format
    L  #TInt;
    ITD ;
    T  #TDoubleInt;  // HEX value
    DTR ;
    T  #Current;  // Current = --------------
    //  1728
    L  1.728000e+003;  // ! /
    /R ;
    T  #Current;
    //  ! /
    // ------ ------ ------- ------
    //  4  20
    L  4.000000e+000;  // Offset correction
    +R ;
    T  #Current;

END_FUNCTION

ORGANIZATION_BLOCK OB 1
TITLE = "Main Program Sweep (Cycle)"
VERSION : 1.0
Appendix
A.1 Source of the user program

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 (milliseconds)
OB1_MIN_CYCLE : INT; //Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE : INT; //Minimum cycle time of OB1 (milliseconds)
OB1_DATE_TIME : DATE_AND_TIME; //Date and time OB1 started
END_VAR
BEGIN
NETWORK
TITLE =Read channels
    // Channel values 0 to 7 are loaded and stored in DB1 (channel values)
    L  PEW 256;     //Channel 0
    T  DB1.DBW 0;
    L  PEW 258;     //Channel 1
    T  DB1.DBW 2;
    L  PEW 260;     //Channel 2
    T  DB1.DBW 4;
    L  PEW 262;     //Channel 3
    T  DB1.DBW 6;
    L  PEW 264;     //Channel 4
    T  DB1.DBW 8;
    L  PEW 266;     //Channel 5
    T  DB1.DBW 10;
    L  PEW 268;     //Channel 6
    T  DB1.DBW 12;
    L  PEW 270;     //Channel 7
    T  DB1.DBW 14;

SM331; AI 8x12 Bit Getting Started part 1: 4-20mA
Getting Started, 05/2008, A5E00253410-04

69
Appendix

A.1 Source of the user program

NETWORK
TITLE = Conversion
// Conversion of the channel’s raw data into current values (mA)
CALL FC 1 (Raw := DB1.DBW 0,
Current := DB2.DBD 0);
CALL FC 1 (Raw := DB1.DBW 4,
Current := DB2.DBD 4);
CALL FC 1 (Raw := DB1.DBW 6,
Current := DB2.DBD 8);

NETWORK
TITLE = Reset hardware interrupt
// Even though the hardware interrupt was reset by the hardware upon terminating OB40
// the value of the hardware interrupt must be reset manually
U M 200.0;
SPBN lb10;
L MW 100;
SSI 4;
T MW 100;
lbl0: NOP;

NETWORK
TITLE = The End
BE;

END_ORGANIZATION_BLOCK

ORGANIZATION_BLOCK OB 40
TITLE = "Hardware Interrupt"
// Processing OB40_POINT_ADDR (L8 to L11)
// L8 High limit value exceeded
// L9 Low limit value exceeded
VERSION : 1.0

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 =Sensor 1 (Channel 0): Lower limit value
U   L      9.0;   // Channel 0 low limit value
SPBNB  L001;
L    W#16#1;
L    MW    100;
OW   ;
T    MW    100;
L001;  NOP  0;

NETWORK
TITLE =Sensor 1 (Channel 0): Upper limit value
U   L      8.0;   // Channel 0 upper limit value
SPBNB  L002;
L    W#16#2;
L    MW    100;
OW   ;
T    MW    100;
L002;  NOP  0;

NETWORK
TITLE =Sensor 2 (Channel 2): Lower limit value
U   L      9.2;   // Channel 2 low limit value
SPBNB  L003;
L    W#16#4;
L    MW    100;
OW   ;
T    MW    100;
L003;  NOP  0;

NETWORK
TITLE =Sensor 2 (Channel 2): Upper limit value
U   L      8.2;   // Channel 2 upper limit value
SPBNB  L004;
L    W#16#8;
L    MW    100;
OW   ;
T    MW    100;
L004;  NOP  0;
Appendix

A.1 Source of the user program

NETWORK
TITLE =Sensor 3 (Channel 3): Lower limit value
// Only for demonstration purposes. Channel 3 has now hardware interrupt capabilities
U   L 9.3;  // Channel 3 low limit value
SPBNB L005;
L   W#16#10;
L   MW 100;
OW  ;
T   MW 100;
L005: NOP 0;

NETWORK
TITLE =Sensor 3 (Channel 3): Upper limit value
// Only for demonstration purposes. Channel 3 has now hardware interrupt capabilities
U   L 8.3;  // Channel 3 upper limit value
SPBNB L006;
L   W#16#20;
L   MW 100;
OW  ;
T   MW 100;
L006: NOP 0;

END_ORGANIZATION_BLOCK
Index

2
2-Wire current transducer
  wiring principle, 22

4
4-Wire current transducer
  wiring principle, 23

A
Add
  SIMATIC components, 34
Adding SIMATIC components, 34
Analog module
  Characteristics, 15
  Configuring, 36
  Inserting, 35
  Installing, 13
  The required hardware and software, 7
  Wiring, 22, 23
Analog value representation, 57
  in Current Measuring Ranges 4 to 20 mA, 57
Assigning
  Project name, 31

B
Basic processing time, 38
Bus connectors
  Insert, 12

C
Calling
  Configuration, 36
  Hardware configuration, 33
Channel group
  Not used, 25
Characteristics
  Analog module, 15
Check
  Line voltage, 21
Clear / Reset, 39
Common mode error, 52
Components
  SM331, 14
Components of the Product
  SM331 modules, 14
Configuration
  Calling, 36
  Configuration / programming error, 52
  Configuration language
    STL, 30
Configuring
  Analog module, 36
  CPU 315-2DP, 27
  Hardware configuration, 33
  with SIMATIC Manager, 27
Connect
  CPU with programming device, 39
Connecting-up
  Errors, 26
  test, 26
Control
  Values, 56
Control value, 56
Converting
  digital values to analog values, 57
CPU
  CPU with programming device, 39
  Installing, 12
  Selecting, 29
  start, 42
  Wiring the power supply, 20
CPU 315-2DP
  Configuring, 27
Create
  User Program, 44
Creating
  STEP 7 project, 27
Current transducers
  wiring principle, 22

D
Define the basic user program
  Defining, 30
  Defining the basic user program, 30
  Functions in event of hardware interrupts, 66
Index

Delete
- Micro memory card, 39
- Diagnostic functionality, 25
- Diagnostic information read from a PG, 59

Diagnostics message
- channel dependent, 62
- General, 61

Digital value
- convert to analog value, 57

Displays
- Errors, 61

Download
- source code, 67
- Source file, 44

Downloading
- Download system data and user program into the automation system, 51
- Hardware configuration, 40

Downloading system data and user program
- Downloading to automation system, 51

E
- Error display, 61
- Errors
  - Connecting-up, 26

F
- Fill out
  - Variable table, 54
- Finish
  - Hardware configuration, 38
  - Front connector, installing, 18

H
- Hardware and software
  - for analog modules, 7
- Hardware catalog opening, 34
- Hardware configuration
  - calling, 33
  - Configuring, 33
  - Downloading, 40
  - Finishing, 38
- Hardware fault find, 60
- Hardware interrupt
  - limit values, 55
- Hardware interrupt, 38, 65

I
- Importing
  - Source file, 45
- Insert
  - Bus connectors, 12
- Inserting
  - Analog module, 35
- Installation
  - Analog module, 13
  - CPU, 12
  - Front connector, 18
  - Plant, 11
  - Power supply, 12
  - SM331, 18
- Integration time, 38
- Interference frequency, 38
- Interference frequency suppression, 38

L
- Labeling strips for modules, 52
- LEDs, 26
  - green, 42
  - red, 59
- limit values
  - Hardware interrupt, 65
- Line frequency, 38
- Line voltage
  - Changing, 21
  - Check, 21
- Load power supply
  - Selecting, 34

M
- Measuring range modules, 38
- Positioning, 17
- Positions, 16
- Micro memory card
  - Deleting, 39
- Monitoring
  - Values, 55
- Mounting rail screw, 12
O

OB82, 43
Organization blocks
  Selecting, 30
Overflow, 64

P

Parameters
  set, 36
Plant
  Installing, 11
Position
  Measuring range modules, 17
Positions
  Measuring range modules, 16
Power supply
  Installing, 12
  screw, 12
  Wiring the CPU, 20
Powering-up, 39
Process Acknowledgement, 56
Project name
  Assigning, 31

R

Reading out
  Read diagnostic information from a PG, 59
Resolution, 38

S

screw
  Mounting rail, 12
  Power supply, 12
Selecting
  CPU, 29
  Load power supply, 34
  Organization blocks, 30
  Target addresses, 41
Sensor values
  Visualize, 53
Settings
  Parameters, 36
  test, 39
SIMATIC Manager
  Creating, 27
  Hardware configuration, 33
  Starting, 27

SM331
  Components, 14
  Installing, 18
SM331 Front connector
  Wiring, 25
SM331 modules
  Components of the Product, 14
Smart Label
  source code
    Compiling, 48
    Download, 67
    User Program, 67
Source code editor, 48
Source file
  Download, 44
  Import, 45
Starting
  CPU, 42
  SIMATIC Manager, 27
STEP 7 project
  Creating, 27
STEP 7 user program, 43
  Tasks, 43
STL, 30
STL source code, 67
Structure
  User Program, 43

T

Target addresses
  Selecting, 41
Terminal
  Transducer, 16
Test
  Connecting-up, 26
  Settings, 39
Testing
  User Program, 51
Transducer
  Terminal, 16
Type of measurement, 38

U

Underflow, 63
User Program
  Creating, 44
  source code, 67
  Structure, 43
  Testing, 51
Index

V

Values
  Controlling, 56
  Monitor, 59
Variable description, 54
Variable table
  Fill out, 54
Visualize
  Sensor values, 53

W

Wire break, 63
Wiring
  Analog module, 22, 23
  SM331 Front connector, 25
  Wiring the power supply and CPU, 20
wiring principle
  2-Wire current transducer, 22
  4-Wire current transducer, 23
  Current transducers, 22
Wizard *, 27