SIEMENS
Ingenuity for life

# Quality Assurance by means of Weighing, Controlling and Logging (Set 6)

SIMATIC S7-1200 + SIWAREX WP231

https://support.industry.siemens.com/cs/ww/en/view/82454336

Siemens Industry Online Support

# Warranty and Liability

**Note**

The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are used correctly. These Application Examples do not relieve you of the responsibility to use safe practices in application, installation, operation and maintenance. When using these Application Examples, you recognize that we cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice.
If there are any deviations between the recommendations provided in these Application Examples and other Siemens publications − e.g. Catalogs − the contents of the other documents have priority.

We do not accept any liability for the information contained in this document.
Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc., described in this Application Example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). The damages for a breach of a substantial contractual obligation are, however, limited to the foreseeable damage, typical for the type of contract, except in the event of intent or gross negligence or injury to life, body or health. The above provisions do not imply a change of the burden of proof to your detriment.
Any form of duplication or distribution of these Application Examples or excerpts hereof is prohibited without the expressed consent of the Siemens AG.

**Security information**

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.
In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens’ products and solutions only form one element of such a concept.
Customer is responsible to prevent unauthorized access to its plants, systems, machines and networks. Systems, machines and components should only be connected to the enterprise network or the internet if and to the extent necessary and with appropriate security measures (e.g. use of firewalls and network segmentation) in place.
Additionally, Siemens’ guidance on appropriate security measures should be taken into account. For more information about industrial security, please visit http://www.siemens.com/industrialsecurity.

Siemens’ products and solutions undergo continuous development to make them more secure. Siemens strongly recommends to apply product updates as soon as available and to always use the latest product versions. Use of product versions that are no longer supported, and failure to apply latest updates may increase customer’s exposure to cyber threats.
To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed under http://www.siemens.com/industrialsecurity.

# Table of Contents

# 1 Task

## Introduction

In the packaging industry, containers are to be filled with a specified number of pieces (e.g. of wall anchors), calculated by weight control.

Before the actual filling, it can be selected via a recipe selection, what (which wall anchor diameter) and how many are to be filled. Apart from the allowed weight tolerance for the packaged goods, the recipe also includes the individual weight of a wall anchor. This can be determined via a teach function, based on the arithmetic average of a counted sample, and be saved in the recipe.

The packaging weight is tared to 0 before filling. The filling process opens two sliders. One opens/closes the storage container with the goods to be packaged. The second limits the filling speed (fast/slow). When reaching the specified threshold (e.g. 90% of the weight setpoint), the filling speed is lowered by the second slider. This prevents overfilling. When the calculated weight setpoint (pieces x individual weight) is reached, the second slider is also closed and filling is completed. Underfilling is not permissible.

Subsequently, a quality assessment is to take place. The packaged goods will pass the quality control if the filling weight is within the tolerance specified, it not, it will not pass.

In the course of this quality assurance measure, the goods are to be clearly identified and all relevant data, including time stamp is to be logged.

It shall be possible to import the log data to Office Excel. The automation of the logging process and the integration of the required components into the existing infrastructure of the packaging system are to be possible.

It must be possible to apply the check for completeness of the packaged goods also to other products with different content without extra work.

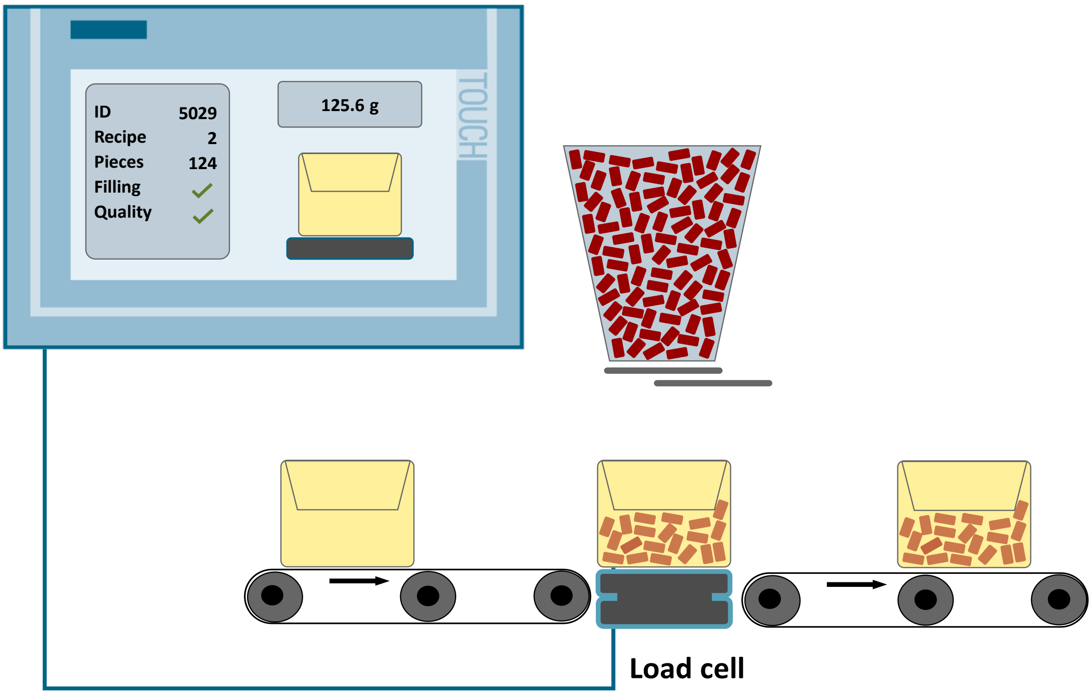
The packaging system is to be operated and maintained exclusively via an HMI device. Operation is to be possible in German and English.

The recipe data is to be edited via remote access through a control station. Accepting the changed data is then again to be done via the operator panel.

### Overview of the automation task

The figure below provides an overview of the automation task.

Figure 1-1



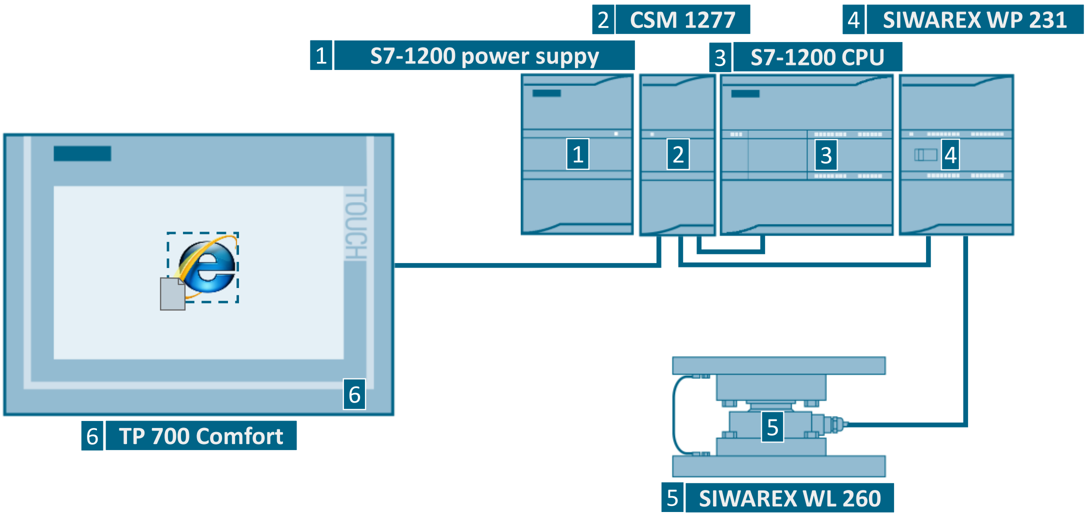
# 2 Solution

## 2.1 Overview

### Schematic layout

The figure below shows a schematic overview of the most important components of the solution:

Figure 2-1



The automation solution uses an **S7-1200 controller** and the SIWAREX MS weighing module WP231 with a WL260 load cell. The weight value of the packaged goods is recorded and compared with a reference value. After filling, it is checked whether all components are complete. The packaged goods can be clearly identified by a batch number.

The DataLog function of the **S7-1200 controller** makes it possible to log the currently measured weight values with a time stamp in the flash memory of the CPU.

You can download your data log files with the help of an internet browser (e.g. the integrated Internet Explorer on the TP 700 Comfort TouchPanel) via the integrated PLC web server of the **S7-1200 controller**.

Connecting the **S7-1200 controller** to a Windows PC enables automatic read out the log data and the evaluation with common spreadsheet applications such as Excel.

The individual stations of the filling process are monitored via screens that can be switched to German or English using a **TP 700 Comfort** touch panel.

The recipes to manage the reference data of different production series are saved in the load memory of the **S7-1200 controller**.

The configuration of the SIWAREX WP231 weighing module is performed via the SIMATIC S7-1200 by means of “write data record” as of firmware V1.1.

### Application areas

Set 6 is suitable for many industrial applications in which cost-effective weight measurements have to be performed with little engineering overhead. This set is particularly suitable if additionally automated logging functions are required in the framework of the measurements.

Set 6, for example, is particularly suitable for the following sectors and fields of application:

- Food industry
- Packaging industry
- Raw materials
- Mechanical engineering

### Advantages

- The integration of the SIWAREX WP231 weighing module into the S7-1200 ensures that the technological functions of the weighing module are combined with all advantages of the PLC world (expandability, flexibility, software, HMI, drives, communication interfaces, etc.)
- Cost-effective high-precision weighing by means of SIWAREX load cell
- Fast and simple configuration via TIA Portal V14
- Automated logging functions provides evidence for the customers
- Easy integration into existing systems due to a connection via PROFINET

### Topics not covered by this application

This application focuses on the filling process by means of weighing technology, the recipe management and the data logging in the load memory of the controller. The filling process is deliberately kept simple and is realized via the digital control of two sliders. When controlling an analog valve, the program code has to be adjusted accordingly.

This application does not include a description of:

- Positioning of the filling container (delivery and forwarding)

### Assumed knowledge

Basic knowledge of SIMATIC S7-1200 and the TIA Portal is assumed.

## 2.2 Hardware and software components

### 2.2.1 Validity

This application is valid for

- STEP 7 V14 (\9\) Update 2 (\11\)
- WinCC Comfort V14 (\10\) Update 2 (\11\)
- CPU 1214C Firmware V4.2.1 (\12\)
- SIWAREX WP231 Firmware V3.1.1 (\5\)

### 2.2.2 Components used

The application was created using the following components:

**Hardware components**

Table 2-1

| Component | Qty. | Article number | Note |
|---|---|---|---|
| PM 1207 power supply | 1 | 6EP1332-1SH71 | Supplies the components with 24V DC |
| CSM 1277 | 1 | 6GK7277-1AA10-0AA0 | Ethernet switch |
| CPU 1214C DC/DC/DC | 1 | 6ES7214-1AG40-0XB0 | S7-1200 controller<br>Firmware: V4.2.1 |
| SIWAREX WP 231 | 1 | 7MH4960-2AA01 | Weighing module<br>Firmware: V3.1.1 |
| SIWAREX WL 260 load cell | 1 | 7MH5102-1KD00 | Rated load: 3 kg |
| SIMATIC HMI TP700 Comfort | 1 | 6AV2124-0GC01-0AX0 | Operator panel |

**Accessories**

Table 2-2

| Component | Qty. | Article number | Note |
|---|---|---|---|
| SIMATIC NET, INDUSTRIAL ETHERNET TP CORD RJ45/RJ45, CAT 6, TP CABLE 4X2, PREPARED WITH 2 RJ45 CONNECTORS, …<br>0.5M<br>1M<br>2M<br>6M<br>10M | 4 | 6XV1870-3Q…<br>…E50<br>…H10<br>…H20<br>…H60<br>…N10 | Ethernet cable |
| Standard mounting rail 35mm | 1 | 6ES5 710-8MA11 | 483 mm |

### Software components

Table 2-3

| Component | Qty. | Article number | Note |
|---|---|---|---|
| STEP 7 Basic V14 | 1 | 6ES7822-0AA04-0YA5 | Configuration and programming of the SIMATIC S7-1200 |
| WinCC Comfort V14 | 1 | 6AV2101-0AA04-0AA5 | Configuration and programming of the TP 700 Comfort |
| SIWATOOL V7<br>Configuration package | 1 | 7MH4960-2AK01 | PC configuration software for the SIWAREX WP 231 weighing module<br>- **optional**: only required for firmware updates. |

### Example files and projects

The following list includes all files and projects that are used in this example.

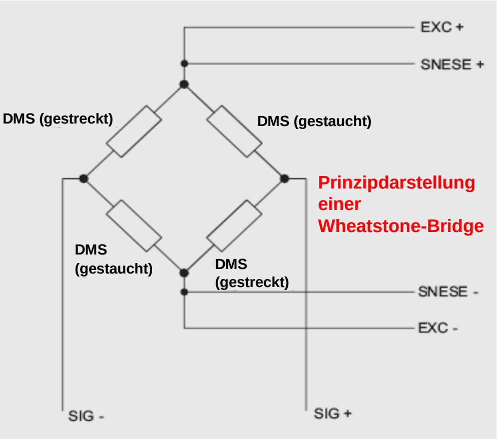
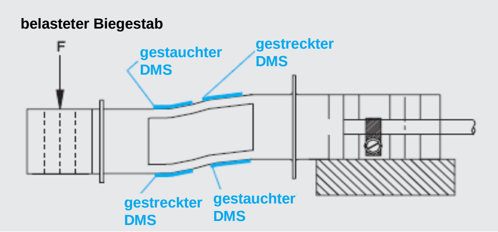
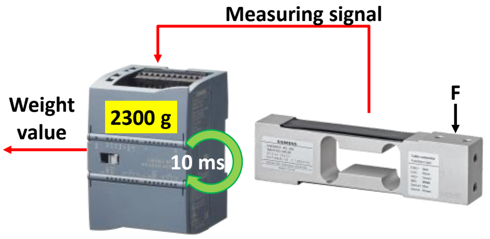
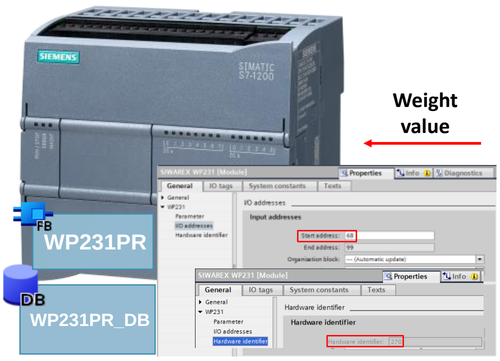
Table 2-4

| Component | Note |
|---|---|
| 82454336_S7-1200_SIWAREX_Set6_PROJ_v2d0.zip | This zip file contains the TIA Portal project. |
| 82454336_S7-1200_SIWAREX_Set6_DOC_v2d0_en.pdf | This document. |

# 3 Basics

## 3.1 Recording weight as measured variable and providing as value

Table 3-1

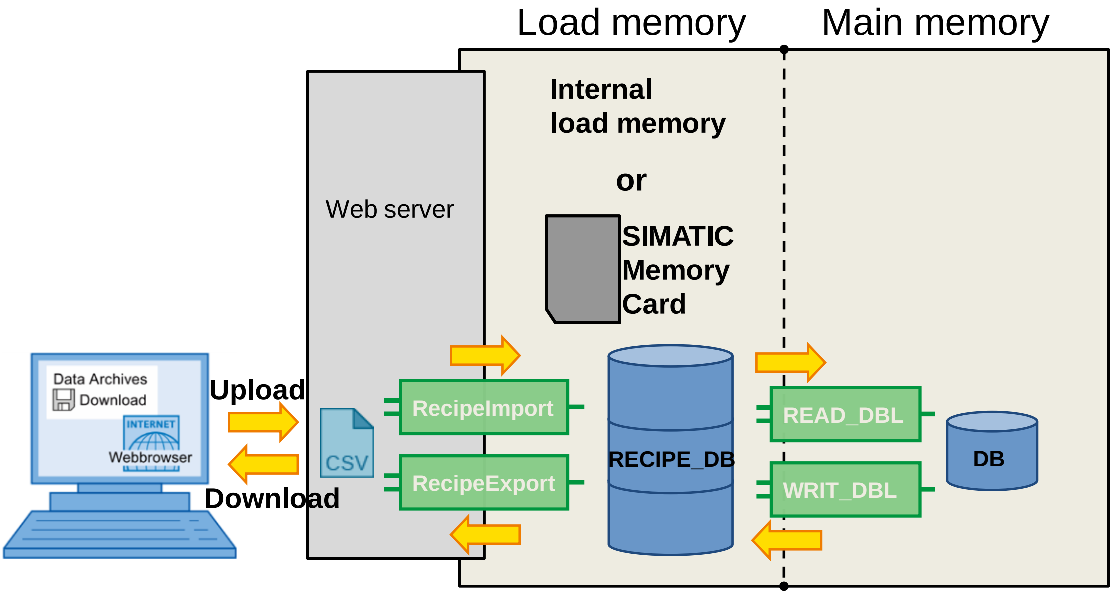
| No. | Function | Comment |
|---|---|---|
| 1. | The SIWAREX WL260 load cell is used to convert a mechanical force into an electrical signal.<br>Four expansion measuring strips (EMS) interconnected to a Wheatstone bridge, are attached to the spring rod of the load cell. |  EXC +<br>SNESE +<br>DMS (gestreckt)<br>DMS (gestaucht)<br>Prinzipdarstellung einer Wheatstone-Bridge<br>DMS (gestaucht)<br>DMS (gestreckt)<br>SNESE -<br>EXC -<br>SIG -<br>SIG + |
| 2. | If a force acts upon the spring rod and compresses or stretches the expansion measuring strips attached to it, an overall misalignment of the spring rod can be determined from the positive and negative changes in resistance. (Measurement voltage, proportional to change in resistance) |  belasteter Biegestab<br>F<br>gestauchter DMS<br>gestreckter DMS<br>gestreckter DMS<br>gestauchter DMS |
| 3. | With the aid of the analog-digital converter integrated in the SIWAREX WP231 weighing module, a weight value is continuously calculated from the measurement voltage. |  |
| 4. | The S7-1200 controller accesses this weight value via the backplane bus in the analog input address area of the WP231 weighing module. The transferred value is a 16-bit integer value.<br>The “WP231PR” function block is used for the conversion into the respective weight value as floating point number.<br>The weight value to be used is stored in the “WP231PR_DB” instance data block. |  SIEMENS<br>SIMATIC S7-1200<br>Weight value<br>FB<br>WP231PR<br>DB<br>WP231PR_DB<br>SIWAREX WP231 [Module]<br>Properties<br>Info<br>Diagnostics<br>General<br>IO tags<br>System constants<br>Texts<br>General<br>WP231<br>Parameter<br>I/O addresses<br>Hardware identifierI/O addresses<br>Input addresses<br>Start address: 68<br>End address: 99<br>Organization block: --- (Automatic update)<br>Hardware identifier<br>Hardware identifier: 270 |

**Note** The real addressing can be read out via the device view of the SIWAREX WP231 weighing module in the “I/O addresses” and “Hardware identifier” menu item in STEP 7 V14 Basic.

## 3.2 Recipe management

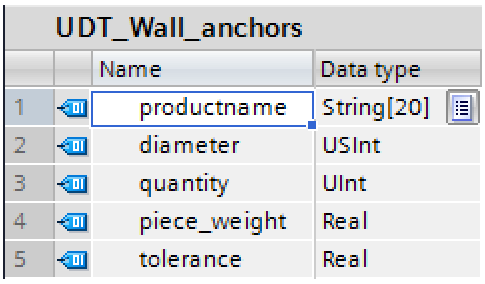
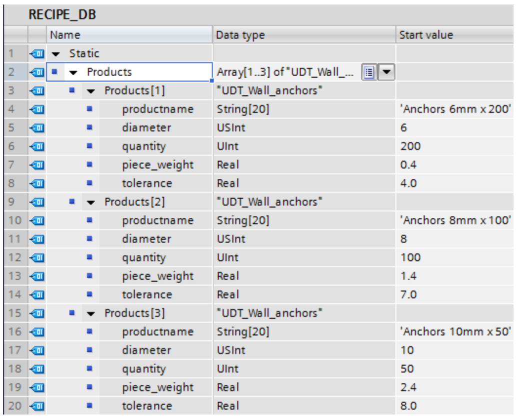
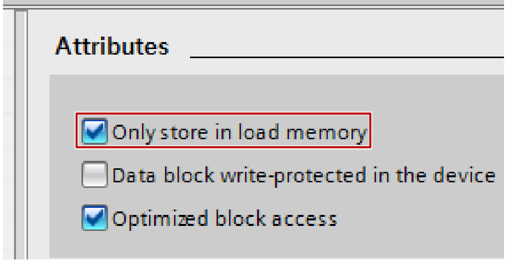
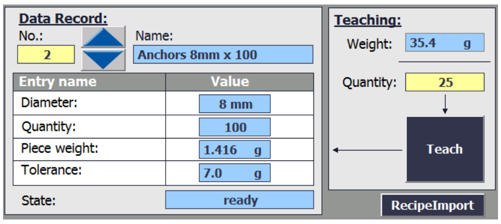
The recipes are saved in the load memory of the controller (internally or on the memory card) to relieve the main memory.
In the main memory, only a data structure of the recipe elements has to be defined as a wild card.
A certain recipe data record is copied from the load memory to the wild card in the main memory via the “READ_DBL” function.
Recipe information of the wild card in the main memory is copied in a certain recipe data record via the “WRIT_DBL” function.

Figure 3-1



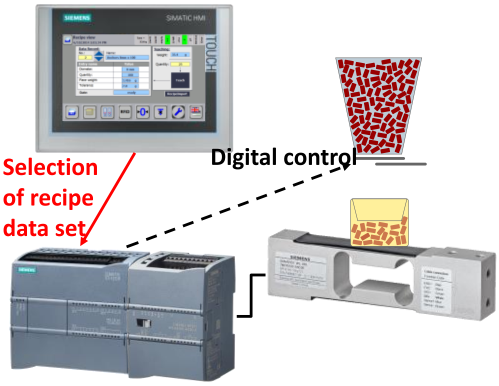
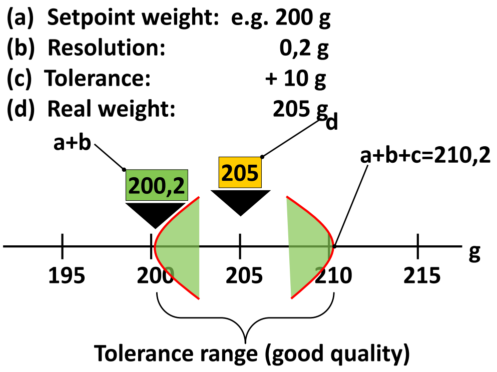
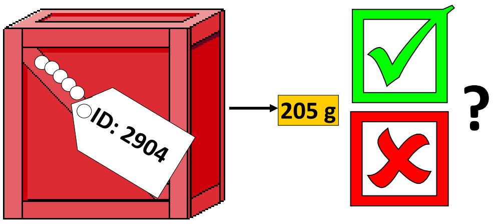
As of CPU firmware V4, the recipe data in the load memory can be copied as csv file onto the integrated web server of the S7-1200 controller via the “RecipeExport” function.
Via a web browser on your PG you can download this csv file from the web server, edit it, save it and upload it again onto the web server (after previously deleting the original file).
The changed recipe data from the csv file on the web server can then be copied into the load memory of the controller via the “RecipeImport” function.

Table 3-2

| No. | Function | Comment |
|---|---|---|
| 1. | Within the framework of the configuration, the packaged goods are selected with the help of the recipe management.<br>In the example project a recipe has been created as PLC data type “UDT_Wall_anchors”.<br>The recipe consists of 5 elements:<br>• Name of the product<br>• Diameter of the wall anchors to be packaged<br>• Individual weight of a wall anchor<br>• Quantity (number of the wall anchor to be packaged)<br>• Tolerance in gram | <br>UDT_Wall_anchors<br>Name – Data type<br>1 productname – String[20]<br>2 diameter – USInt<br>3 quantity – UInt<br>4 piece_weight – Real<br>5 tolerance – Real |
| 2. | 3 recipe data records are predefined in the “RECIPE_DB” recipe data block under the name “Products” as array of “UDT_Wall_anchors” with respective start values.<br>• 200 wall anchors with a diameter of 6mm<br>• 100 wall anchors with a diameter of 8mm<br>• 50 wall anchors with a diameter of 10mm | <br>RECIPE_DB<br>Name – Data type – Start value<br>1 Static<br>2 Products – Array[1..3] of "UDT_Wall_...<br>3 Products[1] – "UDT_Wall_anchors"<br>4 productname – String[20] – 'Anchors 6mm x 200'<br>5 diameter – USInt – 6<br>6 quantity – UInt – 200<br>7 piece_weight – Real – 0.4<br>8 tolerance – Real – 4.0<br>9 Products[2] – "UDT_Wall_anchors"<br>10 productname – String[20] – 'Anchors 8mm x 100'<br>11 diameter – USInt – 8<br>12 quantity – UInt – 100<br>13 piece_weight – Real – 1.4<br>14 tolerance – Real – 7.0<br>15 Products[3] – "UDT_Wall_anchors"<br>16 productname – String[20] – 'Anchors 10mm x 50'<br>17 diameter – USInt – 10<br>18 quantity – UInt – 50<br>19 piece_weight – Real – 2.4<br>20 tolerance – Real – 8.0 |
| 3. | Set the “Only store in load memory” option in the properties of the “RECIPE_DB” data block under “Attributes”. | <br>Attributes<br>☑ Only store in load memory<br>☐ Data block write-protected in the device<br>☑ Optimized block access |
| 4. | The recipe for filling can be selected via the data record no.<br>In addition, there is the option to weigh a counted sample of wall anchors via the “Teach” function. The detected weight is divided by the pieces specified and thus the arithmetic average is calculated and written into the recipe data record as new individual weight of a wall anchor.<br>The weight to be filled can be calculated via the pieces and the individual weight. The tolerance is decisive for the later quality assessment of the packaged product. | <br>Data Record:<br>No.: 2<br>Name: Anchors 8mm x 100<br>Entry name – Value<br>Diameter: 8 mm<br>Quantity: 100<br>Piece weight: 1.416 g<br>Tolerance: 7.0 g<br>State: ready<br>Teaching:<br>Weight: 35.4 g<br>Quantity: 25<br>Teach<br>RecipeImport |

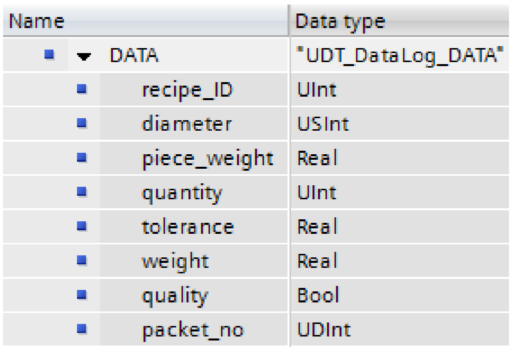
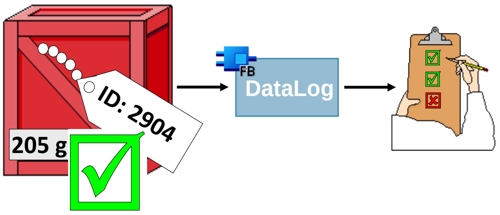
## 3.3 Filling goods

Table 3-3

| No. | Function | Comment |
|---|---|---|
| 1. | Once the product to be packaged is selected via the recipe data record, the setpoint weight is calculated in the CPU and filling is started via the digital control of the two sliders.<br>During the filling process the actual weight is compared with the setpoint weight, with the help of the load cell and the weighing module. When a specified threshold has been reached (e.g. 90% of the setpoint weight), the filling speed is reduced via the slider control. When the setpoint has been reached, the sliders are closed. |  |
| 2. | After filling, the filled weight is evaluated for quality assurance. The weight value for a positive evaluation has be within the following limits:<br>• Setpoint weight + resolution of the load cell<br>• Setpoint weight + resolution of the load cell + tolerance<br>If the real weight value of the product is in the tolerance range of the requirements, the quality of the current product is assessed as good.<br>Underfilling is therefore not possible. |  (a) Setpoint weight: e.g. 200 g<br>(b) Resolution: 0,2 g<br>(c) Tolerance: + 10 g<br>(d) Real weight: 205 g<br>d<br>a+b<br>200,2<br>205<br>a+b+c=210,2<br>g<br>195<br>200<br>205<br>210<br>215<br>Tolerance range (good quality) |
| 3. | Each filled product is counted. The continuous numbering, for example, can be linked with the product via a labeling machine. This is how the product (e.g. via barcode scanner) can be tracked. The product can later also be clearly identified in the log data via this ID. |  |
| 4. | After completing the quality assurance, the result of the quality assurance is assigned to the product ID. | Quality status:<br>• 0 (poor)<br>• 1 (good) |

## 3.4 Logging the quality assurance

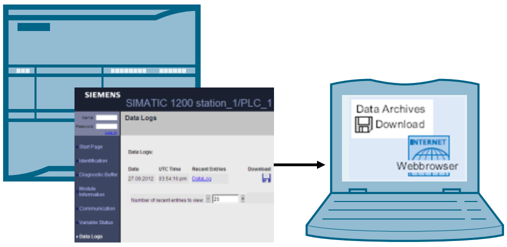
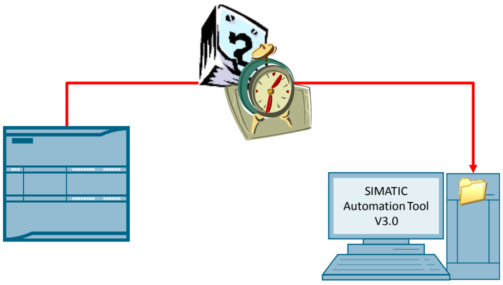
Table 3-4

<table>
<tr><th>No.</th><th>Function</th><th>Comment</th></tr>
<tr><td>1.</td><td>Within the framework of the configuration, logging is executed with the help of the “Data Log” functionality in the flash memory of the S7-1200.<br>Each data log entry includes the following data:<ul><li>Recipe ID</li><li>Diameter of the wall anchors to be packaged</li><li>Individual weight of a wall anchor</li><li>Quantity (number of the wall anchor to be packaged)</li><li>Tolerance in gram</li><li>Total weight measured</li><li>Quality of the filling (good/poor)</li><li>Product ID (packet no.)</li></ul></td><td>
<table>
<tr><th>Name</th><th>Data type</th></tr>
<tr><td>DATA</td><td>"UDT_DataLog_DATA"</td></tr>
<tr><td>recipe_ID</td><td>UInt</td></tr>
<tr><td>diameter</td><td>USInt</td></tr>
<tr><td>piece_weight</td><td>Real</td></tr>
<tr><td>quantity</td><td>UInt</td></tr>
<tr><td>tolerance</td><td>Real</td></tr>
<tr><td>weight</td><td>Real</td></tr>
<tr><td>quality</td><td>Bool</td></tr>
<tr><td>packet_no</td><td>UDInt</td></tr>
</table></td></tr>
<tr><td>2.</td><td>Logging is performed with the help of the “DataLog” function block, once packaging and quality check of the product has been completed.</td><td>
</td></tr>
<tr><td>3.</td><td>When the logging process is started, the FB “DataLog” writes the current values into the flash memory of the CPU 1214C. With every call, a new data record is added to the already existing log data. The size of the log file is specified with 1000 data records before the oldest is overwritten. (Ring buffer)</td><td><ul><li>When creating the log file (first call of FB “DataLog”) its size can be determined (memory capacity of the flash memory is assumed).</li><li>In addition, a date and time stamp is stored for each data record</li></ul></td></tr>
</table>

# 3.5 Automated archiving of the log data

### Archiving the log data

Table 3-5

| No. | Function | Comment |
|---|---|---|
| 1. | The log data can be exported from the flash memory onto the local hard disk of a Window PC with the help of an internet browser via the integrated web server of the S7-1200 and can be saved as CSV file[1].<br>In addition, the **S**IMATIC **A**utomation **T**ool (SAT) as of V3.0, provides access to csv files (saved on a plugged memory card) of S7-1200. |   SIEMENS<br>SIMATIC 1200 station_1/PLC_1<br>Name<br>Password<br>Log in<br>Start Page<br>Identification<br>Diagnostic Buffer<br>Module Information<br>Communication<br>Variable Status<br>Data Logs<br>Data Logs<br>Data Logs:<br>Date<br>UTC Time<br>Recent Entries<br>Download<br>27.09.2012<br>03:54:16 pm<br>DataLog<br>Number of recent entries to view:<br>25<br>Data Archives<br>Download<br>INTERNET<br>Webbrowser |
| 2. | The SIMATIC Automation Tool as of V3.0 additionally provides an API that enables you to, for example, remotely execute SAT functions via Visual Studio.<br>With the help of the "Scheduled Tasks" standard function in MS Windows and the SIMATIC Automation Tool as of V3.0, you can automate the archiving of the log data at user-definable intervals. |  |

**Note** For more information on the SIMATIC Automation Tool V3.0, as well as further links, please refer to the Sales and delivery release (\16\).

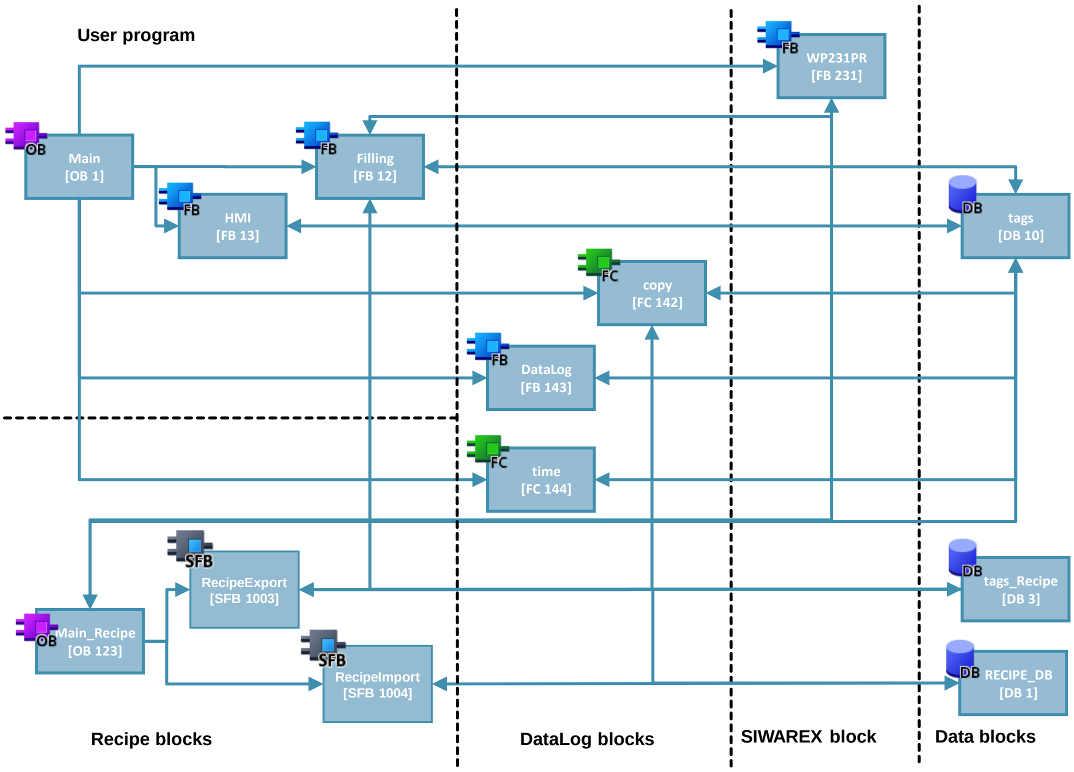
[1] A CSV file is an ASCII file for saving or exchanging simple structured data. The abbreviation CSV stands for *Character Separated Values*, since the individual values are separated by a special separator. There is no general standard for the file format. In the application on hand, the line end is respectively characterized by CR, LF and the individual data is separated by semicolon.

# 4 Mode of Operation

Below, the blocks used are introduced and the most important interface parameters are described.

## 4.1 General overview

Figure 4-1



## 4.2 SIWAREX blocks

For easier use, the WP231 weighing module provides the “WP231PR” (FB 231) function block.

The block is integrated in the example project. However, they are also on the CD of the "SIWAREX WP231 configuration package for SIMATIC S7-1200" (Table 2-3), as well as in the "Ready_for_use_NAWI_WP231" projects (\6\).

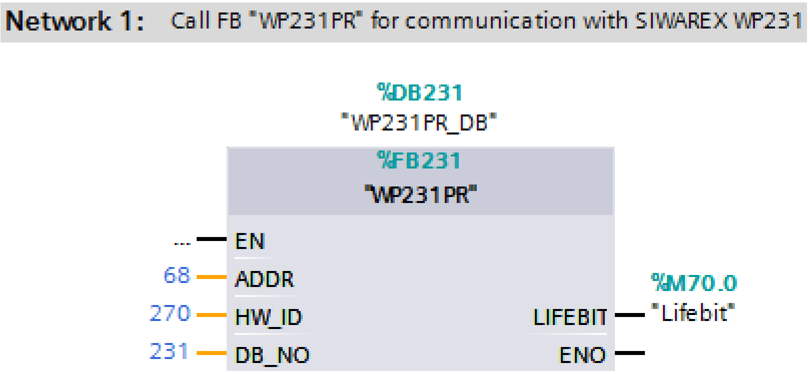
### 4.2.1 "WP231PR" function block

The “WP231PR” FB is used for the communication of the S7-1200 CPU with the SIWAREX WP231 weighing module via the backplane bus.

As of firmware V1.1 of the WP231 weighing module, apart from the communication via the input and output address area, the data records can also read or written in the WP231. Thus, the configuration of the weighing module is also possible without the "SIWAREX WP231 configuration package for SIMATIC S7-1200" (Table 2-3).

The call of the FB8 is in OB1.

Figure 4-2



Network 1: Call FB "WP231PR" for communication with SIWAREX WP231

%DB231
"WP231PR_DB"
%FB231
"WP231PR"

--- EN
68 ADDR
270 HW_ID
231 DB_NO
LIFEBIT %M70.0 "Lifebit"
ENO

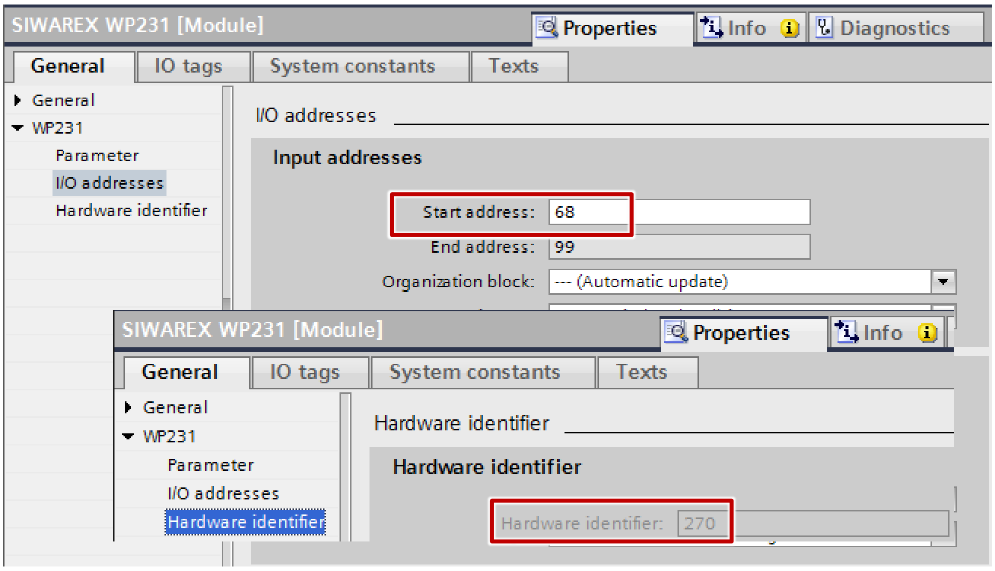
### Interface

Table 4-1

| | Name | Data type | Description |
|---|---|---|---|
| Input | ADDR | DInt | Start address of the I/O area of the WP231 |
| | HW_ID | HW_IO | Hardware identifier of the SIWAREX WP231 |
| | DB_SCALE | Int | Data block number of the instance DB |
| Output | LIFEBIT | Bool | Toggle bit to check communication |

The communication between controller and weighing module requires 32 byte in the input and output area of the S7-1200 CPU.
Parameter ADDR, as well as the hardware identifier has to match the real addressing in the device view of the SIWAREX WP231 weighing module, menu item “I/O addresses” and “Hardware identifier” in STEP 7 V14.

Figure 4-3



I/O addresses

Input addresses

Start address: 68

End address: 99

Organization block: --- (Automatic update)

Hardware identifier

Hardware identifier

Hardware identifier: 270

### 4.2.2 “WP231PR_DB” instance data block

The “WP231PR_DB” instance data block forms the interface between the user program and the "WP231PR” function block.

The following tags of the “WP231PR_DB” are used in the example project:

Table 4-2

| Name | Data type | Offset | Description |
|---|---|---|---|
| s_CMD1.i_CMD_CODE | Int | 464.0 | Command code 3 |
| s_CMD1.bo_CMD_TRIGGER | Bool | 466.0 | Command trigger 3 |
| s_CMD2.i_CMD_CODE | Int | 468.0 | Command code 2 |
| s_CMD2.bo_CMD_TRIGGER | Bool | 470.0 | Command trigger 2 |
| s_IO_DATA.SCALE_STATUS_1 | UInt | 490.0 | Status word 1 |
| • .X8 | Bool | 490.0 | Scale empty |
| • .X9 | Bool | 490.1 | Limit 1 exceeded |
| • .X10 | Bool | 490.2 | Limit 2 exceeded |
| • .X0 | Bool | 491.0 | ¼ numerical increment below minimum |
| • .X1 | Bool | 491.1 | 9 numerical increments exceeded above maximum |
| • .X2 | Bool | 491.2 | Scale tared |
| • .X3 | Bool | 491.3 | Scale manually tared |
| • .X6 | Bool | 491.6 | Scale is at a standstill |
| s_IO_DATA.SCALE_STATUS_2 | UInt | 492.0 | Status word 2 |
| • .X9 | Bool | 492.1 | Service mode enabled |
| • .X15 | Bool | 492.7 | Record |
| s_IO_DATA.PROCESS_VAL_1 | Real | 494.0 | Weight |
| s_IO_DATA.PROCESS_VAL_2 | Real | 498.0 | Tare value |
| DR03 | Struct | 534.0 | Data record 3 (calibration parameters) |
| • WEIGHT_UNIT | String[4] | 556.0 | Weight unit (shows the start value up to the configuration via quick start) |
| • MAX_WEIGHT | Real | 574.0 | Maximum weighing range |
| • CALIB_WEIGHT_0 | Real | 578.0 | Calibration weight 0 |
| • CALIB_WEIGHT_1 | Real | 582.0 | Calibration weight 1 |
| • RESOLUTION_D | Real | 602.0 | Numerical increment d (triggering der load cell) |

The following write commands are used in the example project:

Table 4-3

| Function | Code | |
|---|---|---|
| Load factory setting | s_CMD1.i_CMD_CODE = 12 | s_CMD1.bo_CMD_TRIGGER = 1 |
| Enable service mode | s_CMD2.i_CMD_CODE = 1 | s_CMD2.bo_CMD_TRIGGER = 1 |
| Disable service mode | s_CMD2.i_CMD_CODE = 2 | s_CMD2.bo_CMD_TRIGGER = 1 |
| Confirm empty scale | s_CMD2.i_CMD_CODE = 60 | s_CMD2.bo_CMD_TRIGGER = 1 |
| Confirm calibration weight | s_CMD2.i_CMD_CODE = 61 | s_CMD2.bo_CMD_TRIGGER = 1 |
| Set scale to zero | s_CMD2.i_CMD_CODE = 1001 | s_CMD2.bo_CMD_TRIGGER = 1 |
| Taring | s_CMD2.i_CMD_CODE = 1011 | s_CMD2.bo_CMD_TRIGGER = 1 |

| Function | Code | |
|---|---|---|
| Delete taring | s_CMD2.i_CMD_CODE = 1012 | s_CMD2.bo_CMD_TRIGGER = 1 |
| Read data record 3 | s_CMD2.i_CMD_CODE = 2003 | s_CMD2.bo_CMD_TRIGGER = 1 |
| Write data record 3 | s_CMD2.i_CMD_CODE = 4003 | s_CMD2.bo_CMD_TRIGGER = 1 |

## 4.3 Recipe blocks

The recipe management consists of the following blocks:

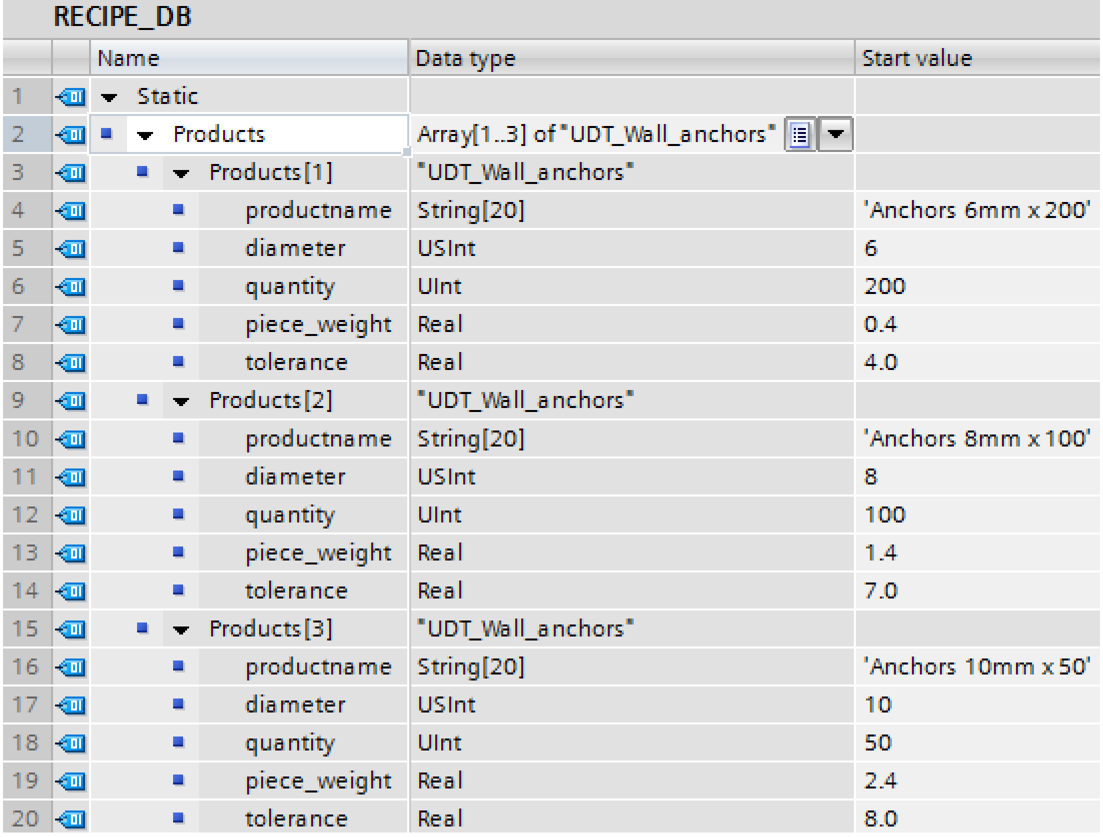
- “RECIPE_DB” data block (DB 1)
- “Main_Recipe” operation block (OB 123)
- “tags_Recipe” data block (DB 3)

### 4.3.1 “Recipe_DB” data block

In “Recipe_DB” the recipes are stored.
This data block is exported in the load memory.
It includes a “Products” array from the “UDT_Wall_anchors” PLC data type.
This array consists of 3 recipes.
The respective recipe values are specified as start values of the elements.

The number of specified recipes can be expanded via the array maximum (3). Before changing the array size, please save the start values by copying them. To be able to operate via the HMI, the “data_record_max” parameter still has to be adjusted in the “tags_Recipe” in the data block (chapter 4.3.3).

Figure 4-4



RECIPE_DB

| | Name | Data type | Start value |
|---|---|---|---|
| 1 | Static | | |
| 2 | Products | Array[1..3] of "UDT_Wall_anchors" | |
| 3 | Products[1] | "UDT_Wall_anchors" | |
| 4 | productname | String[20] | 'Anchors 6mm x 200' |
| 5 | diameter | USInt | 6 |
| 6 | quantity | UInt | 200 |
| 7 | piece_weight | Real | 0.4 |
| 8 | tolerance | Real | 4.0 |
| 9 | Products[2] | "UDT_Wall_anchors" | |
| 10 | productname | String[20] | 'Anchors 8mm x 100' |
| 11 | diameter | USInt | 8 |
| 12 | quantity | UInt | 100 |
| 13 | piece_weight | Real | 1.4 |
| 14 | tolerance | Real | 7.0 |
| 15 | Products[3] | "UDT_Wall_anchors" | |
| 16 | productname | String[20] | 'Anchors 10mm x 50' |
| 17 | diameter | USInt | 10 |
| 18 | quantity | UInt | 50 |
| 19 | piece_weight | Real | 2.4 |
| 20 | tolerance | Real | 8.0 |

**Note** For the proper functioning in connection with the SFBs “RecipeImport” and “RecipeExport”, the recipe DB has to consist of a one-dimensional array of a PLC data type or of a struct and it has to have the “Only store in load memory” attribute in the block properties.

### "UDT_Wall_anchors" PLC data type

The "UDT_Wall_anchors" PLC data type includes the elements of a recipe data record.

Table 4-4

| Name | Data type | Description |
|---|---|---|
| productname | String[20] | Name of the recipe data record |
| diameter | USInt | Wall anchor diameter in mm |
| quantity | UInt | Pieces |
| piece_weight | Real | Individual weight (unit depends on the “Quick Start” configuration) |
| tolerance | Real | Tolerance (unit depends on the “Quick Start” configuration) |

The elements in the recipe data record can be individually adjusted in the PLC data type.

## 4.3.2 “Main_Recipe” organization block

The system instructions and functions are called from the “Main_Recipe” OB:

- READ_DBL (recipe data from the relocated recipe DB in the load memory to the main memory)
- WRIT_DBL (write recipe data from the main memory into the relocated recipe DB in the load memory)
- SFB 1003: RecipeExport (exporting relocated recipe data as csv file on the web server)
- SFB 1004: RecipeImport (importing recipe data from uploaded csv file on the web server into the exported recipe DB in the load memory)

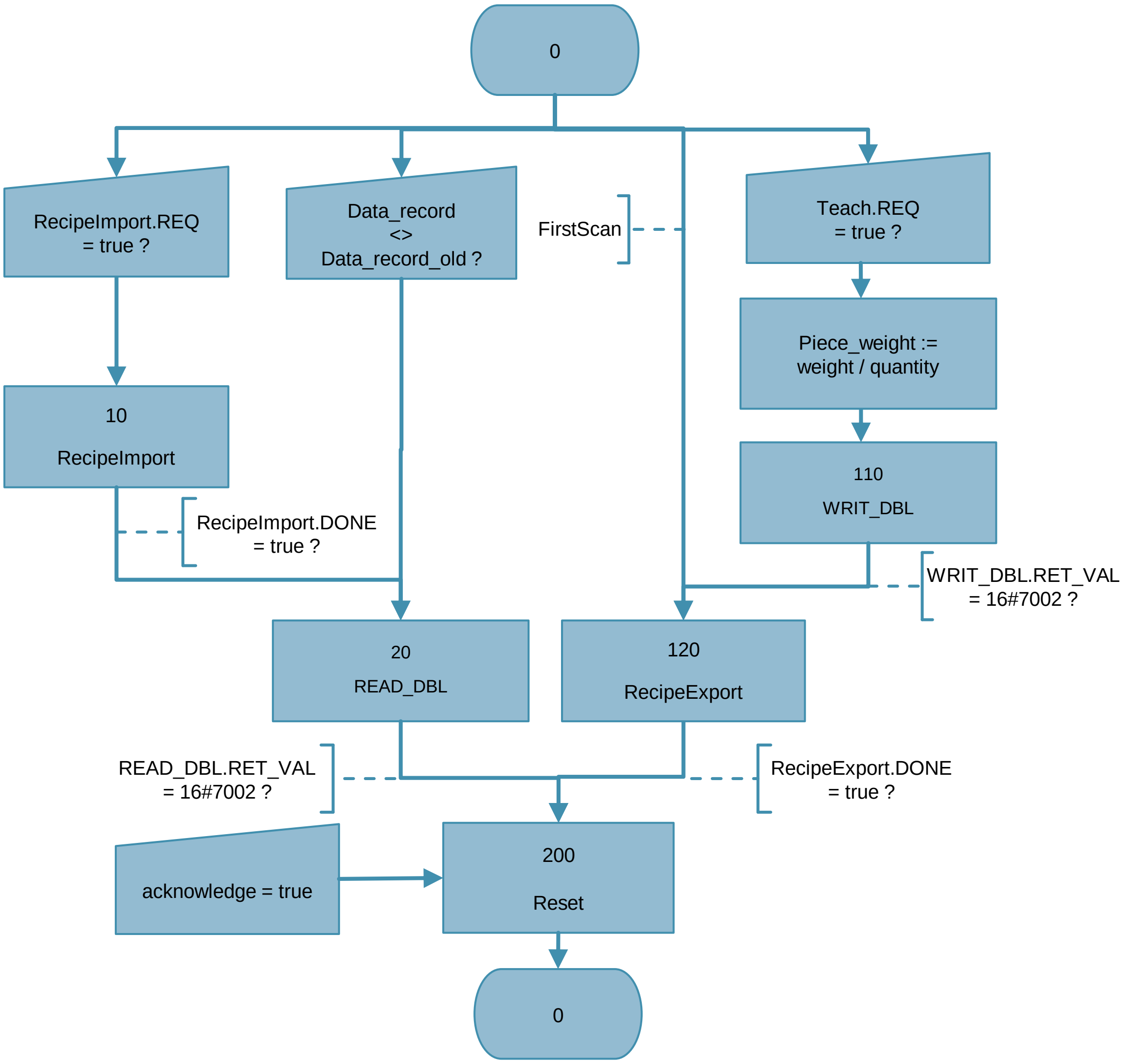
In addition, the teach function to determine the individual weight of a wall anchor is also executed in the “Main_Recipe” OB.
For this purpose, the total weight of the counted sample is divided by its number. The thus calculated arithmetic average is saved in the selected recipe data record as individual weight.

**Note** Further information about the system instructions and functions can be found in the chapter 3.2, as well as in the S7-1200 Programmable controller System Manual (\3\).

### Program flow chart

The “Main_Recipe” OB is programmed as a sequence. The program flow chart is as follows.

Figure 4-5


0
RecipeImport.REQ = true ?
Data_record <> Data_record_old ?
FirstScan
Teach.REQ = true ?
Piece_weight := weight / quantity
110
WRIT_DBL
10
RecipeImport
RecipeImport.DONE = true ?
WRIT_DBL.RET_VAL = 16#7002 ?
20
READ_DBL
120
RecipeExport
READ_DBL.RET_VAL = 16#7002 ?
RecipeExport.DONE = true ?
acknowledge = true
200
Reset
0

### 4.3.3 “tags_Recipe” data block

The “tags_Recipe” data block includes the tags for transmitting the function blocks and instructions in “Main_Recipe” OB to the interfaces.

The following table lists the most important tags for individual adjustment and to understand the program flow chart:

Table 4-5

| Name | Data type | Start value | Description |
|---|---|---|---|
| data_record_max | UInt | 3 | Number of recipe data records (limit of HMI selection) |
| data_record | UInt | 1 | selected recipe data record |
| data_record_old | UInt | 0 | selected recipe data record in last cycle |
| step | USInt | 0 | Step sequence |
| acknowledge | Bool | false | Error acknowledgment (sets the sequence back) |
| current_repice | "UDT_Wall_anchors" | | Current recipe data record in main memory (see Table 4-4) |
| Teach.REQ | Bool | false | Request for individual weight determination |
| Teach.quantity | UInt | 1 | Number of sample (teach mode) |

## 4.4 User program blocks

The user program consists of the following blocks:

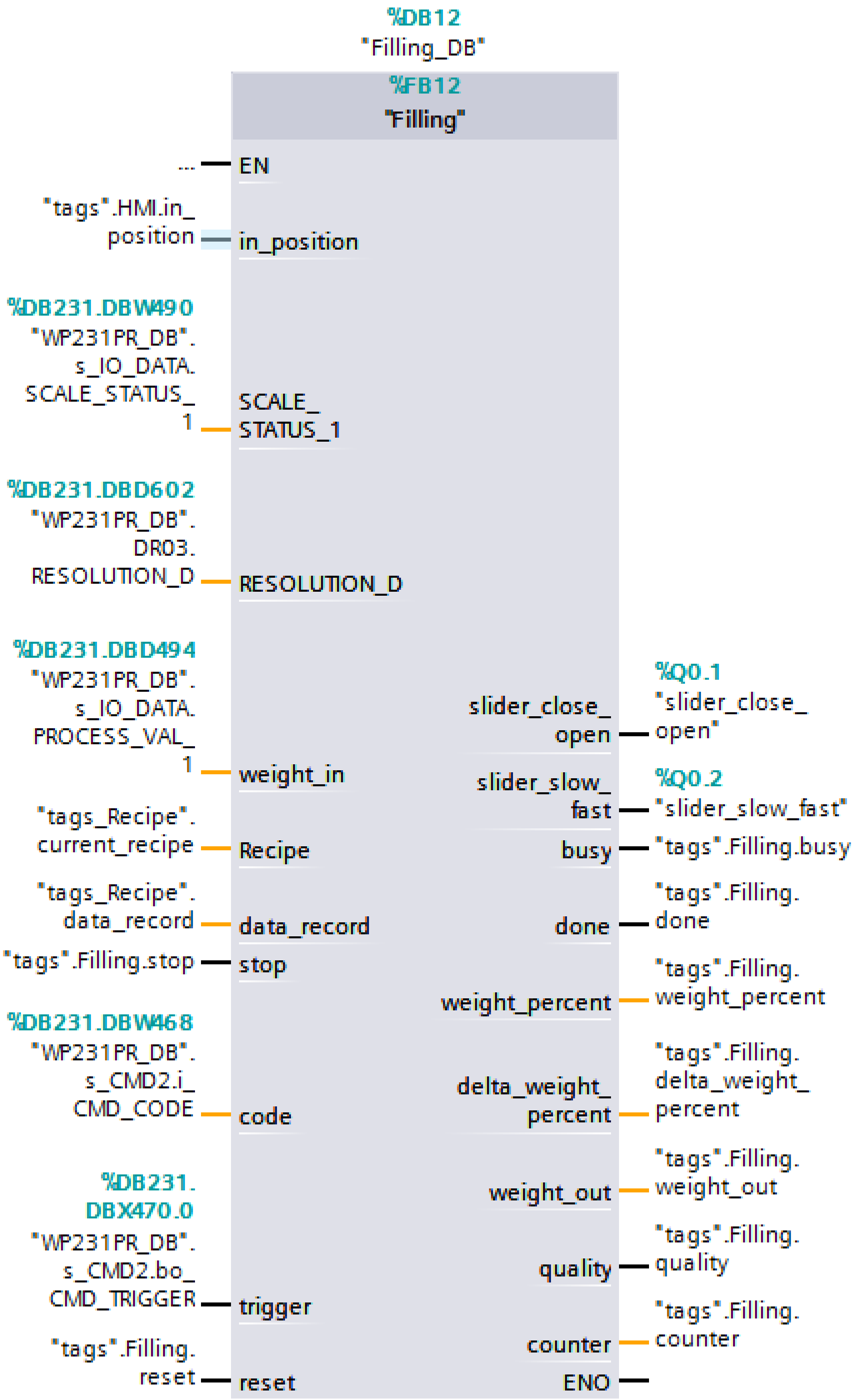
- “Filling” function block (FB 12)
- “HMI” function block (FB 13)
- “tags” data block (DB 10)

### 4.4.1 “Filling” function block

The “Filling” FB is used for filling a container with a selected wall anchor type. For this purpose, the scale is tared to neutralize the weight of the container. The recipe data record of the wall anchor type to be packaged is loaded. The product, made up of pieces and individual weight of a wall anchor, results in the weight setpoint. The “Filling” FB starts the filling process by opening the two sliders. When the specified weight limit is reached, one of the sliders closes, and packaging is slowed down. When the setpoint is reached, the second slider also closes. The weight reached is compared with the tolerance data from the recipe data record and evaluated. The packaged products are continuously counted and therefore receive a clear assignment.

The call of the "Filling" FB is in OB1.

Figure 4-6


%DB12
"Filling_DB"
%FB12
"Filling"
...
EN
"tags".HMI.in_position
in_position
%DB231.DBW490
"WP231PR_DB".s_IO_DATA.SCALE_STATUS_1
SCALE_STATUS_1
%DB231.DBD602
"WP231PR_DB".DR03.RESOLUTION_D
RESOLUTION_D
%DB231.DBD494
"WP231PR_DB".s_IO_DATA.PROCESS_VAL_1
weight_in
"tags_Recipe".current_recipe
Recipe
"tags_Recipe".data_record
data_record
"tags".Filling.stop
stop
%DB231.DBW468
"WP231PR_DB".s_CMD2.i_CMD_CODE
code
%DB231.DBX470.0
"WP231PR_DB".s_CMD2.bo_CMD_TRIGGER
trigger
"tags".Filling.reset
reset
slider_close_open
%Q0.1
"slider_close_open"
slider_slow_fast
%Q0.2
"slider_slow_fast"
busy
"tags".Filling.busy
done
"tags".Filling.done
weight_percent
"tags".Filling.weight_percent
delta_weight_percent
"tags".Filling.delta_weight_percent
weight_out
"tags".Filling.weight_out
quality
"tags".Filling.quality
counter
"tags".Filling.counter
ENO

### Interface

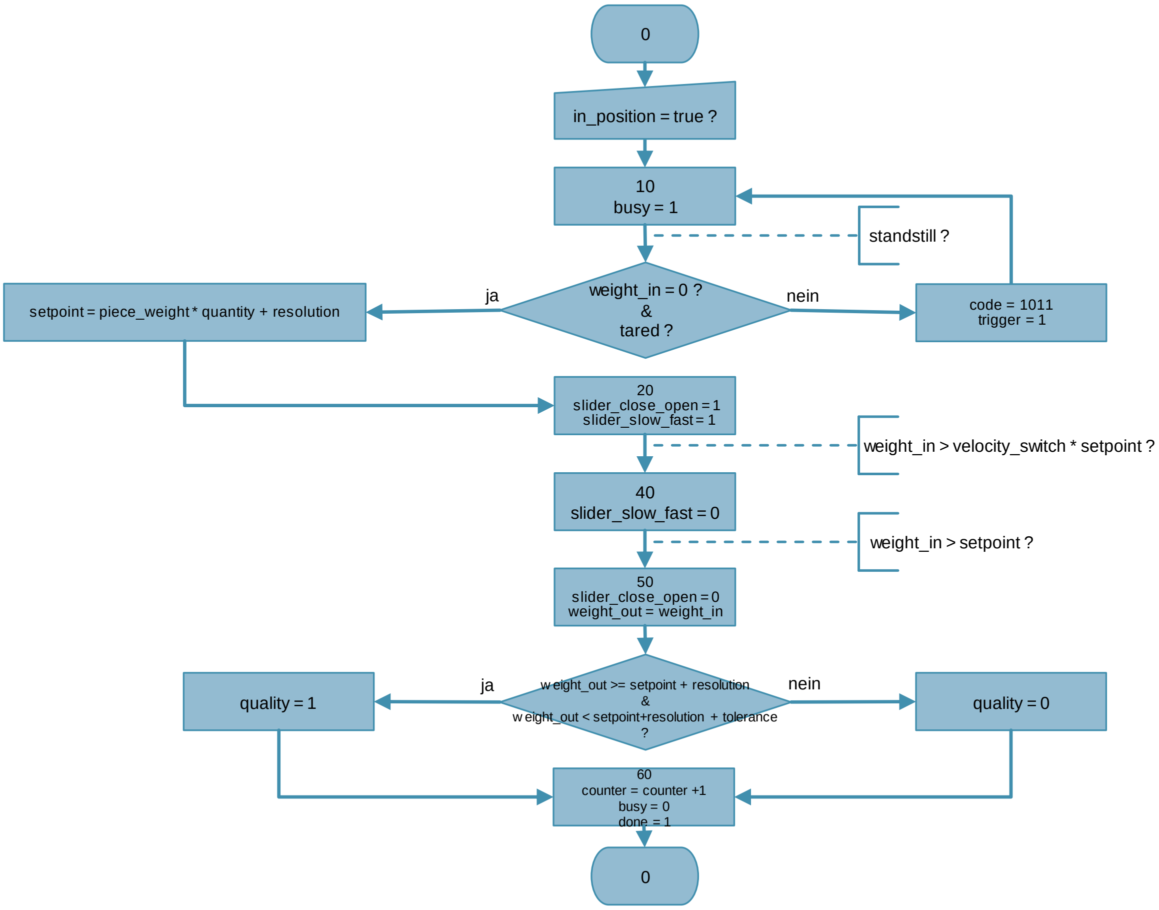
Table 4-6

| | Name | Data type | Description |
|---|---|---|---|
| Input | in_position | Bool | Box is in filling position (start request) |
| | SCALE_STATUS_1 | UInt | Status word 1 of the weighing module (includes taring and standstill of the scale) |
| | RESOLUTION_D | Real | Resolution of the load cell = Numerical increment d (unit depends on "Quick Start" configuration) |
| | weight_in | Real | Weight transfer from WP231PR_DB |
| | Recipe | "UDT_Wall_anchors" | Recipe data (see Table 4-4) |
| | data_record_no | UInt | Recipe data record to be produced |
| Output | slider_close_open | Bool | Slider for opening the storage container |
| | slider_slow_fast | Bool | Slider for accelerating the filling |
| | busy | Bool | Block being processed |
| | done | Bool | Processing completed (one cycle active) |
| | weight_percent | Real | Filling weight in % of setpoint |
| | delta_weight_percent | Real | Filling weight deviation in % of setpoint |
| | weight_out | Real | Filling weight (unit depends on the “Quick Start” configuration) |
| | quality | Bool | Quality assessment (true = okay) |
| | counter | UDInt | Product counter |
| InOut | stop | Bool | Stop (false=operation, true=stop) |
| | code | Int | Pointer on command code 2 (WP231PR_DB) |
| | trigger | Bool | Pointer on command code 2 (WP231PR_DB) |
| | reset | Bool | Reset input (reset after processing) |
| Static | velocity_switch | Real | Value for switching the filling speed (standardized share of setpoint) |

### Program flow chart

The “Filling" FB is programmed as a sequence. The program flow chart is as follows.

Figure 4-7



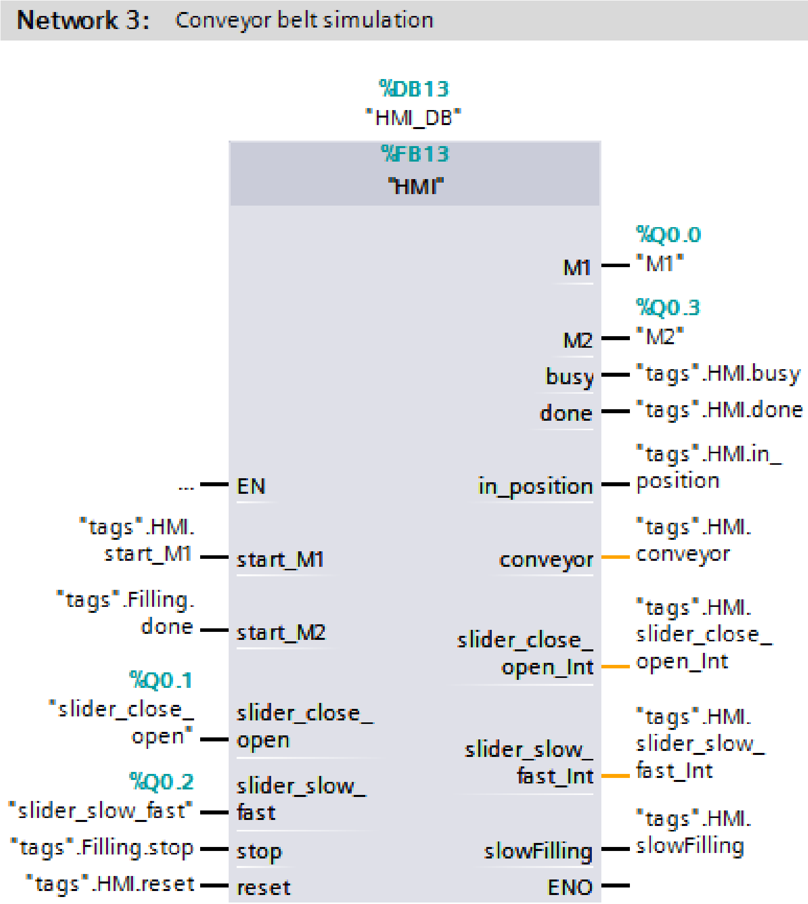
## 4.4.2 “HMI” function block

The “HMI” FB is used to display the transport of the box and the slider on the operator panel. The transport path of the conveyor belt is simulated via the 10Hz clock memory bit “Clock_10Hz” and increments the position of the box by “conveyor_delta”.
The user starts the conveyor belt simulation via “start_M1” on the operator panel. When the setpoint is reached on the scale, “stop_1” stops the “conveyor” position value and the “in_position” output is set for the start request for “Filling” FB.
Once the “Filling” FB has finished, it triggers the “start_M2” request for the removal of the filled box. For the position of the box, the “Clock_10Hz” 10Hz clock memory bit is again incremented by “conveyor_delta”.
When the “stop_2” end position is reached, the conveyor belt stops the conveyor belt simulation and the box jumps back to the start position.

In addition, the display of the slider positions is simulated in this block.

The call of the "HMI" FB is in OB1.

Figure 4-8


Network 3: Conveyor belt simulation
%DB13
"HMI_DB"
%FB13
"HMI"
... EN
"tags".HMI.start_M1 start_M1
"tags".Filling.done start_M2
%Q0.1 "slider_close_open" slider_close_open
%Q0.2 "slider_slow_fast" slider_slow_fast
"tags".Filling.stop stop
"tags".HMI.reset reset
M1 %Q0.0 "M1"
M2 %Q0.3 "M2"
busy "tags".HMI.busy
done "tags".HMI.done
in_position "tags".HMI.in_position
conveyor "tags".HMI.conveyor
slider_close_open_Int "tags".HMI.slider_close_open_Int
slider_slow_fast_Int "tags".HMI.slider_slow_fast_Int
slowFilling "tags".HMI.slowFilling
ENO

**Interface**

Table 4-7

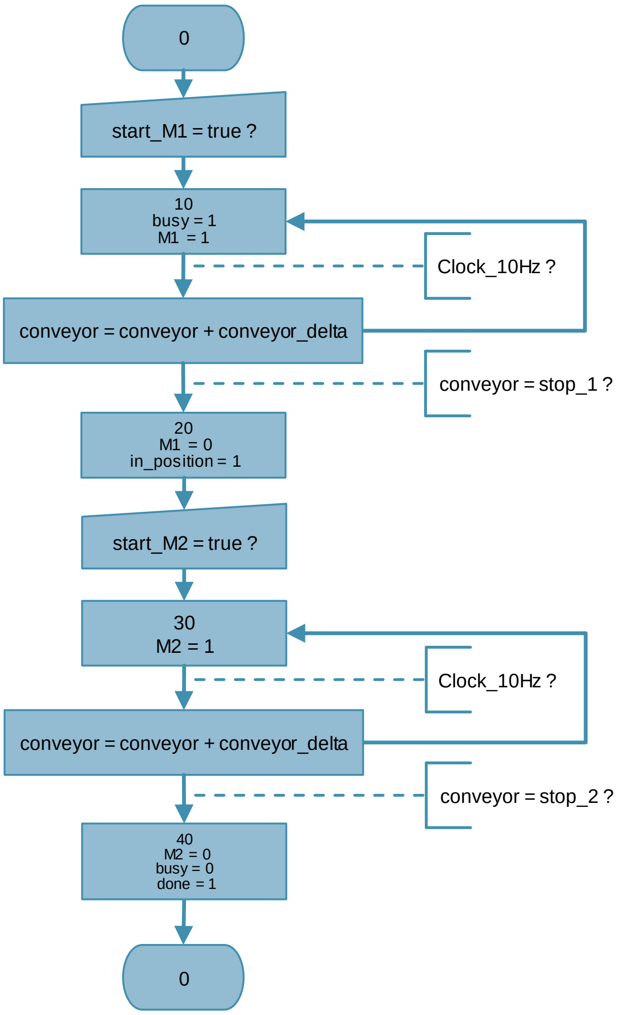
| | Name | Data type | Description |
|---|---|---|---|
| Input | start_M1 | Bool | Start request for the supply conveyor belt |
| | start_M2 | Bool | Start request for the removal conveyor belt |
| | slider_close_open | Bool | Slider for opening the storage container |
| | slider_slow_fast | Bool | Slider for accelerating the filling |
| Output | M1 | Bool | Control of the supply conveyor belt |
| | M2 | Bool | Control of the removal conveyor belt |
| | busy | Bool | Block being processed |
| | done | Bool | Processing completed (one cycle active) |
| | in_position | Bool | Box is in filling position (start request for "Filling" FB) |
| | conveyor | Int | Simulated position of the box |
| | slider_close_open_Int | Int | Simulated position of the slider for opening the storage container |
| | slider_slow_fast_Int | Int | Simulated position of the slider to accelerate the filling |
| | slowFilling | Bool | Slow filling active ("slider_close_open" =TRUE & "slider_slow_fast"=FALSE) |
| InOut | stop | Bool | Stop (false=operation, true=stop) |

| | Name | Data type | Description |
|---|---|---|---|
| | reset | Bool | Reset input (reset after processing) |
| Static | conveyor_delta | Int | Horizontal distance that the box travels in 100ms (in pixel) |
| | stop_1 | Int | Horizontal position of the box on the scale (in pixel) |
| | stop_2 | Int | Horizontal position of the box at the end of the removal conveyor belt (in pixel) |
| | slider_close_open_pixel | Int | Pixel distance between the end positions of the slider for opening the storage container |
| | slider_slow_fast_pixel | Int | Pixel distance between the end positions of the slider for accelerating the filling |

### Program flow chart

The “HMI" FB is programmed as a sequence. The program flow chart is as follows.

Figure 4-9



### 4.4.3 “Tags” data block

The “tags” data block includes the tags for transmitting the function blocks and functions to the interfaces.

The following table shows the tags that have been provided with deviating start values in the example project.

Table 4-8

| Name | Data type | Start value | Description |
|---|---|---|---|
| DataLog.Enable | Bool | true | Enable of “DataLog” FB (false =blocked, true =enabled) |
| DataLog.RECORDS | UDInt | 1000 | Number of data records in the data log |
| DataLog.NAME | String | 'DataLog' | Name of the data log file |

## 4.5 Data logging - blocks

To record the production data, the following blocks are used:

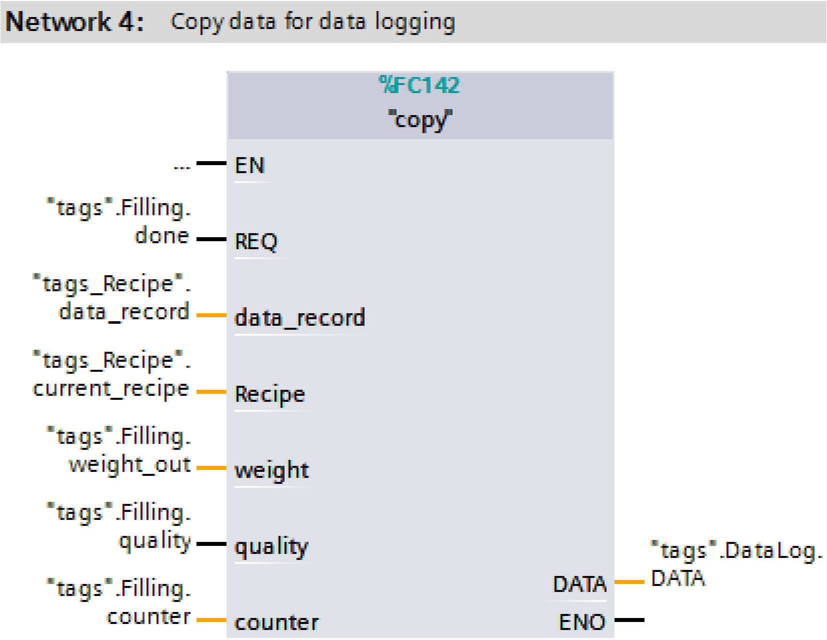
- “Copy” function
- “DataLog” function block
- “Time” function

### 4.5.1 “Copy” function

The “copy” FC only collects all required data that is to be written into the DataLog file and transfers it to the “DataLog” FB.

The call of the “copy" FC is in OB1.

Figure 4-10


Network 4: Copy data for data logging
%FC142
"copy"
... EN
"tags".Filling.done REQ
"tags_Recipe".data_record data_record
"tags_Recipe".current_recipe Recipe
"tags".Filling.weight_out weight
"tags".Filling.quality quality
"tags".Filling.counter counter
DATA "tags".DataLog.DATA
ENO

### Interface

Table 4-9

| | Name | Data type | Description |
|---|---|---|---|
| Input | REQ | Bool | copy request (enabled when REQ = true) |
| | data_record | UInt | Recipe data record number |
| | Recipe | "UDT_Wall_anchors" | Recipe data (see Table 4-4) |
| | weight | Real | Filled weight |
| | quality | Bool | Quality of filling |
| | counter | UDInt | Count value of filling |
| Output | DATA | "UDT_DataLog_DATA" | Summary of all DataLog data |

### "UDT_DataLog_DATA" PLC data type

The "UDT_DataLog_DATA" PLC data type includes all process data that is to be written into the data log data record.

Table 4-10

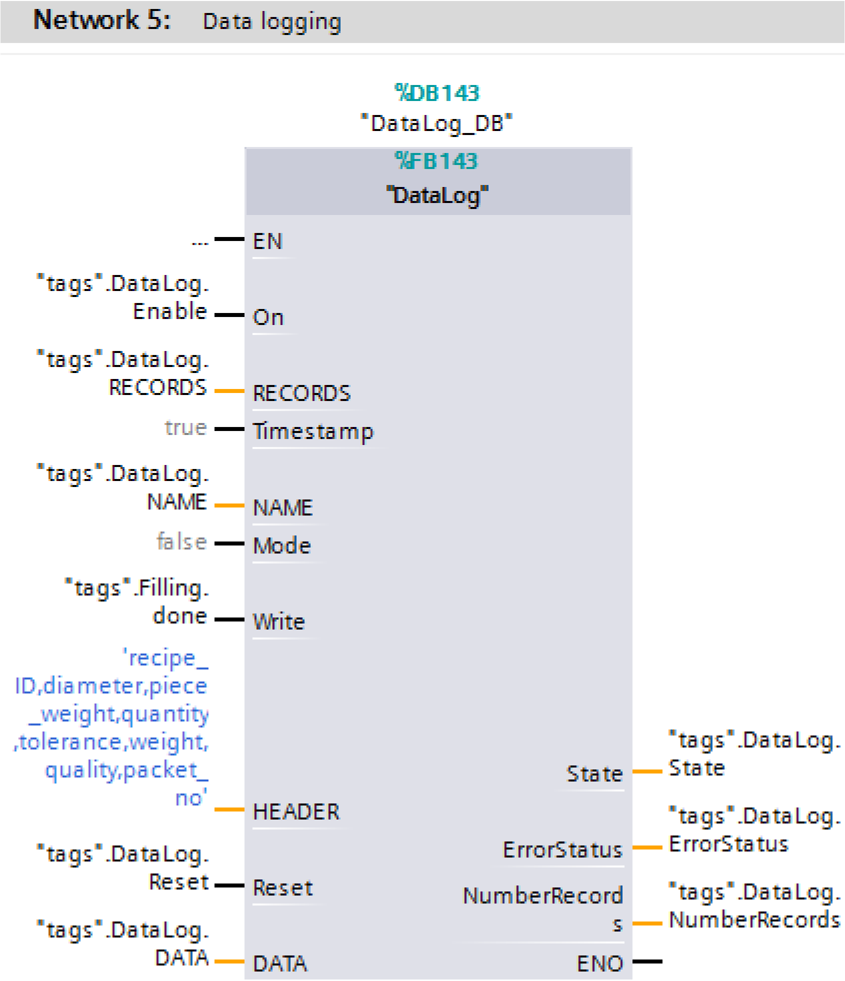
| Name | Data type | Description |
|---|---|---|
| recipe_ID | UInt | Recipe ID (identical with the recipe data record number) |
| diameter | USInt | Wall anchor diameter in mm |
| piece_weight | Real | Individual weight (unit depends on the “Quick Start” configuration) |
| quantity | UInt | Pieces |
| tolerance | Real | Tolerance (unit depends on the “Quick Start” configuration) |
| weight | Real | Filled weight |
| quality | Bool | Quality of filling |
| packet_no | UDInt | Packet number (identical with count value of the filling) |

### 4.5.2 “DataLog” function block

The “DataLog" FB creates, opens and describes a DataLog file according to specifications. When exceeding the maximum entry numbers, the oldest entries are overwritten (ring buffer).

The call of the "Data log" FB is in OB1.

Figure 4-11



**Interface**

Table 4-11

| | Name | Data type | Description |
|---|---|---|---|
| Input | On | Bool | Sequencer enabling (true = processing of the sequence, false = closing of the DataLog files via the "DataLogClose" function) |
| | RECORDS | UDInt | Number of data records in the data log |
| | Timestamp | Bool | Time stamp:<br>• 0: No time stamp<br>• 1: Date and time |

| | Name | Data type | Description |
|---|---|---|---|
| | NAME | String | Name of the data log |
| | Mode | Bool | Mode for opening the data log:<br>• MODE= "0"<br>Data records of the data log remain<br>• MODE= "1"<br>Data records of the data log are deleted |
| | Write | Bool | Execution of the “DataLogWrite” instruction in the event of a rising edge |
| | HEADER | String | Header of the CSV file |
| | Reset | Bool | Reset input |
| Output | State | USInt | Status of the function (identical with the step) |
| | ErrorStatus | Word | Status parameter if an error occurs (in combination with “State”, the corresponding SFB data log can be identified) |
| | NumberRecords | UDInt | Current number of written data records |
| InOut | DATA | Variant | Pointer to the structure or array of the data to be written |

**Program flow chart**

The “DataLog" FB is programmed as a sequence.

When the sequence is enabled (“On”), the sequence starts in step 0.
A DataLog file is created in csv format in the load memory of the CPU with the specifications “NAME”, “RECORDS”, “Timestamp”, “HEADER” and “DATA”.
The file with the “MODE” mode is opened in step 1 once the positive “DONE” feedback appears or the message that the file already exists. If a message appears that the file does not exist, the block returns to step 0. If the feedback is positive, “DONE”, the file is written in step 20, when the “Write” input is enabled. If the error messages “Data log does not exist” or “Data log not open” appear, it is jumped to the respective step in order to remove this error. If there is a positive “DONE” feedback, the number of the written data records (“NumberRecords”) is incremented and the next write request (“Write”) is waited for.

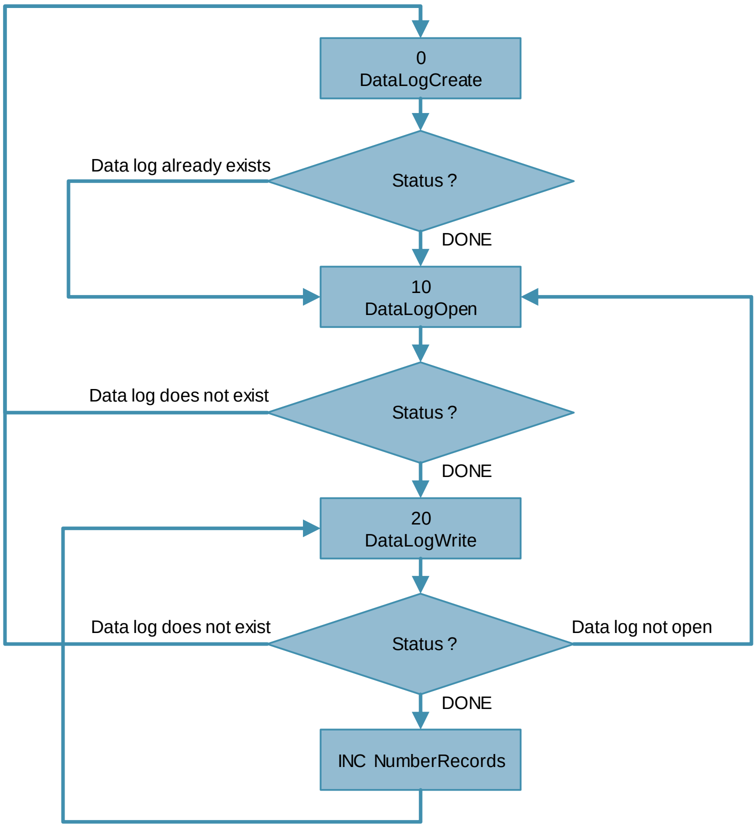
If another error than stated above occurs, the sequence will remain in the current step. From “State” and “ErrorStatus” the appropriate SFBs can be determined from the error description.

The sequence and the counter of the written data records are reset via the “Reset” input.

When the sequencer is disabled ("On" = "false") the DataLog file is closed via the "DataLogClose" function. In this state the appropriate csv file on the web server can be deleted.

The program flow chart is as follows.

Figure 4-12

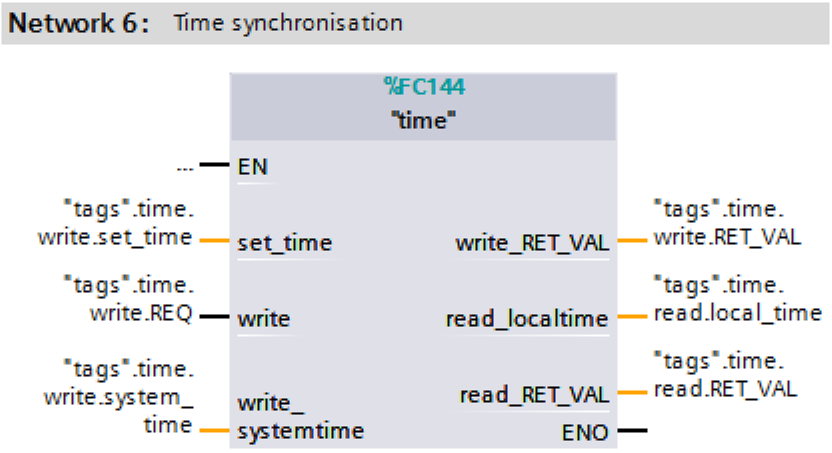


### 4.5.3 “Time” function

The “HMI” FB is used for setting the CPU system and reading the CPU local time. The system time of the controller provides the time stamp of the DataLog data records.
The CPU local time is also displayed in the HMI.
The HMI system time is synchronized with the CPU system time.

The call of the “time" FC is in OB1.

Figure 4-13



Network 6: Time synchronisation

%FC144
"time"

--- — EN
"tags".time.write.set_time — set_time
"tags".time.write.REQ — write
"tags".time.write.system_time — write_systemtime
write_RET_VAL — "tags".time.write.RET_VAL
read_localtime — "tags".time.read.local_time
read_RET_VAL — "tags".time.read.RET_VAL
ENO —

### Interface

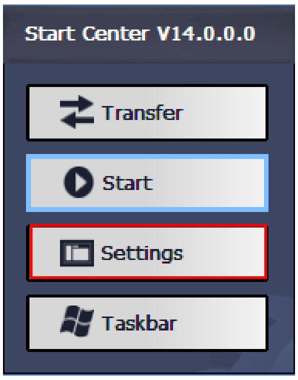
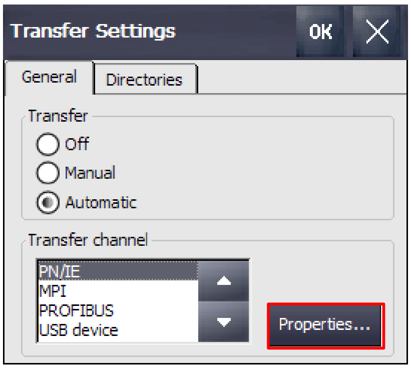
Table 4-12

| | Name | Data type | Description |
|---|---|---|---|
| Input | set_time | Struct | System time for writing (input field in HMI) |
| | • year | UInt | Year |
| | • month | USInt | Month |
| | • day | USInt | Day |
| | • hour | USInt | Hour |
| | • minute | USInt | Minute |
| Output | write_RET_VAL | Int | Status of the "WR_SYS_T" instruction |
| | read_localtime | DTL | Local time (is continuously read) |
| | read_RET_VAL | Int | Status of the "RD_LOC_T" instruction |
| InOut | write | Bool | Request to write system time (reset after processing) |
| | write_systemtime | DTL | System time for writing in DTL format |

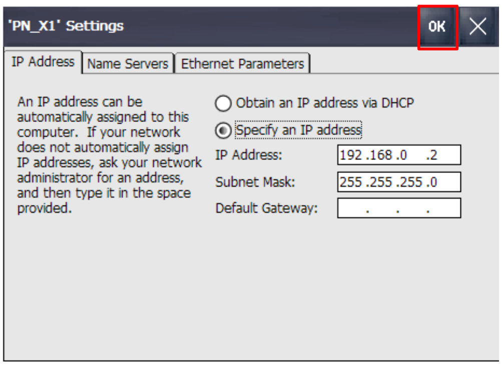
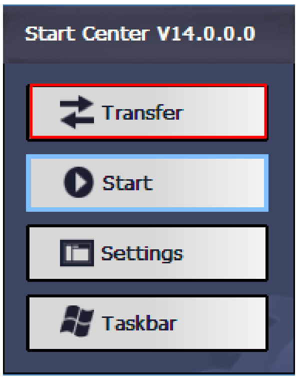
# 5 Configuration and Settings

## 5.1 Configuring SIMATIC Panel TP700 Comfort

Table 5-1

| No. | Action | Comment |
|---|---|---|
| 1. | Provide the TP700 Comfort with 24V. | |
| 2. | Click the “Settings” button in the start center after the initialization phase of Windows CE[2]. | <br>Start Center V14.0.0.0<br> |
| 3. | Double-click the transfer icon in the “Control Panel” settings. | Transfer |
| 4. | Select the “PN/IE” transmission channel in “Transfer Settings”.<br>Then click the “Properties…” button. | <br>Transfer Settings<br>Transfer: ○ Off ○ Manual ● Automatic<br>Transfer channel: PN/IE<br> OK<br>General<br>Directories<br>MPI<br>PROFIBUS<br>USB device<br>Properties... |
| 5. | You will then get to “Network and Dial-up Connections”.<br>Double click the PN_X1 icon. | PN_X1 |

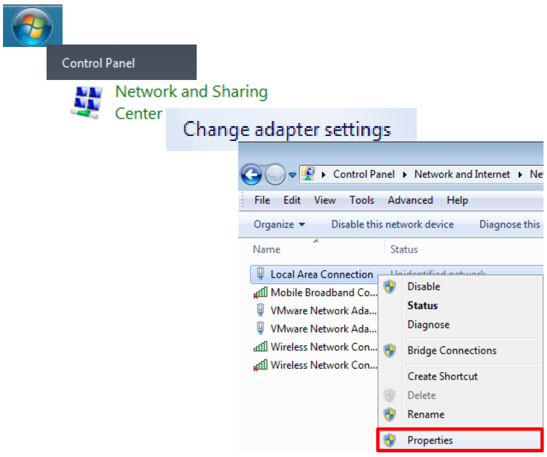
[2] The "bootloader" initialization phase is followed by a startup delay interval after which an already loaded application starts. You therefore have to click an action in the loader within the startup delay time.

| No. | Action | Comment |
|---|---|---|
| 6. | Enable the "Specify an IP address" checkbox and accept the displayed settings:<br>• IP-Address: 192.168.0.2<br>• Subnet Mask: 255.255.255.0<br>Finally, accept the settings made in the "PN_X1" and transfer settings with "OK". | <br>'PN_X1' Settings<br>IP Address \| Name Servers \| Ethernet Parameters<br>An IP address can be automatically assigned to this computer. If your network does not automatically assign IP addresses, ask your network administrator for an address, and then type it in the space provided.<br>○ Obtain an IP address via DHCP<br>● Specify an IP address<br>IP Address: 192 .168 .0 .2<br>Subnet Mask: 255 .255 .255 .0<br>Default Gateway: . . .<br> |
| 7. | The transfer mode in the start center has to be enabled to subsequently transfer the HMI project part into the Comfort Panel. | <br> Start Center V14.0.0.0<br>Transfer<br>Start<br>Settings<br>Taskbar |

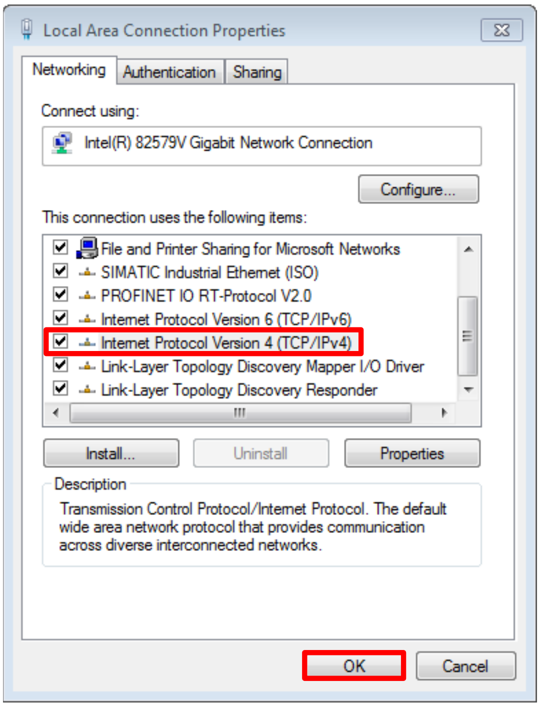
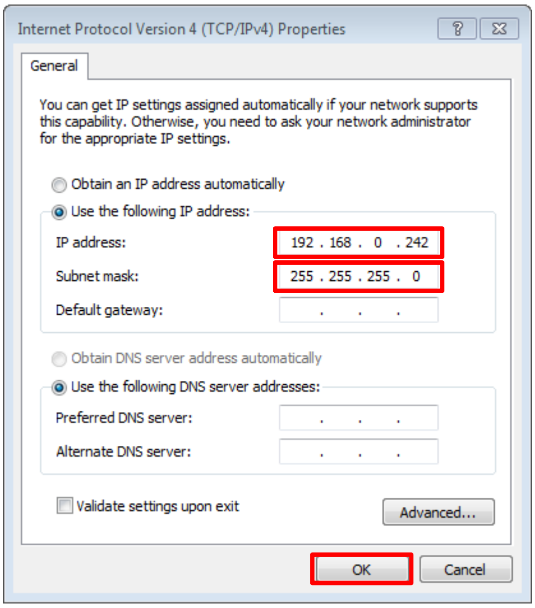
## 5.2 Network connections

The LAN network card of the programming device requires a static IP address to configure the controller and the HMI and to configure the weighing module. The following table describes the configuration of the LAN connection.

Table 5-2

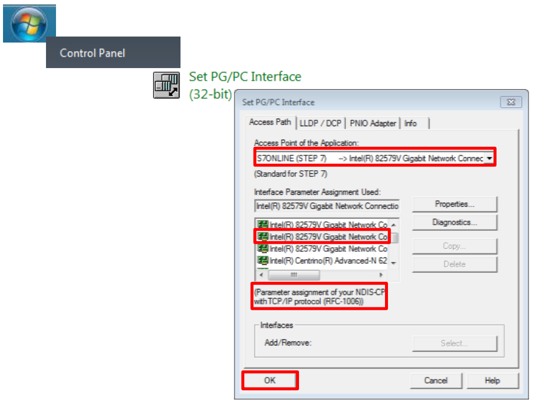
| No. | Action | Comment |
|---|---|---|
| 1. | Select "Start > Control Panel > Network and Sharing Center > Change adapter settings" to open the network connections.<br>• Select your network connection.<br>• Right-click to open the properties. | <br> Control Panel<br>Network and Sharing Center<br>Change adapter settings<br>Control Panel ▸ Network and Internet ▸ Ne<br>File Edit View Tools Advanced Help<br>Organize ▾ Disable this network device Diagnose this<br>Name Status<br>Local Area Connection<br>Mobile Broadband Co...<br>VMware Network Ada...<br>VMware Network Ada...<br>Wireless Network Con...<br>Wireless Network Con...<br>Disable<br>Status<br>Diagnose<br>Bridge Connections<br>Create Shortcut<br>Delete<br>Rename<br>Properties |

## 5.2 Network connections

| No. | Action | Comment |
|---|---|---|
| 2. | In “Networking”, select the “Internet Protocol Version 4 (TCP/IPv4)” item and open its properties. | <br>Local Area Connection Properties<br> <br>Connect using:<br>Intel(R) 82579V Gigabit Network Connection<br> <br>This connection uses the following items:<br>☑ File and Printer Sharing for Microsoft Networks<br>☑ SIMATIC Industrial Ethernet (ISO)<br>☑ PROFINET IO RT-Protocol V2.0<br>☑ Internet Protocol Version 6 (TCP/IPv6)<br>☑ Internet Protocol Version 4 (TCP/IPv4)<br>☑ Link-Layer Topology Discovery Mapper I/O Driver<br>☑ Link-Layer Topology Discovery Responder<br> <br>Description<br>Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.<br> |
| 3. | - Select “Use the following IP address”.<br>- Select an IP address in the CPU’s subnet mask.<br>- Accept the settings by clicking “OK” and “Close”. | <br>Internet Protocol Version 4 (TCP/IPv4) Properties<br> <br>You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.<br>○ Obtain an IP address automatically<br>● Use the following IP address:<br>IP address: 192 . 168 . 0 . 242<br>Subnet mask: 255 . 255 . 255 . 0<br>Default gateway: . . .<br>○ Obtain DNS server address automatically<br>● Use the following DNS server addresses:<br>Preferred DNS server: . . .<br>Alternate DNS server: . . .<br>□ Validate settings upon exit<br> |

### 5.2.1 Setting PG/PC interface

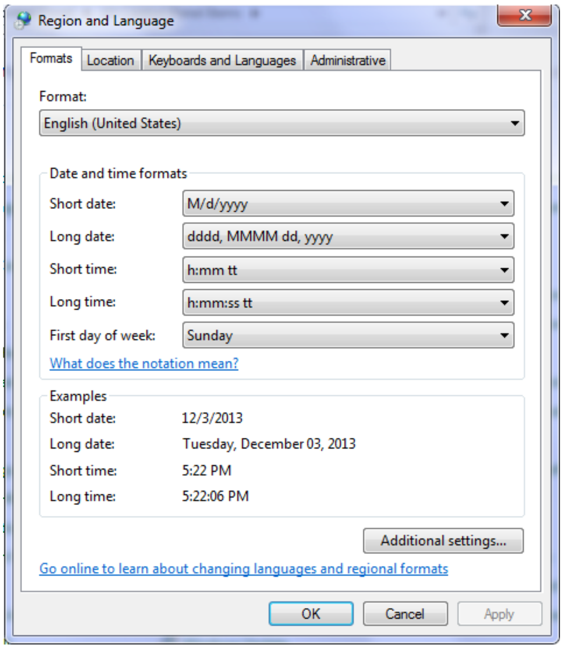
Table 5-3

| No. | Action | Comment |
|---|---|---|
| 1. | Open the PG/PC interface settings via “Start > Control Panel” to set the correct access path for STEP 7 V14.<br>• Select “S7ONLINE (STEP 7)” as the application's access point.<br>• Choose “(Assigning Parameters to Your NDIS CPs with TCP/IP protocol (RFC-1006))” to select your network adapter as the interface parameter assignment you are using.<br>Confirm the settings using “OK”. | <br> Control Panel<br>Set PG/PC Interface (32-bit)<br>Set PG/PC Interface<br>Access Path<br>LLDP / DCP<br>PNIO Adapter<br>Info<br>Properties...<br>Diagnostics...<br>Copy...<br>Delete<br>Interfaces<br>Add/Remove:<br>Select...<br>OK<br>Cancel<br>Help<br>Access Point of the Application:<br>S7ONLINE (STEP 7) --> Intel(R) 82579V Gigabit Network Connec<br>(Standard for STEP 7)<br>Interface Parameter Assignment Used:<br>Intel(R) 82579V Gigabit Network Connectio<br>Intel(R) 82579V Gigabit Network Co<br>Intel(R) 82579V Gigabit Network Co<br>Intel(R) 82579V Gigabit Network Co<br>Intel(R) Centrino(R) Advanced-N 62<br>(Parameter assignment of your NDIS-CP with TCP/IP protocol (RFC-1006)) |

## 5.3 Region and language options

By USA standards, the individual columns are separated by commas. This setting is required here, so as to ensure that the csv files can be opened correctly and directly from Microsoft Excel.

Table 5-4

| No. | Action | Comment |
|---|---|---|
| 1. | Open the region and language options via "Start > Control Panel“.<br>• Set the language to “English (USA)” in standards and format.<br>• Apply the settings with “OK”. | <br> Region and Language<br>Formats<br>Location<br>Keyboards and Languages<br>Administrative<br>What does the notation mean?<br>Additional settings...<br>Go online to learn about changing languages and regional formats<br>OK<br>Cancel<br>Apply<br>Format:<br>English (United States)<br>Date and time formats<br>Short date: M/d/yyyy<br>Long date: dddd, MMMM dd, yyyy<br>Short time: h:mm tt<br>Long time: h:mm:ss tt<br>First day of week: Sunday<br>Examples<br>Short date: 12/3/2013<br>Long date: Tuesday, December 03, 2013<br>Short time: 5:22 PM<br>Long time: 5:22:06 PM |

**Note** How to import data logs in CSV format in non-USA/UK versions of Microsoft Excel is described in the S7-1200 System Manual (\3\).

## 5.4 Configuring SIWAREX WP231 weighing module

The SIWAREX WP231 has to be configured accordingly to adjust the weighing module to the load cell used and the specific application case. New modules are delivered with "factory settings", which satisfies the majority of the applications. Using a command, these factory settings can also be restored later. The following configuration steps have to be made to adapt to the application:

1. Reset to factory settings
2. Specify the calibration and load cell parameters
   The individual elements are:
   - Weight unit
     The weight unit (e.g. 'g' or 'kg') is used for interpreting the weight display. It determines the unit of other parameters, such as maximum weighing range, numerical increment and calibration weight 1.
   - Maximum weighing range
     The maximum weighing range corresponds to the rated load $E_{max}$ of the load cell. For the load cell used this is $E_{max}$ = 3 kg.
   - Resolution of the load cell
     The resolution corresponds to the minimum scale interval $V_{min}$ of the load cell. For the SIWAREX WL260 SP-S AA load cell used, this is $V_{min} = E_{max}/15000$. This corresponds to a numerical increment d of 0.2 g.
   - Calibration weight 1
     "Calibration weight" added later to the scale during calibration. The calibration weight must be at least 5% of the rated load of all connected load cells. As a calibration weight you use an object, whose exact weight you know.

   The application requires the following values which deviate from the factory settings:

Table 5-5: Values deviating from the factory settings

| Parameter | Default | Application |
|---|---|---|
| Weight unit | ' kg ' | ' g ' |
| Maximum weighing range | 100 [→ 100kg] | 3000 [→ 3000g] |
| Numerical increment d | 0.1 [→ 0.1kg] | 0.2 [→ 0.2g] |
| Calibration weight 1 | 100 [→ 100kg] | 1100 [→ 1100g]<br>Known calibration weight which is available to the user |

**Note**

In this example, the SIWAREX WP231 weighing module is used with firmware V3.1.1 (\5\). This firmware allows the configuration and calibration via control commands. Chapter 7.2.2 demonstrates the settings required for these sets. You can also perform the basic commissioning with the help of the SIWATOOL V7 (Table 2-3) and the brief instructions (\7\).
The SIWATOOL V7 is essential to update the SIWAREX WP231 firmware.

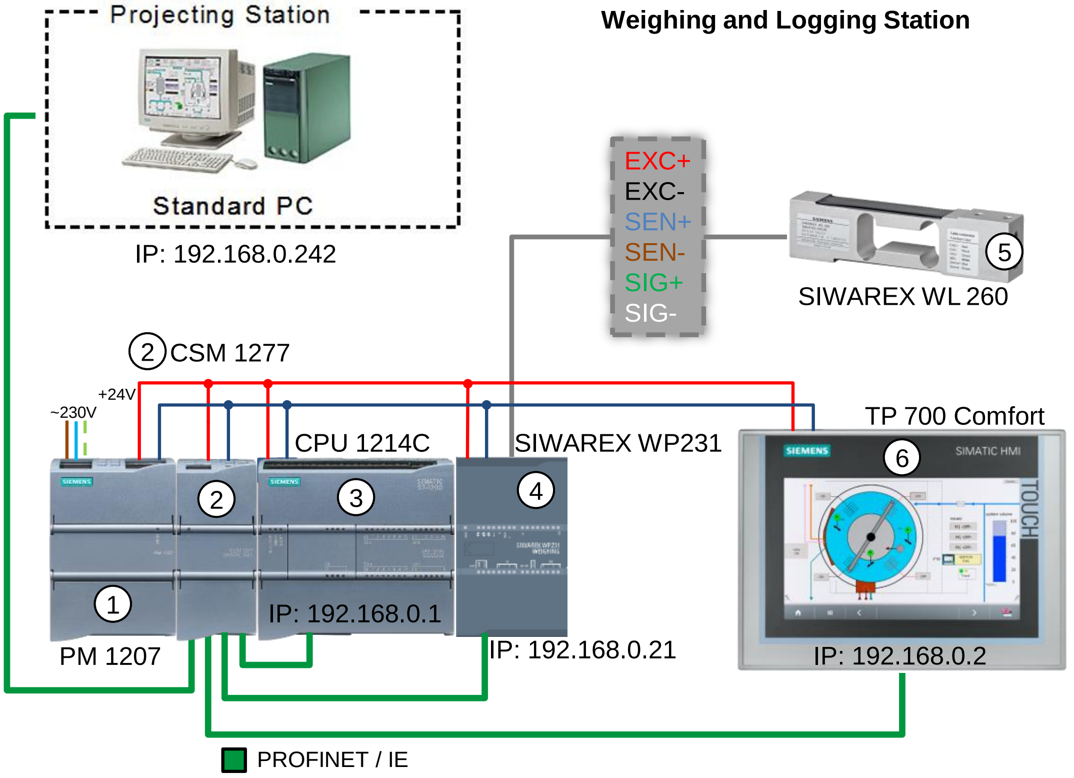
# 6 Installation and Commissioning

This chapter describes the steps to be taken to commission the application example using the code from the download and the hardware list.

## 6.1 Installing the hardware

The figure below shows the hardware setup of the application.

Figure 6-1



A **CPU 1214C** (3) is used as controller for set 6.

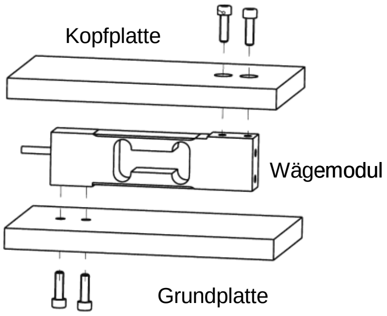
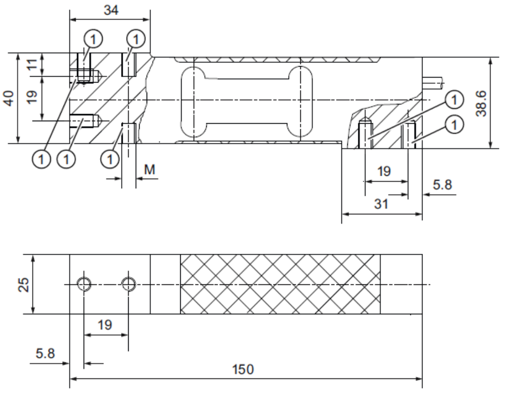
The **SIWAREX WP231** weighing module (4) is used as expansion module. The associated **SIWAREX WL260** load cell (5) is connected to SIWAREX WP231 via a 6-wire cable (plus shielding).

The 24V energy supply of the devices is provided by a **PM 1207** (1).

A Windows PC with **TIA Portal V14** (STEP 7 Basic V14 and WinCC Comfort V14) is used as configuration device for the S7-1200 controller as well as the **TP 700 Comfort**. The SIWAREX WP231 weighing module is configured via the SIMATIC S7-1200 by means of “write data record” as of firmware V1.1.

The **CSM 1277** switch (2) is used as node for the Ethernet communication between the modules (CPU 1214C and SIWAREX WP231), the **TP 700 Comfort** (6) operator panel and the configuration device.

Table 6-1

| No. | Action | Comment |
|---|---|---|
| 1. | Mount the PM1207 power module, the CPU 1214C and the SIWAREX WP231 weighing module onto the hat rail and establish the bus connection to the CPU with the slider switch of the weighing module.<br>Do not yet supply the PM1207 power module with the power supply (230V~). | see Figure 6-1 |
| 2. | Connect the CSM1277 switch, the CPU 1214C, the SIWAREX WP231 weighing module and the TP700 Comfort panel with 24V DC supply voltage of the PM1207 power module. | see Figure 6-1 |
| 3. | A base and top plate has to be attached to the SIWAREX WL260 SP-S AA load cell. The drawing in the “Note” column is to be used for base plate and top plate. Screws (M6 x 15mm) with washers are required.<br><br> | <br> 34<br>①<br>40<br>19<br>11<br>M<br>19<br>31<br>5.8<br>38.6<br>25<br>19<br>5.8<br>150<br>Gewindebohrungen<br>Bezeichnung Gewinde Gewindetiefe<br>① M6 15 mm |
| 4. | Connect the SIWAREX WL260 SP-S AA load cell to the SIWAREX WP231 weighing module. | see Figure 6-1 |
| 5. | Connect the CPU, the weighing module, the Comfort Panel and the programming device with the help of the RJ45 cable to the CSM1277 switch. | see Figure 6-1 |
| 6. | Connect all ground connections to earth. | |
| 7. | Provide the PM1207 power module with the power supply (230V~). | |

## 6.2 Installing the software (download)

This chapter describes the steps for installing the sample code.

**Note** At this point, it is assumed that the necessary software has been installed on your computer and that you are familiar with the software.

### Preliminary remark

For the startup, we offer you software examples with test code and test parameters as a download. The software examples support you during the first steps and tests with set 6. They enable a quick test of hardware and software interfaces between the products described in the set.

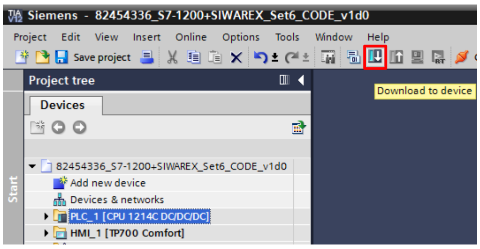
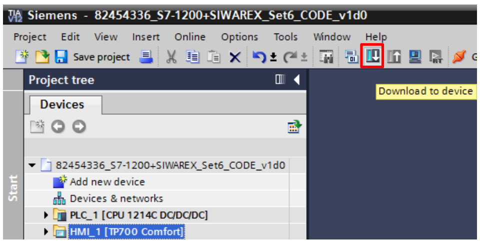
The software examples are always assigned to the components used in the set and show their basic interaction. However, they are not real applications in the sense of a technological problem solution with definable properties.

## 6.3 Downloading the startup code

The software examples are available on the HTML page from which you downloaded this document.

### Downloading the TIA Portal project

Table 6-2

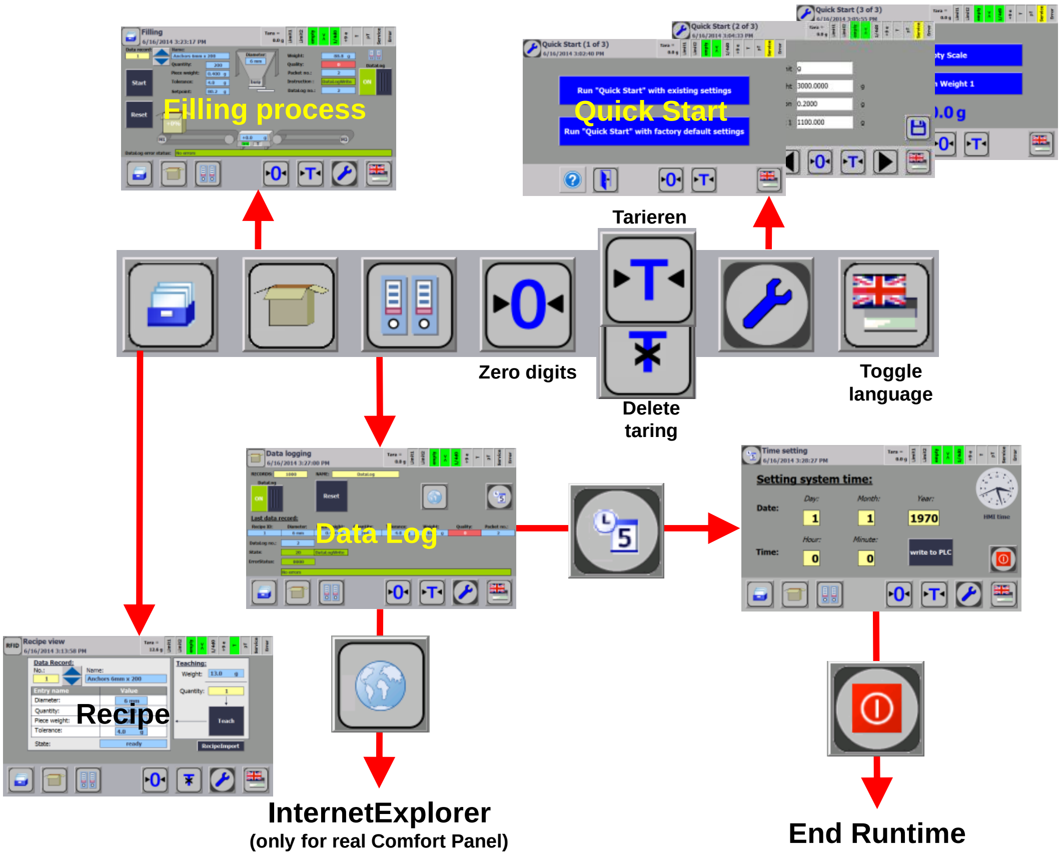
| No. | Action | Comment |
|---|---|---|
| 1. | Unzip and open the project from Table 2-4 using TIA Portal V14. | |
| 2. | Select the “PLC_1” control project folder and confirm the button for loading the CPU. |  Siemens - 82454336_S7-1200+SIWAREX_Set6_CODE_v1d0<br>Project Edit View Insert Online Options Tools Window Help<br>Save project<br>Project tree<br>Devices<br>Download to device<br>82454336_S7-1200+SIWAREX_Set6_CODE_v1d0<br>Add new device<br>Devices & networks<br>PLC_1 [CPU 1214C DC/DC/DC]<br>HMI_1 [TP700 Comfort]<br>Start |
| 3. | Select the project folder for the “HMI_1” operator panel and click the button to load it into the Comfort Panel.<br>Alternatively, you can simulate the TP700 Comfort operator panel also without hardware via the TIA Portal. |  Siemens - 82454336_S7-1200+SIWAREX_Set6_CODE_v1d0<br>Project Edit View Insert Online Options Tools Window Help<br>Save project<br>Project tree<br>Devices<br>Download to device<br>82454336_S7-1200+SIWAREX_Set6_CODE_v1d0<br>Add new device<br>Devices & networks<br>PLC_1 [CPU 1214C DC/DC/DC]<br>HMI_1 [TP700 Comfort]<br>Start |

# 7 Operating the Application

## 7.1 Overview

The following picture shows the menu navigation via the toolbar.
It can be selected from any screen.

Figure 7-1



The user interface consists of 6 menu items:

- Filling process (start screen)
- Recipe with teach function
- "Data Log” process data logging
- Quick start for configuring the weighing module and calibrating the scale
- Time setting for synchronization between CPU and HMI
- Call of the Internet Explorer in Windows CE
(not possible for the simulation via WinCC V14)

### 7.1.1 Toolbar (footer)

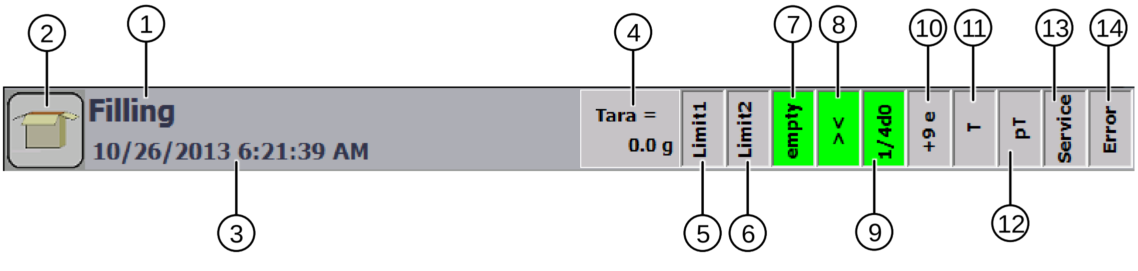
The toolbar consists of 7 buttons:

- Via you get to the recipe view. This is where you can execute the teach function.
- Via you get to the display of the filling process. This is also the start screen.
- Via you get to the overview of the “Data Log” process data logging.
- The scale can be set to zero via ▸0◂. This is used when the scale is empty but does not display precisely 0 g because of soiling.
- The scale can be tared via ▸T◂. This is where scale is set to zero and the weight difference is saved as tare weight. This function is required to determine the net weight (content of the container). These buttons are only visible when taring the scale.
- The taring can be deleted via . The tare value is deleted and the scale will show the gross weight again (weight of container + content). These buttons are only visible when taring the scale.
- Via 🔧 you get to the quick start menu of the weighing module.
- You can toggle between German and English via 🇬🇧.

### 7.1.2 Header with scale status

The header provides you with information on the current screen and the status of the weighing module.

Figure 7-2



The header shows the following:

1. Screen name
2. Screen icon (identical with the button icon in the toolbar)
3. Local time of the controller
4. Tara: Tare value display (weight of the container during taring)
5. Limit value 1 (configurable via SIWATOOL V7)
6. Limit value 2 (configurable via SIWATOOL V7)

7. Empty: The scale is in a defined empty range (configurable via SIWATOOL V7).
8. Standstill: The weight is in a stable state.
9. ¼ d zero: The scale is +/- of a quarter numerical increment to the zero point (important for scales requiring verified calibration).
10. Max. 9e: The current weight is more than 9 numerical increments above the defined weighing range (important for scales requiring verified calibration).
11. Tared: The scale is tared.
12. Preset tare: The scale was tared via a specified tare value.
13. Service: The service mode of the scale is switched on.
14. Error: The operation of the scale is faulty.

## 7.2 Commissioning

The following steps have to be performed to correctly operate the application.

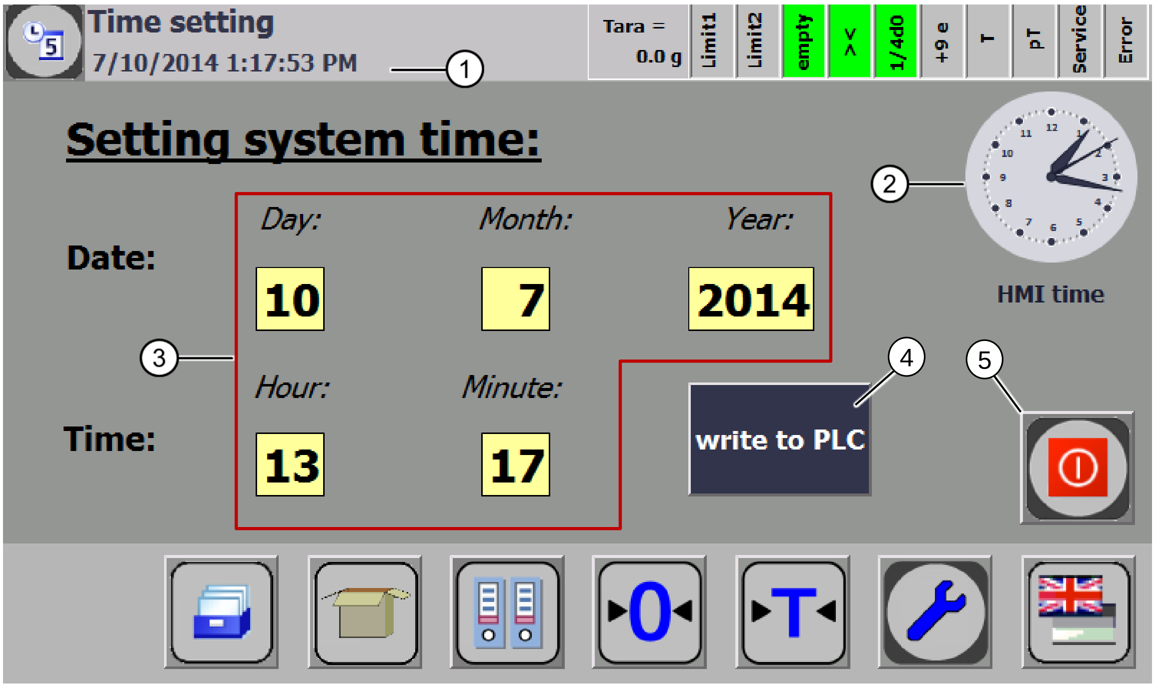
### 7.2.1 Setting time

The application has a time synchronization between PLC and HMI.

For process data acquisition via “DataLog”, the system time of the controller is used as time stamp.

This screen enables you to set the CPU system time.
Depending on the CPU settings for time zone and daylight saving time, there is an offset between system and local time, which is synchronized via the HMI time in the header.

Figure 7-3



Time setting
7/10/2014 1:17:53 PM

Tara = 0.0 g

Setting system time:

Date:

Day: 10

Month: 7

Year: 2014

Time:

Hour: 13

Minute: 17

The following objects are important for setting the time:

1. Local time of the controller
2. Local time of the operator panel (synchronization is only possible for real HMI)
3. Input fields for setting the CPU system time (please note the possible offset between system and local time when entering; in the example project, project system and local time are identical)
4. Button for accepting the set CPU system time
5. Ending runtime (for example, to make time zone settings in Windows CE)

### 7.2.2 Quick Start

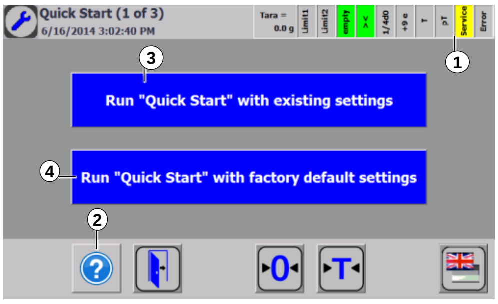
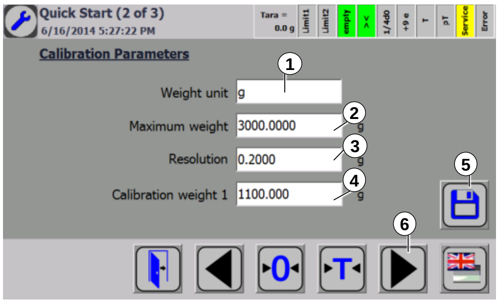
The scale can be calibrated via the SIWATOOL configuration package (Table 2-3) or via this screen.

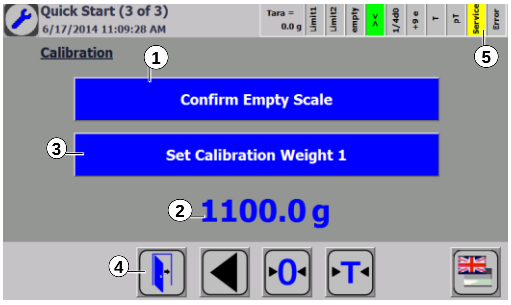
You need a known weight which should be at least 5% of the rated load of the load cell.

The SIWAREX WP231 weighing module is configured via the SIMATIC S7-1200 by means of “write data record” as of firmware V1.1.

In order to configure the load cell and then to calibrate the scale, proceed as follows:

Table 7-1

| No. | Action | Comment |
|---|---|---|
| 1. | Click the icon for the quick start menu in the footer. | |
| 2. | 1. By opening the quick start menu, the weighing module is switched to service mode.<br>2. Via the question mark you will receive information about the quick start menu.<br>You have the option to continue the configuration with the existing (3) or the factory settings (4).<br>3. Click the button to continue with the factory settings. | <br>Quick Start (1 of 3)<br>6/16/2014 3:02:40 PM<br> Tara = 0.0 g<br>Limit1<br>Limit2<br>empty<br>><<br>1/4d0<br>+9 e<br>T<br>pT<br>Service<br>Error<br>1<br>3<br>Run "Quick Start" with existing settings<br>4<br>Run "Quick Start" with factory default settings<br>2 |
| 4. | Enter the following parameters:<br>1. Weight unit: g (gram)<br>2. Maximum weighing range: 3000 g (rated load $E_{max}$ of the load cell = 3 kg)<br>3. Numerical increment d: 0.2 g (minimum scale interval $V_{min}$ = $E_{max}$/15000)<br>4. Calibration weight 1: minimum 5% of rated load Emax<br>5. Save the values on the weighing module<br>6. Go to the next menu item.<br>The technical data of the load cell can be found in the operating instruction for the SIWAREX WL200 load cells (\4\). | <br>Quick Start (2 of 3)<br>6/16/2014 5:27:22 PM<br>Calibration Parameters<br>Weight unit g<br>Maximum weight 3000.0000 g<br>Resolution 0.2000 g<br>Calibration weight 1 1100.000 g<br> Tara = 0.0 g<br>Limit1<br>Limit2<br>empty<br>><<br>1/4d0<br>+9 e<br>T<br>pT<br>Service<br>Error<br>1<br>2<br>3<br>4<br>5<br>6 |

| No. | Action | Comment |
|---|---|---|
| 5. | 1. Confirm the empty weight of the empty scale (the weight display (2) will then show “0.0 g”).<br>2. Then put the calibration weight onto the scale and confirm its validity. The weight display (2) then shows the previously configured value of the “Calibration weight 1”.<br>The scale is now calibrated.<br>4. When exiting the quick start menu, the service mode is disabled (5). |  |

## 7.3 Live demo

This chapter explains the procedure for production.

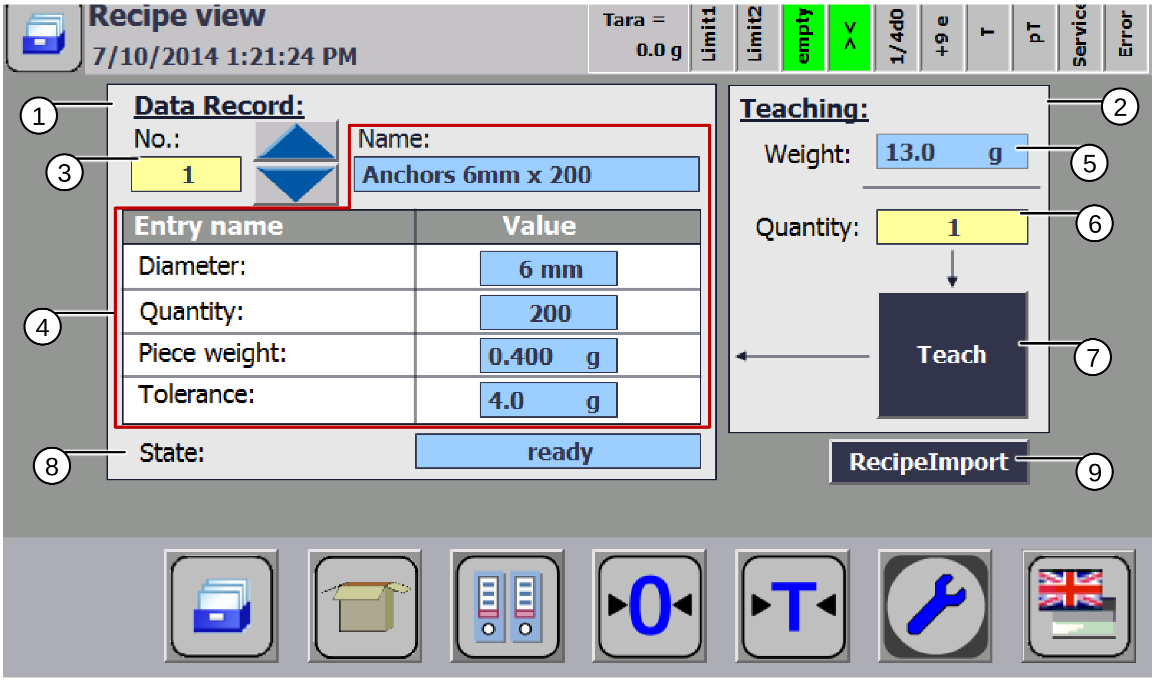
The application-specific HMI operation consists of the screens:

- Recipe
- Filling process
- Data Log

### 7.3.1 Recipe selection and individual weight determination

In this screen you can select the product to be produced, as well as determine the individual weight via the teach function.
In addition, there is the option to download the exported recipe data as csv file on a PC, to edit it and to upload the changed data again on the web server and to import it to the recipe data of the controller.

Figure 7-4



Recipe view
7/10/2014 1:21:24 PM

Data Record:
No.: 1
Name: Anchors 6mm x 200

| Entry name | Value |
|---|---|
| Diameter: | 6 mm |
| Quantity: | 200 |
| Piece weight: | 0.400 g |
| Tolerance: | 4.0 g |

State: ready

Teaching:
Weight: 13.0 g
Quantity: 1

The “recipe” screen consists of an overview field with information on the selected recipe data record (1) and an operator panel for the teaching function (2).

In the exported data blocks, a “wall anchor” recipe has been created for this application with the following three data records:

1. Wall anchor 6mm x 200
2. Wall anchor 8mm x 100
3. Wall anchor 10mm x 50

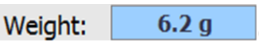
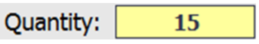
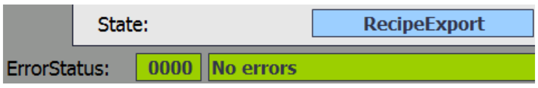
Select the goods to be packaged via the data record no (3) or the arrow buttons. As a result, the data information (4) is read from the load memory via the “READ_DBL” function:

- Data record name
- Wall anchor diameter
- Number of wall anchor
- Individual weight of a wall anchor
- Weight tolerance for the quality assessment of the quantity filled

### Teach function

To determine the individual weight of a wall anchor, proceed as follows:

Table 7-2

| No. | Action | Comment |
|---|---|---|
| 1. | Place a container on the load cell. | |
| 2. | Tare the scale. | |
| 3. | Fill the container with a counted number of wall anchors. |  Weight: 6.2 g (5) displays the weight of the wall anchors. |
| 4. | Enter the number of wall anchors in (6). |  Quantity: 15 |
| 5. | Click the teach button (7). | Teach |
| 6. | The weight is divided by the number of wall anchors and indicates the individual weight of one wall anchor.<br>The newly determined individual weight is written in the selected recipe data record in the exported “RECIPE_DB” data block via the “WRIT_DBL” function. | Piece weight: 0.413 g |
| 7. | Then the entire recipe data records are exported in the relocated “RECIPE_DB” data block via the “RecipeExport” function in the “RECIPE_DB.csv” file on the CPU web server. |  State: RecipeExport<br>ErrorStatus: 0000 No errors |
| 8. | Finally take the container from the scale and delete the taring. | |

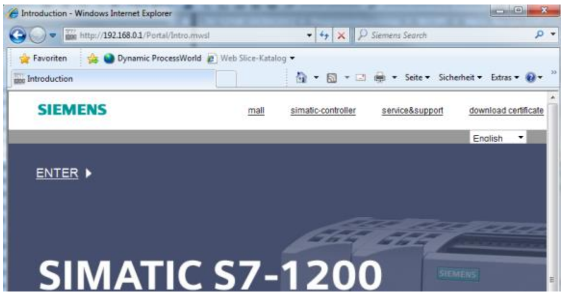
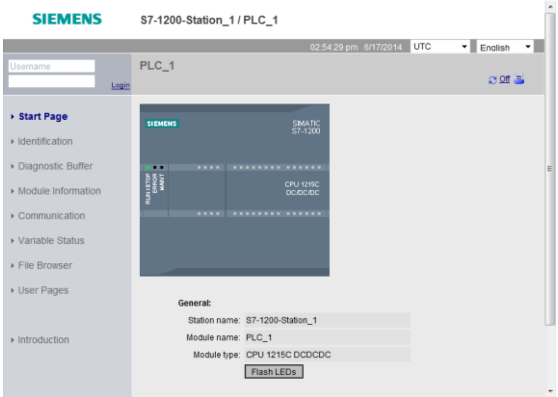
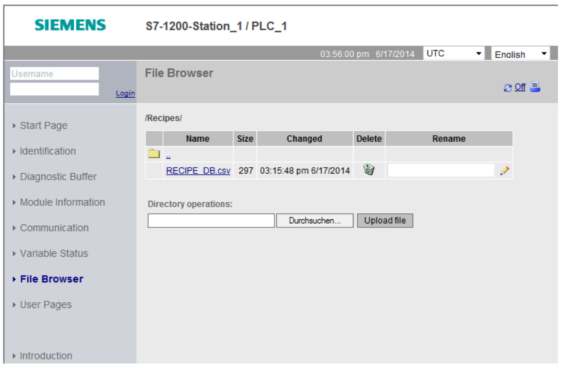
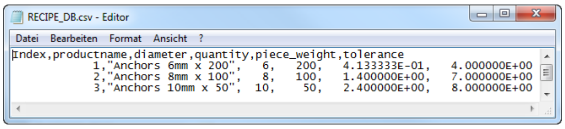
### 7.3.2 Changing the recipe data via the web server

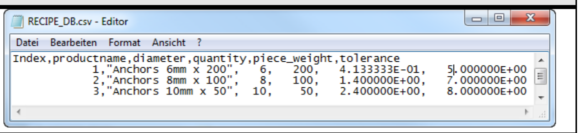
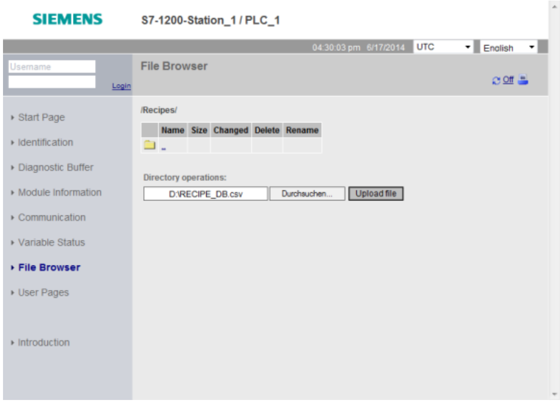
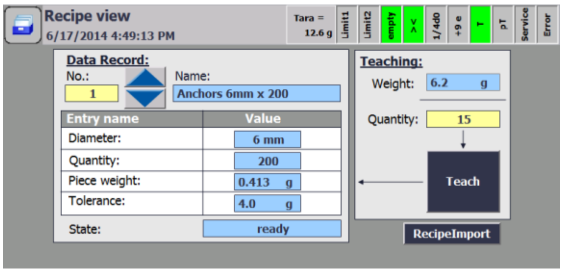
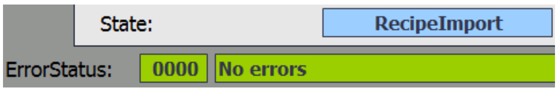
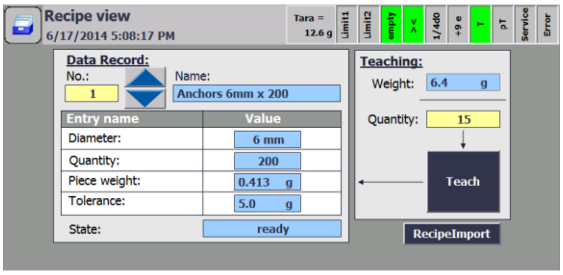
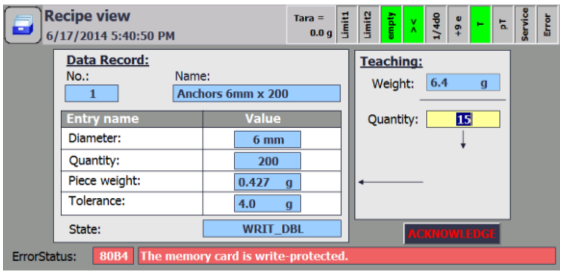
During initialization of the controller, as well as when changing the recipe data using the teach function, the relocated recipe data records are copied into the controller via the “RecipeExport” function as csv file on the web server.

You have the option to read out the csv file with the exported recipe data via the Internet Explorer in Windows CE on your Comfort Panel or via an Internet Browser on your programming device.

Proceed as follows:

Table 7-3

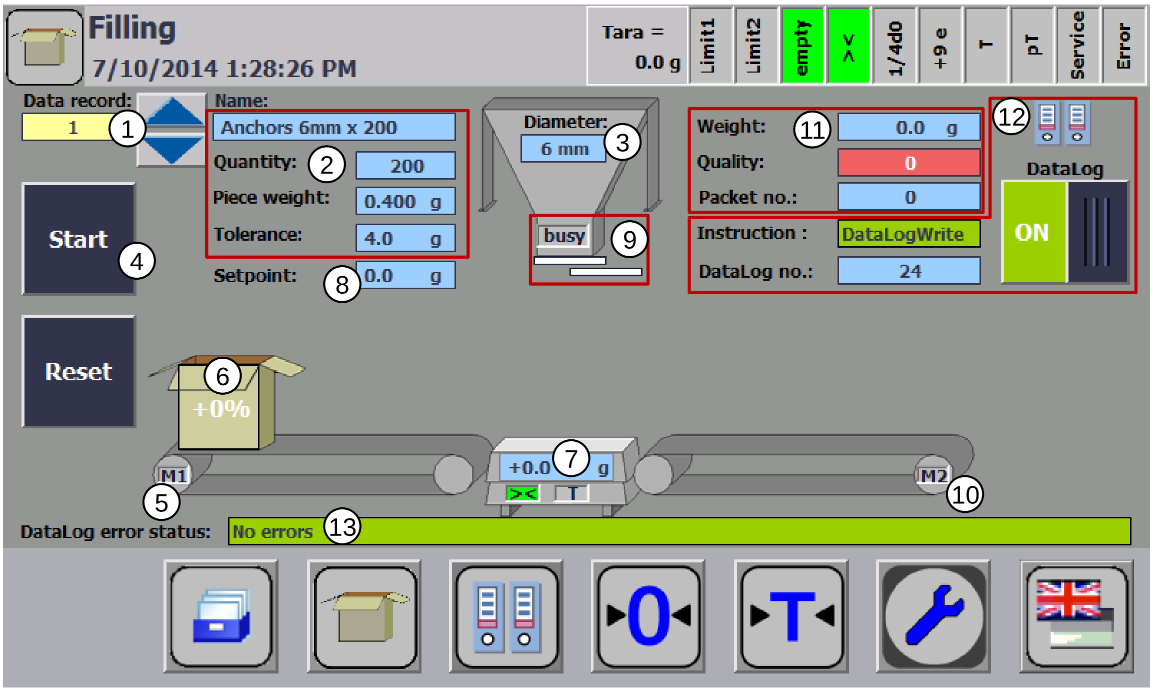
| No. | Action | Comment |
|---|---|---|
| 1. | - Start the Internet Explorer.<br>- As address, enter the IP address of the CPU:<br>here: http://192.168.0.1<br>- Click on "ENTER". | <br> Introduction - Windows Internet Explorer<br>http://192.168.0.1/Portal/Intro.mwsl<br>Siemens Search<br>Favoriten<br>Dynamic ProcessWorld<br>Web Slice-Katalog<br>Introduction<br>Seite<br>Sicherheit<br>Extras<br>SIEMENS<br>mall<br>simatic-controller<br>service&support<br>download certificate<br>English<br>ENTER<br>SIMATIC S7-1200 |
| 2. | You get to the start page. Here, you have the option to logon to the web server with the access data previously determined in the user administration in order to receive certain authorizations. In the example project the administrative access level for all users was created.<br>- Open the file browser.<br>- Open the “Recipes” folder in the file browser. | <br>General:<br>Station name: S7-1200-Station_1<br>Module name: PLC_1<br>Module type: CPU 1215C DCDCDC<br> SIEMENS<br>S7-1200-Station_1 / PLC_1<br>02:54:29 pm 6/17/2014<br>UTC<br>English<br>Username<br>Login<br>PLC_1<br>Off<br>Start Page<br>Identification<br>Diagnostic Buffer<br>Module Information<br>Communication<br>Variable Status<br>File Browser<br>User Pages<br>Introduction<br>SIMATIC S7-1200<br>CPU 1215C DC/DC/DC<br>Flash LEDs |
| 3. | As a result, the list with all available recipe files opens, including file size specification and system time stamp of the last export.<br>You also have the option here to delete the recipe file or to rename it.<br>- Click on the name of the “RECIPE_DB.csv” file and save it on your hard disk. | <br>File Browser<br>/Recipes/<br>Name \| Size \| Changed \| Delete \| Rename<br>..<br>RECIPE_DB.csv \| 297 \| 03:15:48 pm 6/17/2014<br>Directory operations:<br> SIEMENS<br>S7-1200-Station_1 / PLC_1<br>03:56:00 pm 6/17/2014<br>UTC<br>English<br>Username<br>Login<br>Off<br>Start Page<br>Identification<br>Diagnostic Buffer<br>Module Information<br>Communication<br>Variable Status<br>File Browser<br>User Pages<br>Introduction<br>Durchsuchen...<br>Upload file |
| 4. | - Afterwards open the “RECIPE_DB.csv” file with the Windows Editor (Notepad) and change, for example, the tolerance specification in the last column for a recipe data record and save the file.<br>The number of characters between the commas must not be changed! | <br>RECIPE_DB.csv - Editor<br>`Index,productname,diameter,quantity,piece_weight,tolerance`<br>`1,"Anchors 6mm x 200",   6,   200,   4.133333E-01,   4.000000E+00`<br>`2,"Anchors 8mm x 100",   8,   100,   1.400000E+00,   7.000000E+00`<br>`3,"Anchors 10mm x 50",  10,    50,   2.400000E+00,   8.000000E+00`<br> |

| No. | Action | Comment |
|---|---|---|
| | | <br>RECIPE_DB.csv - Editor<br>Index,productname,diameter,quantity,piece_weight,tolerance<br>1,"Anchors 6mm x 200", 6, 200, 4.133333E-01, 5.000000E+00<br>2,"Anchors 8mm x 100", 8, 100, 1.400000E+00, 7.000000E+00<br>3,"Anchors 10mm x 50", 10, 50, 2.400000E+00, 8.000000E+00 |
| 5. | - Delete or rename the "RECIPE_DB.csv" file on the web server.<br>- Browse your hard disk for the changed "RECIPE_DB.csv" and upload the file.<br>Uploading is only possible if there is no file with the same name on the web server.<br>Then the uploaded file is listed with the current time stamp | <br>S7-1200-Station_1 / PLC_1<br>04:30:03 pm 6/17/2014 UTC English<br>File Browser<br>/Recipes/<br>Directory operations: D:\RECIPE_DB.csv |
| 6. | - Press the RecipeImport button in the recipe view on the operator panel.<br>In the example project the "RecipeImport" function is triggered via the operator panel. However, they could also be executed via the web server (tag status or user pages). | <br>Recipe view 6/17/2014 4:49:13 PM<br>Tara = 12.6 g<br>Data Record: No.: 1 Name: Anchors 6mm x 200<br>Diameter: 6 mm<br>Quantity: 200<br>Piece weight: 0.413 g<br>Tolerance: 4.0 g<br>State: ready<br>Teaching: Weight: 6.2 g Quantity: 15 |
| 7. | The changed "RECIPE_DB.csv" file on the web server is imported to the exported "RECIPE_DB" data block in the load memory, using the "RecipeImport" function. | <br>State: RecipeImport<br>ErrorStatus: 0000 No errors |
| 8. | Then the selected recipe data record is copied to the main memory via the "READ_DBL" function.<br>The "READ_DBL" function is executed with every change of the selected recipe data record. | <br>Recipe view 6/17/2014 5:08:17 PM<br>Tara = 12.6 g<br>Data Record: No.: 1 Name: Anchors 6mm x 200<br>Diameter: 6 mm<br>Quantity: 200<br>Piece weight: 0.413 g<br>Tolerance: 5.0 g<br>State: ready<br>Teaching: Weight: 6.4 g Quantity: 15 |
| 9. | Possibly occurring errors during the processing of the functions ("step") READ_DBL, WRIT_DBL, RecipeImport or RecipeExport are displayed in "ErrorStatus" as hexadecimal code with error description and have to be acknowledged to reset the ACKNOWLEDGE sequence. | <br>Recipe view 6/17/2014 5:40:50 PM<br>Tara = 0.0 g<br>Data Record: No.: 1 Name: Anchors 6mm x 200<br>Diameter: 6 mm<br>Quantity: 200<br>Piece weight: 0.427 g<br>Tolerance: 4.0 g<br>State: WRIT_DBL<br>Teaching: Weight: 6.4 g Quantity: 15<br>ErrorStatus: 80B4 The memory card is write-protected. |

### 7.3.3 Filling process

This screen presents the production of the wall anchors that are to be packaged.

Figure 7-5



Filling

7/10/2014 1:28:26 PM

Tara = 0.0 g

Data record: 1

Name: Anchors 6mm x 200

Quantity: 200

Piece weight: 0.400 g

Tolerance: 4.0 g

Setpoint: 0.0 g

Diameter: 6 mm

Weight: 0.0 g

Quality: 0

Packet no.: 0

Instruction : DataLogWrite

DataLog no.: 24

DataLog error status: No errors

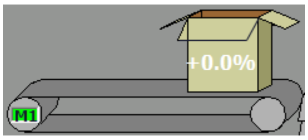
Select the wall anchors (size and number) to be packaged through the data record no (1). As a result, the field (2) will show the recipe data:

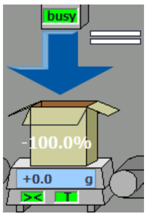
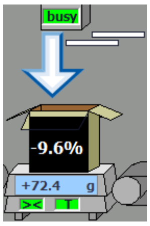
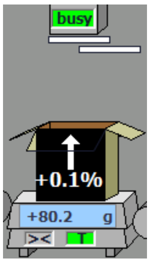
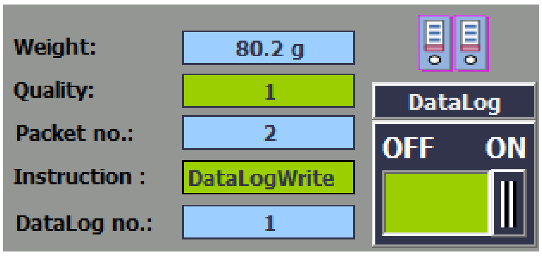
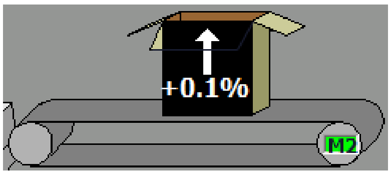
- Data record name
- Number of wall anchor
- Individual weight of a wall anchor
- Weight tolerance for the quality assessment of the quantity filled

The storage container contains the wall anchor size depending on the selected recipe data record with the respective diameter (3).

To be able to start the filling of the selected wall anchor recipe, proceed as follows:

Table 7-4

| No. | Action | Comment |
|---|---|---|
| 1. | Place a container on the load cell. | The scale will then display the weight of the container: +12.4 g (7) |
| 2. | Click the start button (4). | Start |
| 3. | This clears the taring and the conveyor belt "M1" (5) moves the packet (6) onto the scale. |  |

| No. | Action | Comment | |
|---|---|---|---|
| 4. | When the scale is reached, and the conditions standstill and weight recording larger than zero have been fulfilled:<br>• the scale is tared,<br>• the setpoint specification (8) is calculated:<br>$m_{setpoint}$ = $m_{setpoint}$ (recipe) + $m_{resolution}$ (load cell)<br>• the sliders for filling (9) are opened. |  | The filling process is indicated by the flashing blue arrow.<br>The setpoint weight deviation is displayed on the packet. |
| 5. | Fill the container with the selected wall anchor type. |  | As of 90% of the setpoint weight (>-10% of the weight deviation) the 2nd slider is closed and reduces the filling speed.<br>The slow filling speed is indicated by the flashing white arrow. |
| 6. | Continue to fill the container with the selected dowel type. |  | From 100% of the setpoint weight (>=0% of the weight deviation) the 1st slider is also closed and filling is finished. |
| 7. | The weight reached is compared with the setpoint specifications as soon as the scale comes to a standstill.<br>The quality is positive (true), when<br>$m_{setpoint} <= m_{actual} <= m_{setpoint} + m_{torance}$<br>Outside of these limits, the quality is assessed as negative (false). The determined values are saved as new entry in the DataLog file. | Weight: 80.2 g<br>Quality: 1<br>Packet no.: 2<br>Instruction : DataLogWrite<br>DataLog no.: 1<br>DataLog<br>OFF ON | |
| 8. | Finally the assessed packet is removed.<br>Remove the packet from the scale and place a new empty container on the empty scale for the next filling process. |  | |

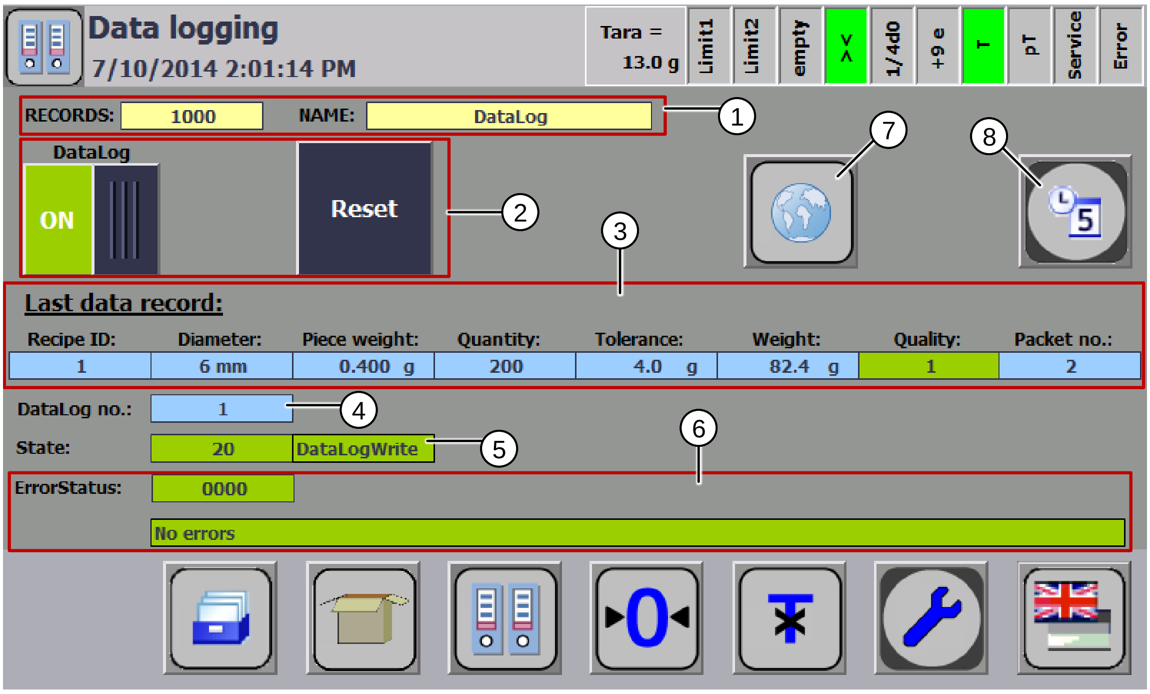
During transport and filling, the process can be stopped via Stop and continued with Continue.

The filling process is cancelled via Reset and the packet is reset to the initial position on the conveyor belt simulation.

### 7.3.4 DataLog

This screen is used to evaluate the status of the process data logging.

Figure 7-6



Data logging
7/10/2014 2:01:14 PM

Tara = 13.0 g

RECORDS: 1000 NAME: DataLog

Last data record:

| Recipe ID: | Diameter: | Piece weight: | Quantity: | Tolerance: | Weight: | Quality: | Packet no.: |
|---|---|---|---|---|---|---|---|
| 1 | 6 mm | 0.400 g | 200 | 4.0 g | 82.4 g | 1 | 2 |

DataLog no.: 1

State: 20 DataLogWrite

ErrorStatus: 0000

No errors

1. Determine the maximum number of data records and the name of the DataLog file via the input fields “RECORDS” and “NAME”.
2. The sequence of the “DataLog” FB is enabled via the “ON” button and the csv file is created and opened according to the specifications of the input fields “RECORDS” and “NAME” (if not yet existing).
   When disabling ("OFF”), the csv file is closed and it can be deleted via the web server in this state.

   The sequence of the “DataLog” FB is reset via the “Reset” button. The csv file is created and opened according to the specifications of the input fields “RECORDS” and “NAME”. Prerequisite for this is that there is no file yet with the same name.
3. The last data record of the DataLog file is displayed with the logged process values.
4. After each filling, the determined process data is logged and the DataLog number is incremented by one.
5. The status of the sequence specifies at which step the “DataLog” FB is located:
   - Step 0: DataLogCreate
   - Step 10: DataLogOpen
   - Step 20: DataLogWrite
6. If there are any possible errors, the hexadecimal error code is displayed with the respective error message.

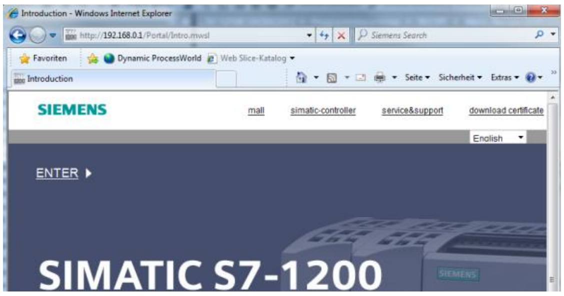
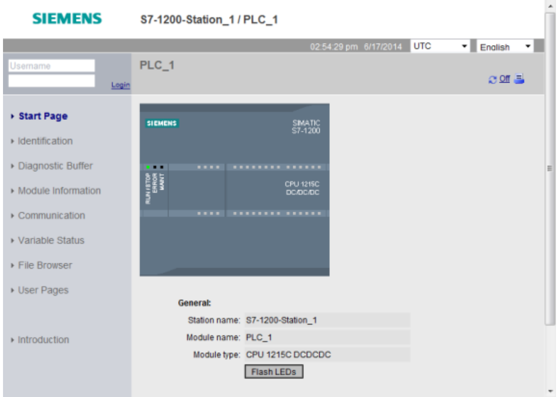
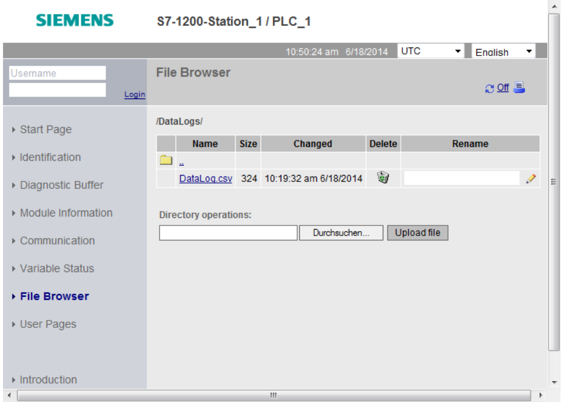
7. Via this button you get to the web server of the S7-1200 controller (only possible for real Comfort Panel).
8. Via this button you get to the time setting of the S7-1200 controller.

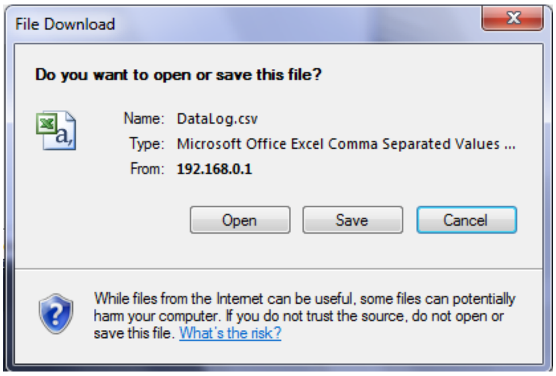
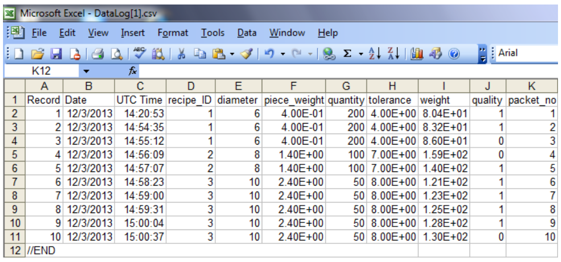
### 7.3.5 Reading out the DataLog file

You have the option to read out the csv file with the recorded process data via the Internet Explorer in Windows CE on your Comfort Panel or via an Internet Browser on your programming device.

Proceed as follows:

Table 7-5

| No. | Action | Comment |
|---|---|---|
| 1. | • Start the Internet Explorer.<br>• As address, enter the IP address of the CPU:<br>here: http://192.168.0.1<br>• Click on "ENTER". |  Introduction - Windows Internet Explorer<br>http://192.168.0.1/Portal/Intro.mwsl<br>Siemens Search<br>Favoriten<br>Dynamic ProcessWorld<br>Web Slice-Katalog<br>Introduction<br>Seite<br>Sicherheit<br>Extras<br>SIEMENS<br>mall<br>simatic-controller<br>service&support<br>download certificate<br>English<br>ENTER<br>SIMATIC S7-1200 |
| 2. | You get to the start page. Here, you have the option to logon to the web server with the access data previously determined in the user administration in order to receive certain authorizations. In the example project the administrative access level for all users was created.<br>• Open the file browser.<br>• Open the “DataLogs” folder in the file browser. |  SIEMENS<br>S7-1200-Station_1 / PLC_1<br>02:54:29 pm 6/17/2014<br>UTC<br>English<br>Username<br>Login<br>PLC_1<br>Off<br>Start Page<br>Identification<br>Diagnostic Buffer<br>Module Information<br>Communication<br>Variable Status<br>File Browser<br>User Pages<br>Introduction<br>SIEMENS<br>SIMATIC S7-1200<br>RUN / STOP<br>ERROR<br>MAINT<br>CPU 1215C DC/DC/DC<br>Flash LEDs<br>General:<br>Station name: S7-1200-Station_1<br>Module name: PLC_1<br>Module type: CPU 1215C DCDCDC |
| 3. | As a result, the list with all available DataLog files opens, including file size specification and system time stamp of the last access.<br>You also have the option here to delete the DataLog files or to rename them.<br>They have to be closed to be able to do this.<br>• Click on the name of the “DataLog.csv” file. |  SIEMENS<br>S7-1200-Station_1 / PLC_1<br>10:50:24 am 6/18/2014<br>UTC<br>English<br>Username<br>Login<br>File Browser<br>Off<br>Start Page<br>Identification<br>Diagnostic Buffer<br>Module Information<br>Communication<br>Variable Status<br>File Browser<br>User Pages<br>Introduction<br>Durchsuchen...<br>Upload file<br>/DataLogs/<br>Name \| Size \| Changed \| Delete \| Rename<br>..<br>DataLog.csv \| 324 \| 10:19:32 am 6/18/2014<br>Directory operations: |

<table>
<tr><th>No.</th><th>Action</th><th>Comment</th></tr>
<tr><td>4.</td><td>The “File Download” window gives you the option to open the csv file directly with a selected program or to save it first.<br>• Click the “Open” button.</td><td>
File Download<br>Do you want to open or save this file?<br>Name: DataLog.csv<br>Type: Microsoft Office Excel Comma Separated Values ...<br>From: 192.168.0.1<br>While files from the Internet can be useful, some files can potentially harm your computer. If you do not trust the source, do not open or save this file. What's the risk?
</td></tr>
<tr><td>5.</td><td>Due to the previously defined default and format settings (see chapter 5.3), the csv file will be correctly displayed in Microsoft Excel with the columns<br>• “Record” entry number<br>• "Date"<br>• “UTC Time” time stamp<br>• “recipe_ID” recipe number<br>• "diameter"<br>• "piece_weight" individual weight<br>• "quantity“ number of pieces<br>• "tolerance"<br>• "weight" measured weight<br>• "quality"<br>• "packet_no"</td><td>
Microsoft Excel - DataLog[1].csv

<table>
<tr><th></th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th><th>G</th><th>H</th><th>I</th><th>J</th><th>K</th></tr>
<tr><td>1</td><td>Record</td><td>Date</td><td>UTC Time</td><td>recipe_ID</td><td>diameter</td><td>piece_weight</td><td>quantity</td><td>tolerance</td><td>weight</td><td>quality</td><td>packet_no</td></tr>
<tr><td>2</td><td>1</td><td>12/3/2013</td><td>14:20:53</td><td>1</td><td>6</td><td>4.00E-01</td><td>200</td><td>4.00E+00</td><td>8.04E+01</td><td>1</td><td>1</td></tr>
<tr><td>3</td><td>2</td><td>12/3/2013</td><td>14:54:35</td><td>1</td><td>6</td><td>4.00E-01</td><td>200</td><td>4.00E+00</td><td>8.32E+01</td><td>1</td><td>2</td></tr>
<tr><td>4</td><td>3</td><td>12/3/2013</td><td>14:55:12</td><td>1</td><td>6</td><td>4.00E-01</td><td>200</td><td>4.00E+00</td><td>8.60E+01</td><td>0</td><td>3</td></tr>
<tr><td>5</td><td>4</td><td>12/3/2013</td><td>14:56:09</td><td>2</td><td>8</td><td>1.40E+00</td><td>100</td><td>7.00E+00</td><td>1.59E+02</td><td>0</td><td>4</td></tr>
<tr><td>6</td><td>5</td><td>12/3/2013</td><td>14:57:07</td><td>2</td><td>8</td><td>1.40E+00</td><td>100</td><td>7.00E+00</td><td>1.40E+02</td><td>1</td><td>5</td></tr>
<tr><td>7</td><td>6</td><td>12/3/2013</td><td>14:58:23</td><td>3</td><td>10</td><td>2.40E+00</td><td>50</td><td>8.00E+00</td><td>1.21E+02</td><td>1</td><td>6</td></tr>
<tr><td>8</td><td>7</td><td>12/3/2013</td><td>14:59:00</td><td>3</td><td>10</td><td>2.40E+00</td><td>50</td><td>8.00E+00</td><td>1.23E+02</td><td>1</td><td>7</td></tr>
<tr><td>9</td><td>8</td><td>12/3/2013</td><td>14:59:31</td><td>3</td><td>10</td><td>2.40E+00</td><td>50</td><td>8.00E+00</td><td>1.25E+02</td><td>1</td><td>8</td></tr>
<tr><td>10</td><td>9</td><td>12/3/2013</td><td>15:00:04</td><td>3</td><td>10</td><td>2.40E+00</td><td>50</td><td>8.00E+00</td><td>1.28E+02</td><td>1</td><td>9</td></tr>
<tr><td>11</td><td>10</td><td>12/3/2013</td><td>15:00:37</td><td>3</td><td>10</td><td>2.40E+00</td><td>50</td><td>8.00E+00</td><td>1.30E+02</td><td>0</td><td>10</td></tr>
<tr><td>12</td><td>//END</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table></td></tr>
</table>

The Comfort Panel has USB ports and offers you the option to save the csv file onto an USB stick.

The csv file can also be downloaded automatically at specified times (see chapter 3.5) together with SIMATIC Automation Tool as of V3.0, via “planned task” in the Windows operating system.

# 8 Links & Literature

Table 8-1

| No. | Topic |
|---|---|
| \1\ | Siemens Industry Online Support<br>https://support.industry.siemens.com |
| \2\ | Link to the entry page of the application example<br>https://support.industry.siemens.com/cs/ww/en/view/82454336 |
| \3\ | S7-1200 Automation System - System Manual<br>https://support.industry.siemens.com/cs/ww/en/view/109741593 |
| \4\ | Operating instruction SIWAREX WL200 load cells<br>https://support.industry.siemens.com/cs/ww/en/view/109749190 |
| \5\ | Firmware SIWAREX WP231<br>https://support.industry.siemens.com/cs/ww/en/view/75231231 |
| \6\ | TIA Portal program "Ready for use“ for SIWAREX WP231<br>https://support.industry.siemens.com/cs/ww/en/view/66825585 |
| \7\ | Quick Guide for Basic-Commissioning of a SIWAREX WP231<br>https://support.industry.siemens.com/cs/ww/en/view/73517989 |
| \8\ | SIWAREX WP231 electronic weighing system<br>https://support.industry.siemens.com/cs/ww/en/view/109738522 |
| \9\ | Delivery Release for SIMATIC STEP 7 Professional / Basic V14<br>https://support.industry.siemens.com/cs/ww/en/view/109740340 |
| \10\ | Delivery release SIMATIC WinCC V14<br>https://support.industry.siemens.com/cs/ww/en/view/109739719 |
| \11\ | Updates for STEP 7 V14 and WinCC V14<br>https://support.industry.siemens.com/cs/ww/en/view/109742377 |
| \12\ | Where do you find the latest operating system updates (firmware) for SIMATIC S7-1200 controllers?<br>https://support.industry.siemens.com/cs/ww/en/view/77430184 |
| \13\ | System Manual STEP 7 Basic V14.0<br>https://support.industry.siemens.com/cs/ww/en/view/109742266 |
| \14\ | Manual WinCC Advanced V14.0<br>https://support.industry.siemens.com/cs/ww/en/view/109742297 |
| \15\ | SIMATIC HMI HMI Devices Comfort Panels<br>https://support.industry.siemens.com/cs/ww/en/view/49313233 |
| \16\ | Sales and delivery release SIMATIC Automation Tool V3.0<br>https://support.industry.siemens.com/cs/ww/en/view/109749055 |

# 9 History

Table 9-1

| Version | Date | Modification |
|---|---|---|
| V1.0 | 01/2014 | First version |
| V1.1 | 09/2014 | Extension by weighing module configuration via operator panel and recipe management in the controller (CPU V4/TIA V13) |
| V2.0 | 10/2017 | Update to TIA Portal V14 |