MICROSENS

# Application Note

## microSys Demonstrator

MICROSENS GmbH & Co. KG
Kueferstr. 16
59067 Hamm/Germany

Tel. +49 2381 9452-0
FAX +49 2381 9452-100
E-Mail info@microsens.de
Web www.microsens.de

## Summary

This Quick Start Guide helps to configure and operate microRTS functionality within a MICROSENS Smart Building infrastructure. It explains installation and programming of MICROSENS microSys and upload to the MICROSENS microRTS.

The content of the document builds on information provided with the Quick Start Guide that is shipped together with MICROSENS Smart Building devices. Additionally it refers to functionality delivered by MICROSENS Smart IO Controller.

For further information about using the Web Manager for Smart Office Configuration of MICROSENS Smart Building devices please refer to the chapter “Smart Office” in [1] (see page 2).

## Glossary

Terms, acronyms and abbreviations used in the document.

| Term | Description |
|---|---|
| AHU | Air Handling Unit |
| Application | Automation task programmed with microSys |
| ENO | Enabled Output (status feedback of a function block) |
| IoT | Internet of Things |
| NESDA | Network Embedded Software Defined Automation |
| PLC | Programmable Logic Controller |
| RTS | Runtime System |

## Bibliography

[1] Product Manual Firmware Generation 6

This manual is included in each software archive containing Firmware G6. It can also be downloaded from the link “Documentation” in the firmwares’ Web Manager navigation bar.

# Table of Contents

## Typographical Conventions

The following typographical elements are used in this document:

| **Typographical Elements** | **Explanation** |
|---|---|
| ● | List element, 1st order |
| ◦ | List element, 2nd order |
| www.microsens.de | Hyperlink to a website or email address |
| **Note:** | A note tags an important fact |
| □ | Work step |
| **<…>** | Placeholder for a real value. Replace <IP Address> with e.g. 192.168.1.2. |
| **{…\|…}** | Choose one of the values offered, e.g. from {Disabled\|Enabled}, choose Enabled. |
| **Visualisation** | A label or GUI element that appears in the Web Manager |
| » `Output` | A result of an action within the Web Manager |
| ▌ | Work step(s) in the Web Manager (GUI) |

## Information available from the MICROSENS Website

Registered users can find the latest firmware and MICROSENS Smart Director app versions as well as further information on our web site:

- Registration: www.microsens.de > Partner-Login > Follow the link 'Please register here' > Fill in the online registration form and submit it
  - You will receive an email from MICROSENS with a user name and a password
- Login: www.microsens.de > Partner-Login > Enter user name and password > Click the 'Login' button
  - Firmware images: Navigate to the device and select the tab 'Services'
  - For further information select one of the other tabs

**Note:**
Make sure the browser allows the execution of scripts.

# 1 Introduction to MICROSENS microSys and microRTS

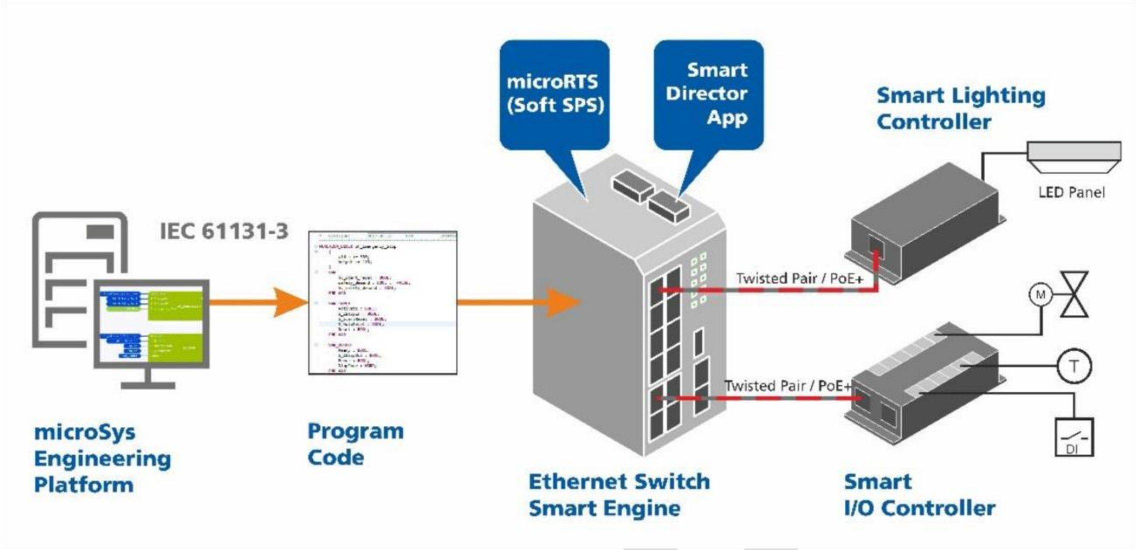


*Fig. 1: MICROSENS microSys Schematic*

MICROSENS microSys is the engineering software for creating applications for network embedded software defined automation (NESDA). Applications of all kinds can be programmed in accordance with the industry norm IEC 61131-3.

MICROSENS microSys, together with the switch based runtime system MICROSENS microRTS , enables the implementation of cost-effective and powerful automation on MICROSENS G6 network switches. This results in solutions that are precisely configured to meet network embedded software defined automation for projects in IoT and Industry 4.0.

The modern Eclipse framework allows source code and review tools to be connected and continuous integration and test environments to be seamlessly integrated.

The switch based PLC runtime system MICROSENS microRTS converts any MICROSENS G6 managed network switch into a software based automation system (Soft PLC). The I/O connection is based on MICROSENS Smart I/O Controller, Smart Light Controller and compatible bus systems (MQTT, Modbus, SNMP).

**Note:**
This document is focused on introduction to MICROSENS microSys and microRTS on MICROSENS Smart Building enabled hardware. A full training is mandatory and should not be replaced by experiencing the examples provided here.

For further information about training and qualification programs please contact your MICROSENS area sales manager.

# 2 Basic Configuration of MICROSENS Smart Office

## 2.1 Prerequisites

In order to be able to configure the Smart Buidling components, the following basic measures must be put into place:

- Administrator access rights to the specific Smart Building device are a must-have.
- Smart Director has to be enabled on G6 switch.
- All Smart Building components have to be properly connected, installed and accessible via MICROSENS NMP, MICROSENS SBM, IPConfig Tool and MICROSENS Smart Director.
- All affected devices must have installed the latest G6 firmware.

**Note:**
For more information about the respective administrator credentials and on how to access the device please refer to the device's quick start guide.

**Note:**
For more information about how to use MICROSENS NMP, MICROSENS SBM and IPConfig Tool please refer to the respective user manual.

The following descriptions assume you are using the Web Manager and logged in into the MICROSENS G6 switch, from where you want to configure and manage the Smart Building infrastructure.

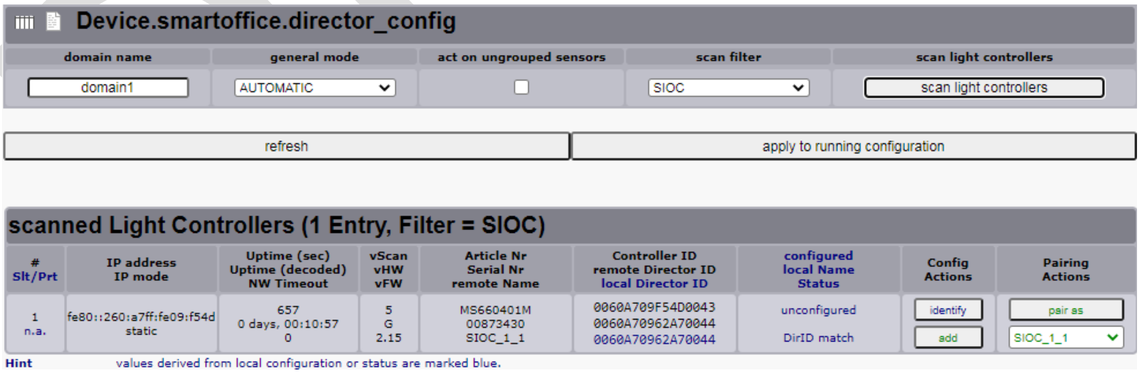
## 2.2 Pairing the MICROSENS Smart I/O Controller

All basic MICROSENS Smart Building settings of the G6 devices are pre-configured during manufacturing. What remains to be done after installation is to pair all used devices.

Using the Web Manager on the G6 switch:

□ Select the **SmartOffice** screen, then select the tab **Basic Configuration**.

□ In the section **Device.smartoffice.director_config** select "SIOC" from the scroll-down list **scan filter** and click on the button **scan light controllers**.



Device.smartoffice.director_config

| domain name | general mode | act on ungrouped sensors | scan filter | scan light controllers |
|---|---|---|---|---|
| domain1 | AUTOMATIC | □ | SIOC | scan light controllers |

scanned Light Controllers (1 Entry, Filter = SIOC)

| #<br>Slt/Prt | IP address<br>IP mode | Uptime (sec)<br>Uptime (decoded)<br>NW Timeout | vScan<br>vHW<br>vFW | Article Nr<br>Serial Nr<br>remote Name | Controller ID<br>remote Director ID<br>local Director ID | configured<br>local Name<br>Status | Config<br>Actions | Pairing<br>Actions |
|---|---|---|---|---|---|---|---|---|
| 1<br>n.a. | fe80::260:a7ff:fe09:f54d<br>static | 657<br>0 days, 00:10:57<br>0 | 5<br>G<br>2.15 | MS660401M<br>00873430<br>SIOC_1_1 | 0060A709F54D0043<br>0060A70962A70044<br>0060A70962A70044 | unconfigured<br>DirID match | identify<br>add | pair as<br>SIOC_1_1 |

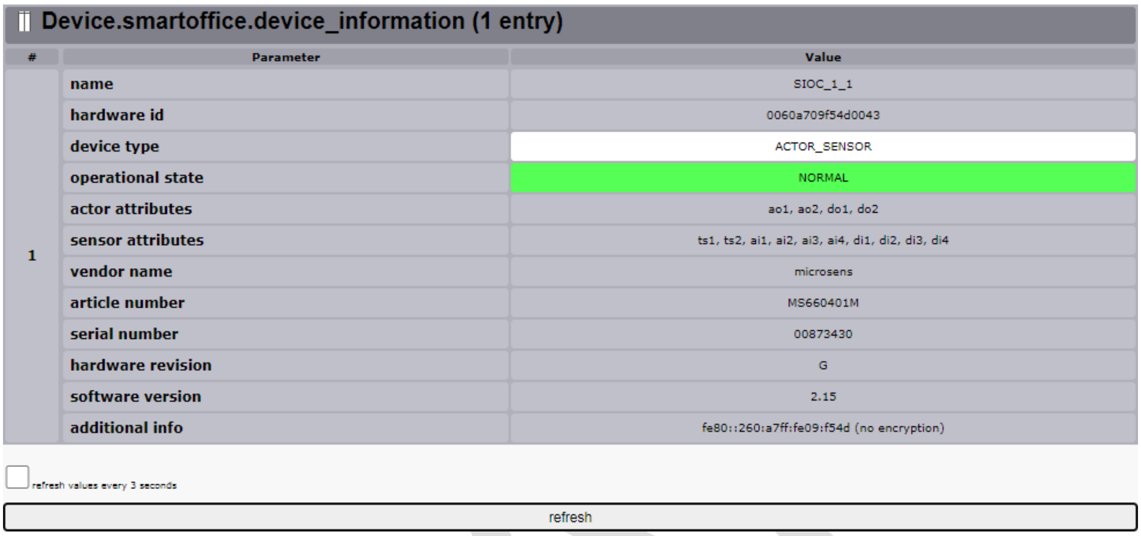
Hint values derived from local configuration or status are marked blue.

*Fig. 2: Web Manager - SmartOffice - Basic Configuration - Scan Controllers*

» After the scanning process a new entry "SIOC_1_1" for the Smart I/O Controller will appear in the section **scanned Light Controllers**.

Using the Web Manager on the G6 switch:

- □ In the column **Pairing Actions** select the template "SIOC_1_1" from the drop-down list and click on the button **pair as** to pair the Smart I/O Controller to the running configuration.
- □ Change to the tab **Device Status**.



Device.smartoffice.device_information (1 entry)

| # | Parameter | Value |
|---|---|---|
| 1 | name | SIOC_1_1 |
| | hardware id | 0060a709f54d0043 |
| | device type | ACTOR_SENSOR |
| | operational state | NORMAL |
| | actor attributes | ao1, ao2, do1, do2 |
| | sensor attributes | ts1, ts2, ai1, ai2, ai3, ai4, di1, di2, di3, di4 |
| | vendor name | microsens |
| | article number | MS660401M |
| | serial number | 00873430 |
| | hardware revision | G |
| | software version | 2.15 |
| | additional info | fe80::260:a7ff:fe09:f54d (no encryption) |

□ refresh values every 3 seconds

*Fig. 3: Web Manager - SmartOffice - Device Status - Smart I/O Controller*

» The device information should contain the typical **actor** and **sensor attributes** for this Smart I/O Controller.

» The **operational state** should be displayed as "normal".

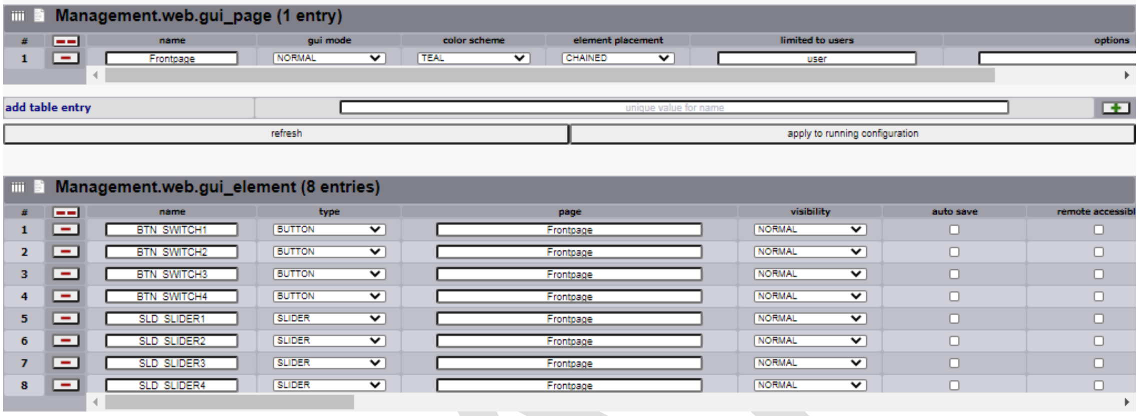
## 2.3 Configuration of the Smart Director Web GUI

MICROSENS Smart Building offers the possibility to operate the Smart Building installation via a Smart Director web GUI. The web GUI and its corresponding SPS settings are pre-configured during manufacturing.

Open the Smart Director web GUI configuration with the following steps

Using the Web Manager:

□ Select the **Userinterface** screen, then select the tab **Web**.



**Management.web.gui_page (1 entry)**

| # | | name | gui mode | color scheme | element placement | limited to users | options |
|---|---|---|---|---|---|---|---|
| 1 | | Frontpage | NORMAL | TEAL | CHAINED | user | |

add table entry — unique value for name

refresh | apply to running configuration

**Management.web.gui_element (8 entries)**

| # | | name | type | page | visibility | auto save | remote accessibl |
|---|---|---|---|---|---|---|---|
| 1 | | BTN SWITCH1 | BUTTON | Frontpage | NORMAL | □ | □ |
| 2 | | BTN SWITCH2 | BUTTON | Frontpage | NORMAL | □ | □ |
| 3 | | BTN SWITCH3 | BUTTON | Frontpage | NORMAL | □ | □ |
| 4 | | BTN SWITCH4 | BUTTON | Frontpage | NORMAL | □ | □ |
| 5 | | SLD SLIDER1 | SLIDER | Frontpage | NORMAL | □ | □ |
| 6 | | SLD SLIDER2 | SLIDER | Frontpage | NORMAL | □ | □ |
| 7 | | SLD SLIDER3 | SLIDER | Frontpage | NORMAL | □ | □ |
| 8 | | SLD SLIDER4 | SLIDER | Frontpage | NORMAL | □ | □ |

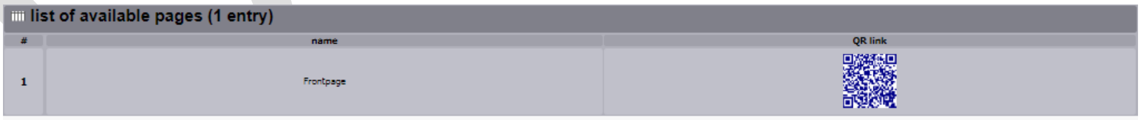
*Fig. 4: Web Manager - Userinterface - Web*

» The section **Management.web.gui_page** contains a webpage named "Frontpage". This is the HTML page containing all SPS related GUI elements.

» The section **Management.web.gui_element** contains a **name** list of buttons ("BTN Switchx") and sliders ("SLD Sliderx") assigned to the **page** "Frontpage".

**Note:**
Do not change these settings as they correspond to the pre-configured SPS settings. Any improper alteration will cause the infrastructure to stop working properly.

□ The section **list of available pages** shows a QR code containing the URL address of the respective web GUI page.



**list of available pages (1 entry)**

| # | name | QR link |
|---|---|---|
| 1 | Frontpage | |

*Fig. 5: Web Manager - Userinterface - Web - Page List*

» To open this page click on the QR code.

» Alternatively enter the address `https://<device IP address>/gui/Frontpage.html` into the browser address bar.

Using the Web Manager:

- □ The following page opens in a new browser window:

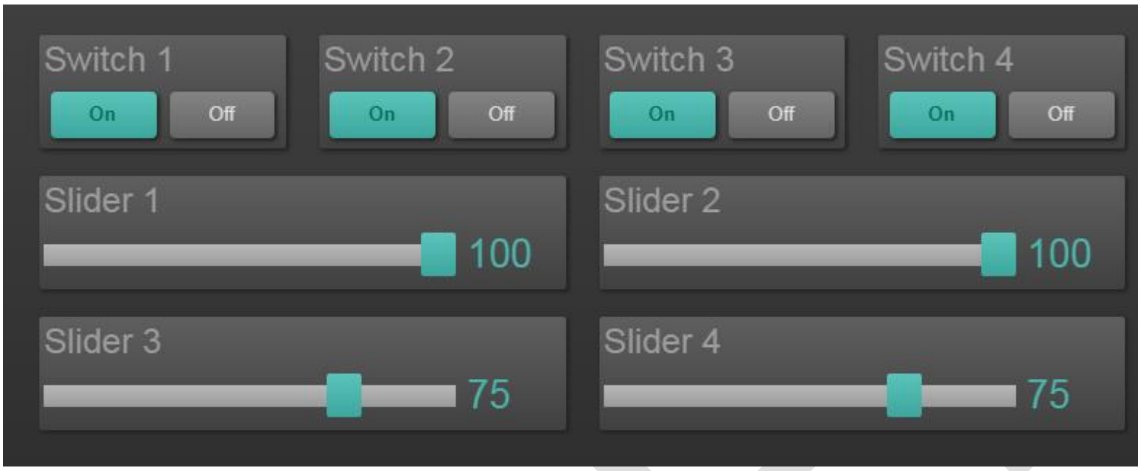


*Fig. 6: Smart Director Web GUI*

» The window shows all elements (buttons and sliders) defined on the previously described section **Management.web.gui_element**.

**Note:**
Due to its implementation as web server providing a simple web page all clients in the same network are able to access the Smart Director web GUI via web browser. No additional plugins like Java or the like are necessary.

# 3 Using MICROSENS microSys

## 3.1 System Requirements

- Operating system:
    - Microsoft® Windows® 7 (64-Bit)
    - Microsoft® Windows® 8.1 (64-Bit)
    - Microsoft® Windows® 10 (64-Bit)

    **Note:**
    Microsoft® Windows® 32-bit operating systems are not supported!
- Hardware:
  Generally, the identical hardware requirements apply as for the operating system in use. Additionally, the following minimum requirements apply:
    - Memory: min. 8 GB RAM
    - Hard disk: min. 4 GB free space
- General system recommendation:
    - Memory: 16 GB RAM
    - I7- CPU with min. 3.4 GHz (or similar)
    - Hard disk: SSD
- Licence dongle containing a valid MICROSENS microSys licence plugged into a USB port of the workstation

**Note:**
A plugged-in and valid licence dongle is recommended for using microSys to the full extent, whereby without dongle it is possible to use microSys with reduced functionality (i.e. using PLC structured text, but neither programming nor viewing of PLC function blocks).

## 3.2 Installation

Copy the file **`microsys.zip`** from the supplied USB flash drive to a temporary location on your hard disk drive. The zip file contains all necessary directories and files. It is recommended to unpack the zip file to the drive **`C:\`**.

**Note:**
When using a different installation path make sure to avoid special characters or blanks with the path name. Otherwise microSys will not be able to load the application onto the software PLC of the G6 switch. The following characters are allowed for paths and project names:

- A - Z
- a - z
- 0 - 9
- dot (".")
- underscore ("_")
- dash ("-")

## 3.3 Creating microSys Workspace Directory

microSys uses a directory **`\workspace`** for editing projects. This directory has to be created manually before executing microSys the first time. Due to access permissions it is recommended to create this directory within the user folder **`C:\Users\...\...\workspace`**.

**Note:**
It is possible to change this path in the microSys settings.

## 3.4 Starting microSys and Importing Demo Project

Click on the file **`microsys.exe`** to start microSys with the project that was edited before closing microSys the last time.

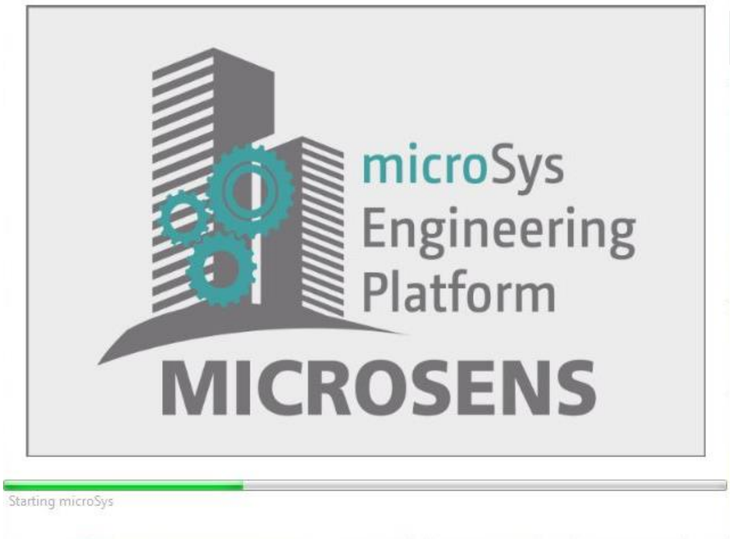


*Fig. 7: microSys Startup Screen*

Depending on how you closed microSys the last time, the project explorer is either empty or shows the last active project.

To open an existing project click on **File > Open Projects from File System** in the microSys menu bar.

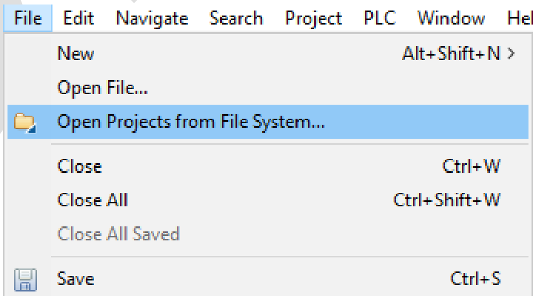
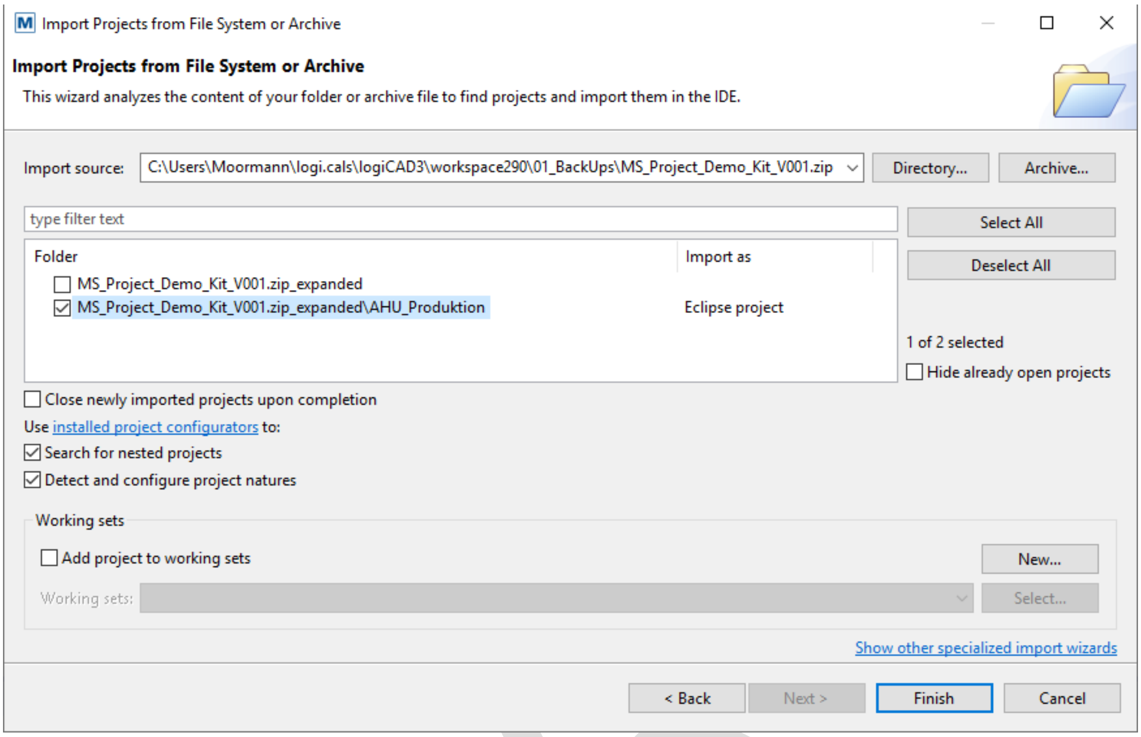


*Fig. 8: microSys - File - Open Projects*

**Note:**
This Application Note focuses on using an existing demo project with microSys. So developing a new project from scratch is part of an individual MICROSENS workshop. Contact your sales representative for more information on specific workshops.

In the opening dialogue it is possible to either open an active or an archived and compressed project from the file system:


Import Projects from File System or Archive

**Import Projects from File System or Archive**

This wizard analyzes the content of your folder or archive file to find projects and import them in the IDE.

Import source: C:\Users\Moormann\logi.cals\logiCAD3\workspace290\01_BackUps\MS_Project_Demo_Kit_V001.zip

| Folder | Import as |
|---|---|
| □ MS_Project_Demo_Kit_V001.zip_expanded | |
| ☑ MS_Project_Demo_Kit_V001.zip_expanded\AHU_Produktion | Eclipse project |

1 of 2 selected

□ Hide already open projects

□ Close newly imported projects upon completion

Use installed project configurators to:

☑ Search for nested projects

☑ Detect and configure project natures

Working sets

□ Add project to working sets

*Fig. 9: microSys - Import Project Wizard*

- » Click on the button **Directory...** to navigate to an active project.
- » Click on the button **Archive...** to navigate to an archived project.

After selecting the respective directory the contained project data folder is shown in the list.

**Note:**
The folder list of an archived project additionally shows the temporary folder containing the unzipped project data marked with the suffix **`_expanded`**. It is recommended to uncheck this entry because otherwise the zip file unnecessarily appears in the project explorer, too.

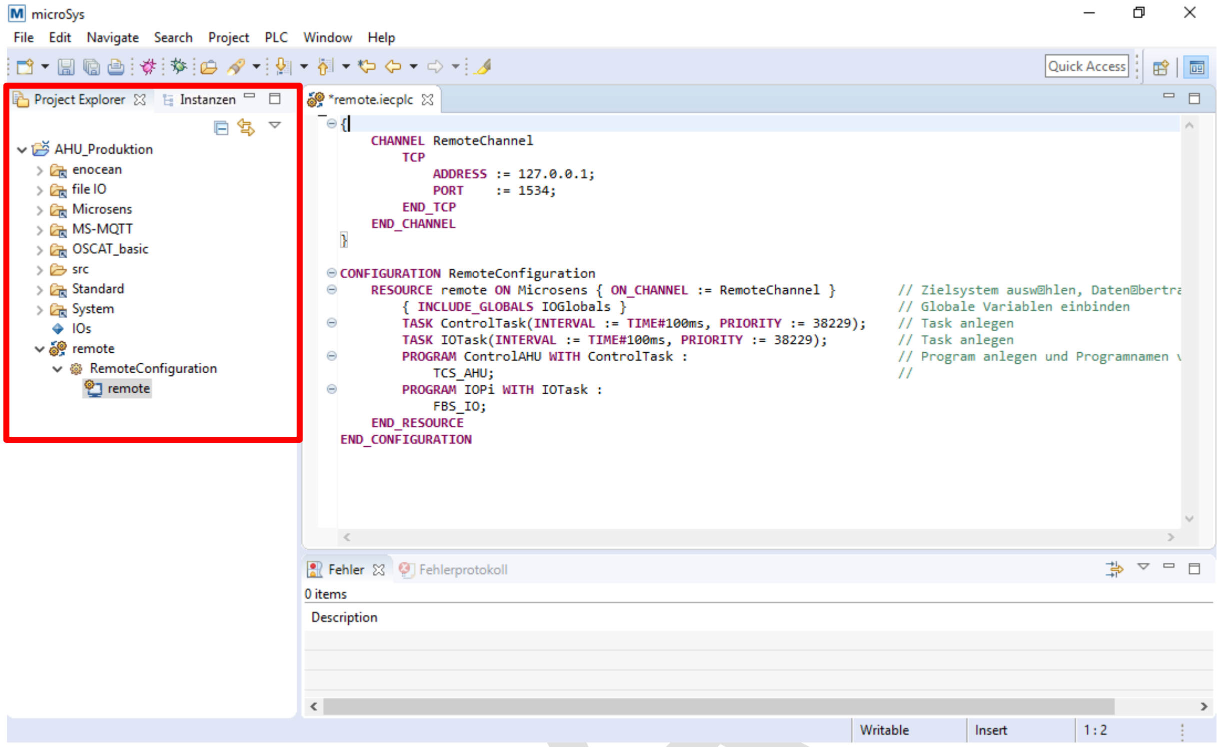
By default the list contains all recognised projects in the selected directory or archive, regardless of whether a project already is opened in the workplace. Check the option **Hide already open projects** to suppress the respective entry in the list.

## 3.5 Configure Project Resources

Project ressources are considered to be function libraries maintained by MICROSENS or third party vendors.

**Note:**
After opening a project microSys compares the present project resources with the resources it was previously compiled with. If there is any discrepancy of library versions microSys issues an error message. To be able to carry on working with the project it is mandatory to save the latest project resources version into the library directory and re-compile the project.

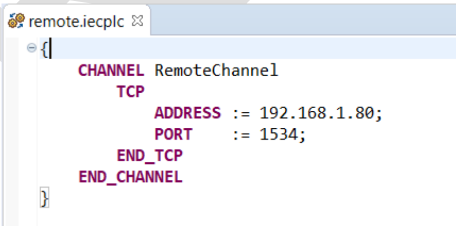
To change the settings of a resource expand the respective project folder in the **Project Explorer** tab on the left-hand pane and select the relevant resource.



```
{
    CHANNEL RemoteChannel
        TCP
            ADDRESS := 127.0.0.1;
            PORT    := 1534;
        END_TCP
    END_CHANNEL
}

CONFIGURATION RemoteConfiguration
    RESOURCE remote ON Microsens { ON_CHANNEL := RemoteChannel }        // Zielsystem ausw�hlen, Daten�bertra
        { INCLUDE_GLOBALS IOGlobals }                                   // Globale Variablen einbinden
        TASK ControlTask(INTERVAL := TIME#100ms, PRIORITY := 38229);    // Task anlegen
        TASK IOTask(INTERVAL := TIME#100ms, PRIORITY := 38229);         // Task anlegen
        PROGRAM ControlAHU WITH ControlTask :                           // Program anlegen und Programnamen v
            TCS_AHU;                                                    //
        PROGRAM IOPi WITH IOTask :
            FBS_IO;
    END_RESOURCE
END_CONFIGURATION
```

*Fig. 10: microSys - Configure Resources*

For this demo project the connection information has to be adapted so that later on microSys is able to provide the program code to the correct resource. Double-click on the resource “remote” in the project path **remote > RemoteConfiguration**. The editor window on the right-hand pane shows the respective resource’s program code.



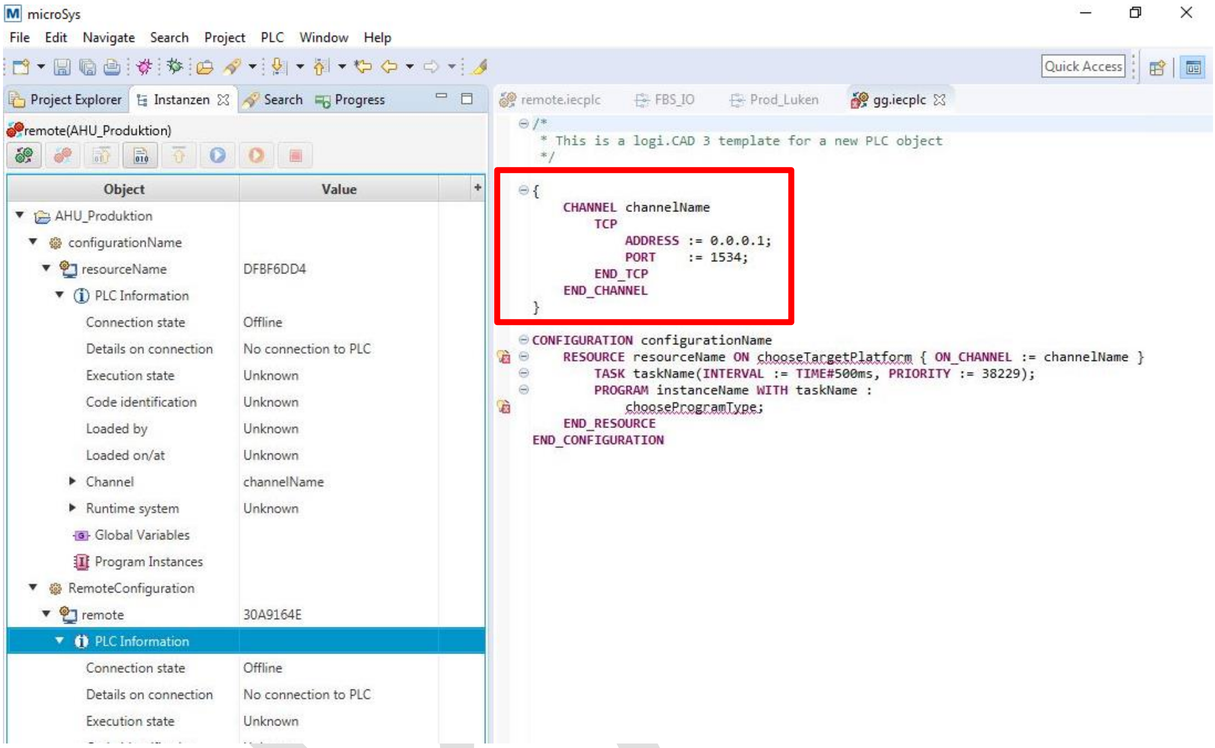
```
{
    CHANNEL RemoteChannel
        TCP
            ADDRESS := 192.168.1.80;
            PORT    := 1534;
        END_TCP
    END_CHANNEL
}
```

*Fig. 11: microSys - Resource Editor*

**Note:**
If it should be necessary to use a port number that differs from the default value additionally change the port number in the settings of both the microRTS and the respective G6 switch. Otherwise microSys will not be able to reach the microRTS on the switch.

If several microRTS in the corporate network are available through the engineering platform it is defined here which one of these resources should receive the program code. Therefore the IP address of the MICROSENS switch must be set in its respective channel.

Select the **Instanzen** tab and navigate to **<Project name> > RemoteConfiguration > remote > PLC Information**. In the editor window on the right-hand pane enter the IP address.



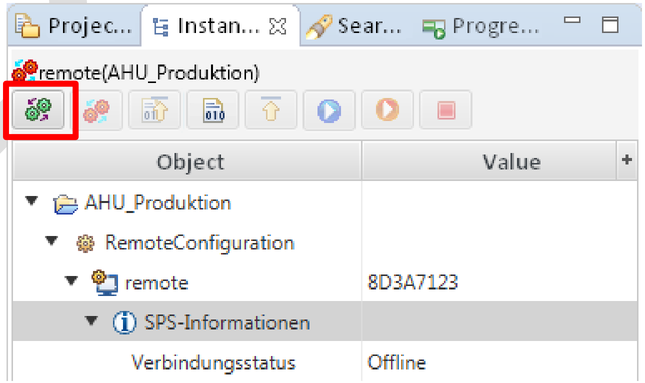
| Object | Value |
|---|---|
| AHU_Produktion | |
| configurationName | |
| resourceName | DFBF6DD4 |
| PLC Information | |
| Connection state | Offline |
| Details on connection | No connection to PLC |
| Execution state | Unknown |
| Code identification | Unknown |
| Loaded by | Unknown |
| Loaded on/at | Unknown |
| Channel | channelName |
| Runtime system | Unknown |
| Global Variables | |
| Program Instances | |
| RemoteConfiguration | |
| remote | 30A9164E |
| PLC Information | |
| Connection state | Offline |
| Details on connection | No connection to PLC |
| Execution state | Unknown |

```
/*
 * This is a logi.CAD 3 template for a new PLC object
 */

{
    CHANNEL channelName
        TCP
            ADDRESS := 0.0.0.1;
            PORT    := 1534;
        END_TCP
    END_CHANNEL
}

CONFIGURATION configurationName
    RESOURCE resourceName ON chooseTargetPlatform { ON_CHANNEL := channelName }
        TASK taskName(INTERVAL := TIME#500ms, PRIORITY := 38229);
        PROGRAM instanceName WITH taskName :
            chooseProgramType;
    END_RESOURCE
END_CONFIGURATION
```

*Fig. 12: microSys - Channel Configuration*

## 3.6 Establish Connection to microRTS

To connect microSys to a specific microRTS select the respective project on the tab **Project Explorer** on the left-hand pane. Change to the tab **Instanzen** and navigate to **RemoteConfiguration > remote > SPS_Information**.



| Object | Value |
|---|---|
| AHU_Produktion | |
| RemoteConfiguration | |
| remote | 8D3A7123 |
| SPS-Informationen | |
| Verbindungsstatus | Offline |

*Fig. 13: microSys - Instanzen - Connect to microRTS*

The trailing **red gears** in front of the resource name indicate that microSys and microRTS are not connected.

Click on the button **Connect** (green gears) to establish a connection to the microRTS on the G6 switch.

After successfully connected to microRTS the trailing gears change to green.

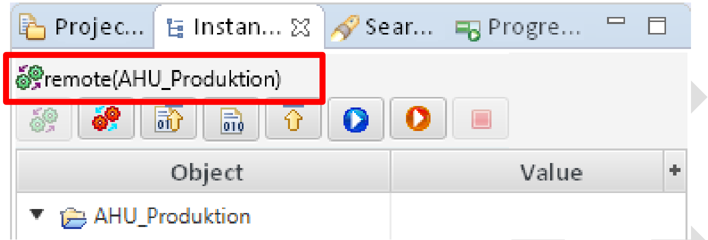


*Fig. 14: microSys - Instanzen - Connection established*

**Note:**
To disconnect from microRTS click on the button **Disconnect** (red gears).

As long as the program code is not compiled and loaded into the microRTS the status of the microRTS is set to "stopped".

## 3.7 Compiling PLC Application

Before loading the application into the microRTS it is necessary to compile the program code with microSys.

**Note:**
Compiling the code is possible without a network connection between microSys and the MICROSENS sG6 switch.

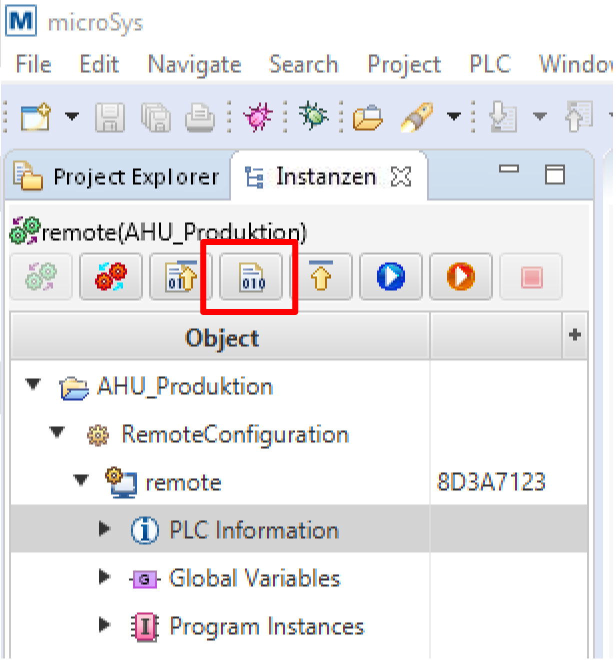


*Fig. 15: microSys - Instanzen - Build Application*

To compile the program code select the tab **Instanzen** on the left-hand pane, navigate the path **<Project name> > Remote Configuration > remote** and select the entry **PLC Information** with a double-click to show the program code in the editor pane.

With the respective entry selected the button **Build application** becomes active. Click the button to start the compiling process.

Errors the microSys interpreter does not recognise are displayed in the **Error** pane and in the view **Error Log** (e.g. if a global variable is not defined or the variable definition does not match the reference in the program code).

**Note:**
With every change of the program code it is necessary to compile the code again before loading it to the microRTS.

## 3.8 Loading Application to microRTS

**Note:**
To load an application to the microRTS a network connection is necessary. When not connected the button **Load Application** is disabled.

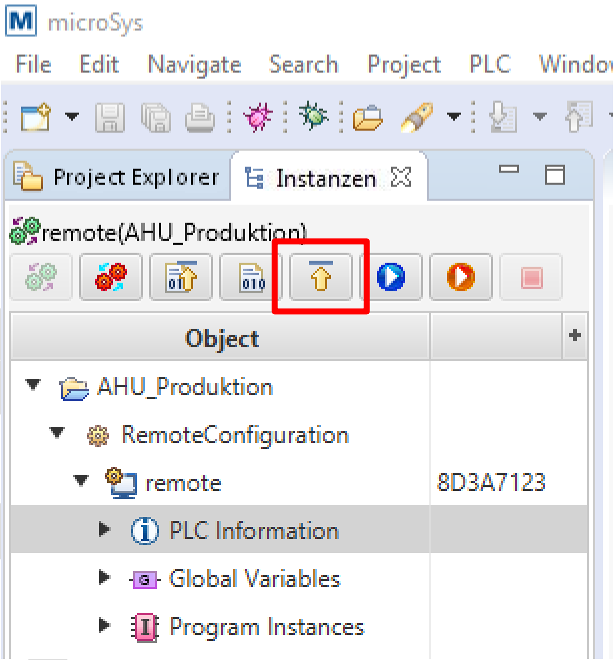


*Fig. 16: microSys - Instanzen - Load Application*

To load the program to a specific microRTS select the tab **Instanzen** on the left-hand pane, navigate the path **<Project name> > Remote Configuration > remote** and select the entry **PLC Information** with a double-click to show the program code in the editor pane.

With the respective entry selected and the associated program previously compiled the button **Load application** becomes active. Click the button to start the loading process. After successfully loading the application the microrRTS restarts and changes its status to "running".

**Note:**
It is possible to build and load an application automatically in one step with the button **Build and Load Application** as long as a network connection is established.

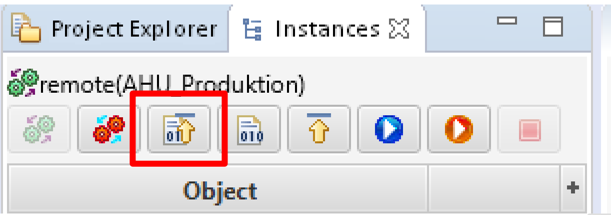


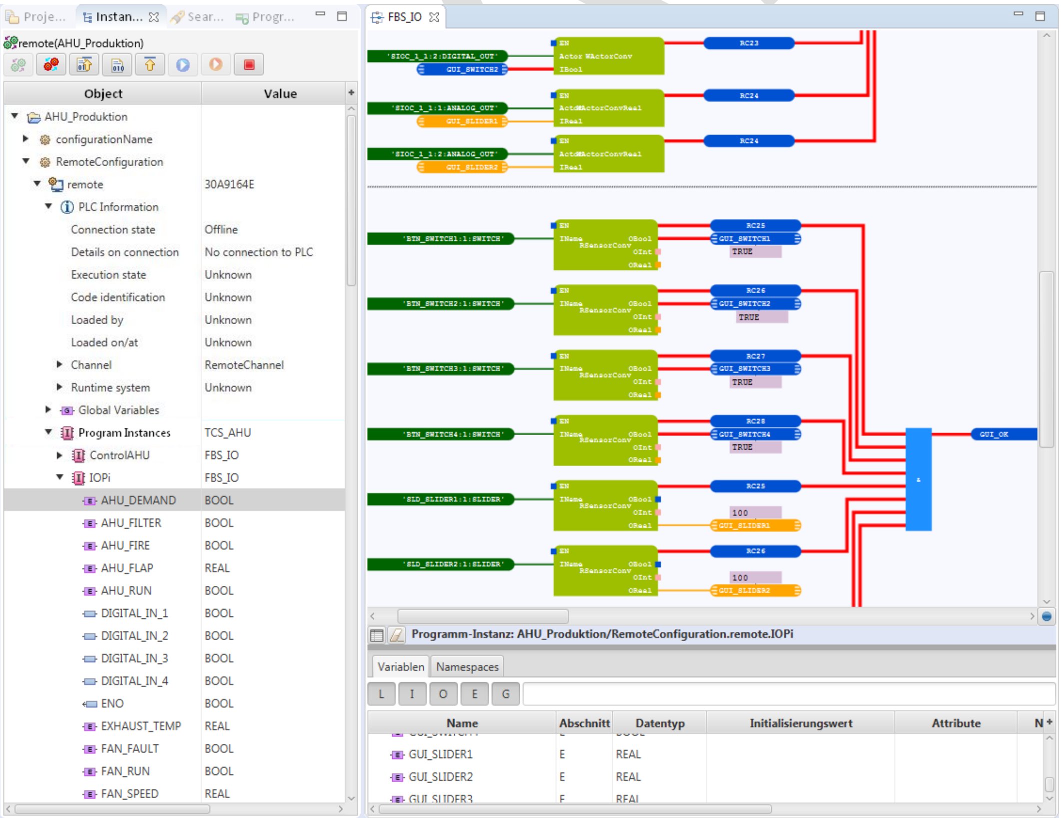
*Fig. 17: microSys - Instances - Build and Load Application*

With a click on this button microSys compiles the program code and loads the compiled code to microRTS. microRTS will restart even when errors occur during the compiling process, using the previous version of the application without the latest changes.

## 3.9 Online Testing of Applications

To see the program code in action navigate to the folder **Program Instances** , open the folder **IOPI** and select the first variable with a double-click to show the online test in the right-hand pane.

The program visualisation consists of function blocks which can represents function predefined or defined along the project. It reads the external device attributes and assigns the values to internal variables.



remote(AHU_Produktion)

| Object | Value |
|---|---|
| AHU_Produktion | |
| configurationName | |
| RemoteConfiguration | |
| remote | 30A9164E |
| PLC Information | |
| Connection state | Offline |
| Details on connection | No connection to PLC |
| Execution state | Unknown |
| Code identification | Unknown |
| Loaded by | Unknown |
| Loaded on/at | Unknown |
| Channel | RemoteChannel |
| Runtime system | Unknown |
| Global Variables | |
| Program Instances | TCS_AHU |
| ControlAHU | FBS_IO |
| IOPi | FBS_IO |
| AHU_DEMAND | BOOL |
| AHU_FILTER | BOOL |
| AHU_FIRE | BOOL |
| AHU_FLAP | REAL |
| AHU_RUN | BOOL |
| DIGITAL_IN_1 | BOOL |
| DIGITAL_IN_2 | BOOL |
| DIGITAL_IN_3 | BOOL |
| DIGITAL_IN_4 | BOOL |
| ENO | BOOL |
| EXHAUST_TEMP | REAL |
| FAN_FAULT | BOOL |
| FAN_RUN | BOOL |
| FAN_SPEED | REAL |

Programm-Instanz: AHU_Produktion/RemoteConfiguration.remote.IOPi

| Name | Abschnitt | Datentyp | Initialisierungswert | Attribute | N |
|---|---|---|---|---|---|
| GUI_SLIDER1 | E | REAL | | | |
| GUI_SLIDER2 | E | REAL | | | |
| GUI_SLIDER3 | E | REAL | | | |

*Fig. 18: microSys - Instanzen - Online Test Visualisation*

In the Smart Director web GUI (see section 0 on page 8) click on one of the given buttons or sliders. This should instantly result in the update of the respective values shown by the microSys online test.

The meaning of the values and information shown in the online test pane is explained using the function "RSensorConv". This function is defined along the project to reduce repeating tasks while programming like the conversion of datatypes from String to Boolean.

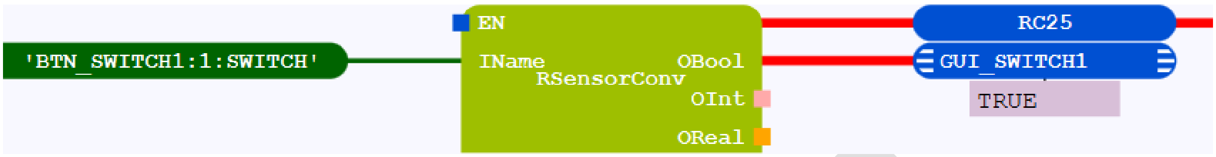


*Fig. 19: microSys - Online Test Information- Example "RSensorConv"*

- **BTN_SWITCH1:1:SWITCH:** This string variable corresponds to the sensor element assigned in the section **SmartOffice.SensorStatus.Sensor_Status_List** of the G6 switch's Web Manager.
- **RSensorConv:** This function block reads a variable name as string and responds with a Boolean value "high" if the sensor exists on the enabled output (ENO).
    - **OBool** responds the value of the sensor as Boolean value.
    - **OInt** responds the value of the sensor as Integer value.
    - **OReal** responds the value of the sensor as Real value
- **RC25:** This variable is connected to the ENO of the function block.
- **GUI_SWITCH:** This variable is connected to the **OBool** output.

  The different indicator design shows that this variable is used as a global variable, that can be used to transfer information between different tasks.

Double-click on the function "RSensConv" shows the internal processes of this function.

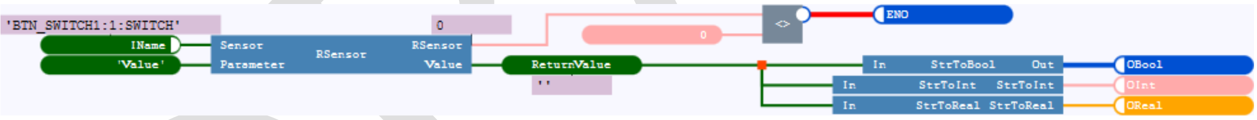
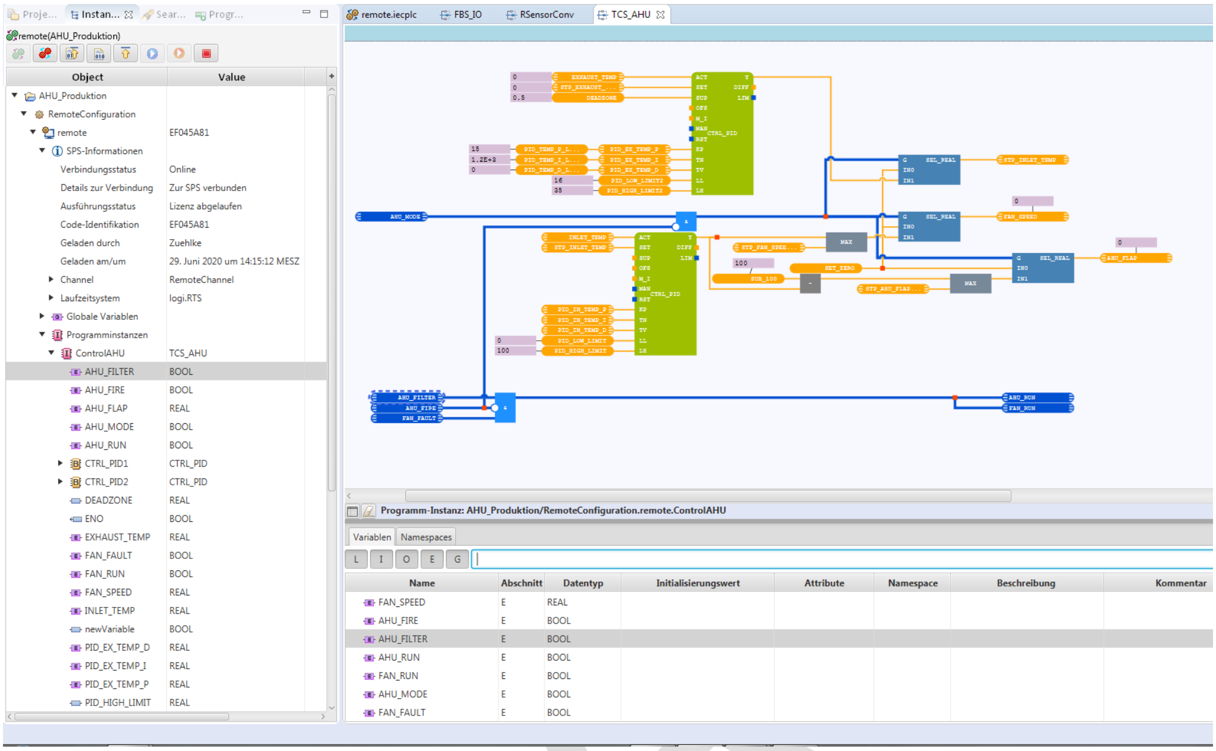


*Fig. 20: microSys - Online Test Information - Function Internals*

This versatile predefined function addresses a specific read datapoint (here: 'Value' as STRING). The block element "RSensor" returns the measured 'ReturnValue' as STRING, which is afterwords transformed into one of the three common data types BOOL, INT and REAL. At the same time "RSensor" writes the result to the ENO of the function block for further use.

For a visualisation of a more complex program open the folder **ControlAHU** and select the first variable "AHU-Filter" with a double-click. The program opens in the online test editor.



| Object | Value |
|---|---|
| AHU_Produktion | |
| RemoteConfiguration | |
| remote | EF045A81 |
| SPS-Informationen | |
| Verbindungsstatus | Online |
| Details zur Verbindung | Zur SPS verbunden |
| Ausführungsstatus | Lizenz abgelaufen |
| Code-Identifikation | EF045A81 |
| Geladen durch | Zuehlke |
| Geladen am/um | 29. Juni 2020 um 14:15:12 MESZ |
| Channel | RemoteChannel |
| Laufzeitsystem | logi.RTS |
| Globale Variablen | |
| Programminstanzen | |
| ControlAHU | TCS_AHU |
| AHU_FILTER | BOOL |
| AHU_FIRE | BOOL |
| AHU_FLAP | REAL |
| AHU_MODE | BOOL |
| AHU_RUN | BOOL |
| CTRL_PID1 | CTRL_PID |
| CTRL_PID2 | CTRL_PID |
| DEADZONE | REAL |
| ENO | BOOL |
| EXHAUST_TEMP | REAL |
| FAN_FAULT | BOOL |
| FAN_RUN | BOOL |
| FAN_SPEED | REAL |
| INLET_TEMP | REAL |
| newVariable | BOOL |
| PID_EX_TEMP_D | REAL |
| PID_EX_TEMP_I | REAL |
| PID_EX_TEMP_P | REAL |
| PID_HIGH_LIMIT | REAL |

Programm-Instanz: AHU_Produktion/RemoteConfiguration.remote.ControlAHU

| Name | Abschnitt | Datentyp | Initialisierungswert | Attribute | Namespace | Beschreibung | Kommentar |
|---|---|---|---|---|---|---|---|
| FAN_SPEED | E | REAL | | | | | |
| AHU_FIRE | E | BOOL | | | | | |
| AHU_FILTER | E | BOOL | | | | | |
| AHU_RUN | E | BOOL | | | | | |
| FAN_RUN | E | BOOL | | | | | |
| AHU_MODE | E | BOOL | | | | | |
| FAN_FAULT | E | BOOL | | | | | |

*Fig. 21: microSys - Online Test - Complex Program*

This program shows an air handling unit (AHU) in its simplest implementation:

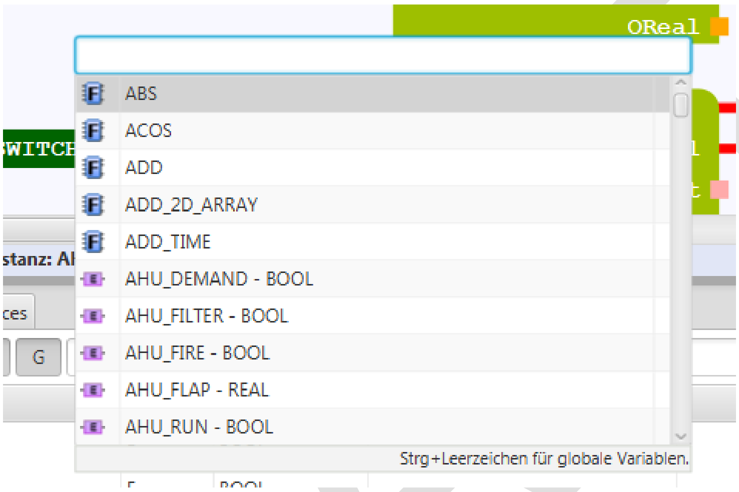
Just the external comand "Go" and a "high" signal on fireprotection are essential to start the ventilator relais. As soon as the ventilator is running and returns a "high" signal, both loops are set on running.

The two loops are in serial, the exhaust air loop calculates the inlet air setpoint for the inlet temp loop. The resulting signal for valve and flap can be minimized for convenience. When the start parameter is not true, the signals are set to 0%.

## 3.10 Using Variables

As previously mentioned the different designs of the variable indicators show whether the respective variable act as local (used only for a specific task) or global (usable for different tasks of the project).

- Global variables have to be defined separately within the global variables file in the project folder "IOs".
- Local variables are defined in the variable editor. Open the variable editor by using the keys **`Ctrl`** + **`Space`**. This will open a variable list where existing variables can be selected or new variables can be created.



Strg+Leerzeichen für globale Variablen.

*Fig. 22: microSys - Variable Editor*

E.g., the entry "TestVAR:BOOL" will create a new local Boolean variable named "TestVAR", that can be placed freely in the editor but is valid only for the assigned task.

**Note:**
To enter a string variable its length has to be attended like: "TestString:STRING[16]". This will create a local string variable with the length of 16 caracters.

## Disclaimer

All information in this document is provided ‘as is’ and subject to change without notice.

MICROSENS GmbH & Co. KG disclaims any liability for the correctness, completeness or quality of the information provided, fitness for a particular purpose or consecutive damage.

Any product names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

Document ID: AN-20xxx_2020_xx_xx