WAGO-I/O-SYSTEM 750
WagoAppEnocean
Connecting EnOcean Sensors / Actuators
Version 1.0.2
Every conceivable measure has been taken to ensure the accuracy and completeness of this documentation. However, as errors can never be fully excluded, we always appreciate any information or suggestions for improving the documentation.

We wish to point out that the software and hardware names, as well as the trademarks of companies used and/or mentioned in the present document, are generally protected by trademark or patent.
Information about This Documentation

Copyright

This documentation, including all figures and illustrations contained therein, is subject to copyright protection. Any use of this documentation that infringes upon the copyright provisions stipulated herein is prohibited. Reproduction, translation, electronic, photo-technical filing/archiving and any amendments require the written consent of WAGO Kontakttechnik GmbH & Co. KG, Minden, Germany. Non-observance will entail the right of claims for damages. WAGO is a registered trademark of WAGO Verwaltungsgesellschaft mbH.

Number Notation

Table 1: Number Notation

<table>
<thead>
<tr>
<th>Number System</th>
<th>Example</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decimal</td>
<td>100</td>
<td>Normal notation</td>
</tr>
<tr>
<td>Hexadecimal</td>
<td>0x64</td>
<td>C notation</td>
</tr>
<tr>
<td>Binary</td>
<td>'100'</td>
<td>In single quotes, nibble separated by a period</td>
</tr>
<tr>
<td></td>
<td>'0110.0100'</td>
<td></td>
</tr>
</tbody>
</table>

Font Conventions

Table 2: Font Conventions

<table>
<thead>
<tr>
<th>Font Type</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>italic</td>
<td>Names of paths and files are shown in italics, e.g.: C:\Programs\WAGO-I/O-CHECK</td>
</tr>
<tr>
<td>Menu</td>
<td>Menu options are shown in bold, e.g.: Save</td>
</tr>
<tr>
<td>&gt;</td>
<td>A “greater than” symbol between two names denotes the selection of a menu option, e.g.: File &gt; New</td>
</tr>
<tr>
<td>Input</td>
<td>Names of input or selection fields are shown in bold, e.g.: Start of measurement range</td>
</tr>
<tr>
<td>“Value”</td>
<td>Input or selection values are shown in quotation marks, e.g.: Enter the value “4 mA” under Start of measurement range.</td>
</tr>
<tr>
<td>[Button]</td>
<td>Button labels within the dialogs are bold and enclosed in square brackets, e.g.: [Input]</td>
</tr>
<tr>
<td>[Key]</td>
<td>Key labels on the keyboard are shown in bold and enclosed in square brackets, e.g.: [F5]</td>
</tr>
</tbody>
</table>
# Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="warning.png" alt="WARNING)" /></td>
<td><strong>Warning against personal injury!</strong> Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.</td>
</tr>
<tr>
<td><img src="warning.png" alt="WARNING)" /></td>
<td><strong>Do not work on components while energized!</strong> Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.</td>
</tr>
<tr>
<td><img src="warning.png" alt="WARNING)" /></td>
<td><strong>Warning against personal injury!</strong> Indicates a moderate-risk, potentially hazardous situation which, if not avoided, could result in death or serious injury.</td>
</tr>
<tr>
<td><img src="caution.png" alt="CAUTION)" /></td>
<td><strong>Warning against personal injury!</strong> Indicates a low-risk, potentially hazardous situation which, if not avoided, may result in minor or moderate injury.</td>
</tr>
<tr>
<td><img src="attention.png" alt="ATTENTION)" /></td>
<td><strong>Warning against damage to property!</strong> Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.</td>
</tr>
<tr>
<td><img src="electrostatic_discharge.png" alt="ESD" /></td>
<td><strong>Warning against damage to property caused by electrostatic discharge!</strong> Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.</td>
</tr>
<tr>
<td><img src="note.png" alt="Note" /></td>
<td><strong>Important note!</strong> Indicates a potential malfunction, but one which will not result in damage to property if not avoided.</td>
</tr>
<tr>
<td><img src="information.png" alt="Information" /></td>
<td><strong>Additional information</strong> Refers to additional information which is not an integral part of this documentation (e.g., the Internet).</td>
</tr>
</tbody>
</table>
Legal Principles

Subject to Change
WAGO Kontakttechnik GmbH & Co. KG reserves the right to make any alterations or modifications that serve the purpose of technical progress. WAGO Kontakttechnik GmbH & Co. KG owns all rights arising from granting patents or from the legal protection of utility patents. Third-party products are always mentioned without any reference to patent rights. Thus, the existence of such rights cannot be excluded.

Personnel Qualification
The use of the product described in this document is exclusively geared to specialists having qualifications in PLC programming, electrical specialists or persons instructed by electrical specialists who are also familiar with the appropriate current standards.

Moreover, the persons named here must also be familiar with all of the products cited in this document, along with the operating instructions. They must also be capable of correctly predicting any hazards which may not arise until the products are combined.

WAGO Kontakttechnik GmbH & Co. KG assumes no liability resulting from improper action and damage to WAGO products and third-party products due to non-observance of the information contained in this document.

Limitation of Liability
This documentation describes the use of various hardware and software components in specific example applications. The components may represent products or parts of products from different manufacturers. The respective operating instructions from the manufacturers apply exclusively with regard to intended and safe use of the products. The manufacturers of the respective products are solely responsible for the contents of these instructions.

The sample applications described in this documentation represent concepts, that is, technically feasible applications. Whether these concepts can actually be implemented depends on various general conditions. For example, different versions of the hardware or software components may require different handling than that described here. Therefore, the descriptions contained in this documentation do not form the basis for assertion of a particular product characteristic.

Responsibility for safe use of a specific software or hardware configuration lies with the party that produces or operates the configuration. This also applies if one of the concepts described in this document was used for implementation of the configuration.

WAGO Kontakttechnik GmbH & Co. KG assumes no liability for the realization of these concepts.
Table of Contents

Table of Contents .......................................................................................................................... 6
1 Description ................................................................................................................................. 7
2 Material Used ............................................................................................................................ 7
  2.1 Required Libraries .............................................................................................................. 7
  2.2 Devices ............................................................................................................................... 7
  2.3 Tools .................................................................................................................................. 7
3 Setup ......................................................................................................................................... 8
4 EnOcean Equipment Profile (EEP) ......................................................................................... 9
  4.1 Structure of an EEP ............................................................................................................ 9
5 Sample Program .................................................................................................................... 10
  5.1 Tasks .................................................................................................................................. 10
  5.2 Programming ..................................................................................................................... 11
  5.3 Determining the ID Number .............................................................................................. 13
    5.3.1 Example: SR04 (Thermokon) .................................................................................... 14
  5.4 Determining the Suitable EEP Function Block ................................................................. 15
  5.5 Receive Telegram ............................................................................................................... 16
    5.5.1 Evaluating the Signal of a Room Operating Panel .................................................... 16
    5.5.2 Evaluating the Signal of a Touch Sensor ................................................................. 17
    5.5.3 Evaluating the Signal of a Window Contact .......................................................... 18
    5.5.4 Receive Raw Data ..................................................................................................... 19
  5.6 Transmit Telegram ............................................................................................................ 20
    5.6.1 Virtual EnOcean Button in CODESYS ................................................................. 20
    5.6.2 Transmit Data Byte ................................................................................................. 21
  5.7 Bidirectional Communication ............................................................................................. 22
    5.7.1 Bidirectional Function Block .................................................................................... 22
1 Description

This application note describes how communication between the WAGO-I/O-SYSTEM and different EnOcean wireless sensors can be implemented using the WagoAppEnocean library.

2 Material Used

2.1 Required Libraries

<table>
<thead>
<tr>
<th>Library</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WagoAppEnocean</td>
<td>Library with different function blocks for connecting EnOcean wireless sensors / actuators</td>
</tr>
</tbody>
</table>

2.2 Devices

<table>
<thead>
<tr>
<th>Supplier</th>
<th>Quantity</th>
<th>Designation</th>
<th>Item No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>WAGO</td>
<td>1</td>
<td>PFC100 Controller; 2 x ETHERNET</td>
<td>750-8101</td>
</tr>
<tr>
<td>WAGO</td>
<td>1</td>
<td>RS-232/-485 serial interface</td>
<td>750-652</td>
</tr>
<tr>
<td>WAGO</td>
<td>1</td>
<td>End module</td>
<td>750-600</td>
</tr>
<tr>
<td>WAGO</td>
<td>1</td>
<td>EnOcean-receiver/sender with RS-485 interface STC65-RS485 EVC</td>
<td>2852-7101</td>
</tr>
<tr>
<td>Thermokon</td>
<td>1</td>
<td>PTM200 wireless transmitter</td>
<td></td>
</tr>
<tr>
<td>Thermokon</td>
<td>1</td>
<td>EasySens SRW01 window contact</td>
<td></td>
</tr>
<tr>
<td>Thermokon</td>
<td>1</td>
<td>Thermokon SR04PST room operating panel</td>
<td></td>
</tr>
</tbody>
</table>

2.3 Tools

<table>
<thead>
<tr>
<th>Designation</th>
<th>Item No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>e!COCKPIT workstation license</td>
<td>2759-101/1110-2002</td>
</tr>
</tbody>
</table>
3 Setup

Figure 1: Node Configuration

**Note**

**Node configuration!**
The node structure described is only one example of how communication with the EnOcean sensors can be realized. The modules may be exchanged and expanded as required by the respective application.
4 EnOcean Equipment Profile (EEP)

4.1 Structure of an EEP

By standardizing the communication profiles (EnOcean Equipment Profile, EEP), the interoperability of the terminals based on EnOcean technology is ensured. In this way, for example, sensors from one device manufacturer can communicate with receiver gateways from another manufacturer. The standard can be downloaded at www.enocean.com.

![Figure 2: Structure of an EEP](image)

An EEP consists of three fields:

1. **RORG or ORG** number describes the telegram type.
2. **FUNC** number describes the functionality of the data bytes.
3. **TYPE** number describes the properties of the device/device type.

The field values are displayed as hexadecimal numbers. The range of values is limited by the bits available (see Figure 2).

With the new EEP2.1 the telegram type is relabeled RORG number instead of ORG number (see below).

<table>
<thead>
<tr>
<th>Telegram</th>
<th>RORG</th>
<th>ORG</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPS</td>
<td>F6</td>
<td>05</td>
<td>Repeated Switch Communication</td>
</tr>
<tr>
<td>1BS</td>
<td>D5</td>
<td>06</td>
<td>1 Byte Communication</td>
</tr>
<tr>
<td>4BS</td>
<td>A5</td>
<td>07</td>
<td>4 Byte Communication</td>
</tr>
<tr>
<td>VLD</td>
<td>D2</td>
<td>=RORG</td>
<td>Variable Length Data</td>
</tr>
<tr>
<td>MSC</td>
<td>D1</td>
<td>=RORG</td>
<td>Manufacturer Specific Communication</td>
</tr>
<tr>
<td>ADT</td>
<td>A6</td>
<td>=RORG</td>
<td>Addressing Destination Telegram</td>
</tr>
<tr>
<td>SM_LRN_REQ</td>
<td>C6</td>
<td>=RORG</td>
<td>Smart Ack Learn Request</td>
</tr>
<tr>
<td>SM_LRN_ANS</td>
<td>C7</td>
<td>=RORG</td>
<td>Smart Ack Learn Answer</td>
</tr>
<tr>
<td>SM_REC</td>
<td>A7</td>
<td>=RORG</td>
<td>Smart Ack Reclaim</td>
</tr>
<tr>
<td>SYS_EX</td>
<td>C5</td>
<td>=RORG</td>
<td>Remote Management</td>
</tr>
</tbody>
</table>
5 Sample Program

Note

Installation of Sample Projects for e!COCKPIT

Sample programs can be called up from the e!COCKPIT Backstage view by clicking the Updates & Add-ons button in the navigation bar.

5.1 Tasks

This application note describes connecting various sensors that operate based on EnOcean wireless technology on the WAGO-I/O-SYSTEM. An EnOcean STC65-RS485-EVC gateway from Thermokon is connected to a serial interface (750-652) of the WAGO-I/O-SYSTEM. The application program is based on the WagoAppEnocean library.
5.2 Programming

Figure 4: Program for Receiving/Sending EnOcean Telegrams

Figure 4 shows the basic structure for programming an application with an EnOcean STC65-RS485-EVC wireless gateway from Thermokon. A requirement for communication with the gateway is the `FbEnOcean_STC65_RS485_EVC` function block.

To address the function block to the required RS-485 module, the designation of the module with serial interface (753-652) assigned under the device structure must be indicated at the “I_Port” input (see Fig. 4).

Figure 5: Device Structure – Module Designations (I_Port)

Different function blocks can be assigned to a master module via the “bPortEnOcean” input by entering the same number both at the inputs of the function blocks and at the input of a master module.

The Thermokon gateway has a 4-byte long station address. The station address is determined automatically and output at the “dwStationAddress” output.

Any further EnOcean function blocks in the program can only be used in conjunction with the `FbEnOcean_STC65_RS485_EVC` module.
### Function Blocks
The **FbEnOcean_STC65_RS485_EVC** function block may only be called up once per gateway.

### Serial Interface
The RS-485 module (750-652) should be used as the interface. The **FbEnOcean_STC65_RS485_EVC** function block configures the module with the following parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baud rate</td>
<td>9600</td>
</tr>
<tr>
<td>Data bits</td>
<td>8</td>
</tr>
<tr>
<td>Stop bits</td>
<td>1</td>
</tr>
<tr>
<td>Parity</td>
<td>Even</td>
</tr>
<tr>
<td>Duplex mode</td>
<td>Half duplex</td>
</tr>
</tbody>
</table>
5.3 Determining the ID Number

The FbEnoceanShowID function block (see Figure 6) is used to determine the ID number of a sensor. Each sensor has a unique ID number.

![EEP2.0: 05-02-01
EEP2.1: F6-02-01](image)

Figure 6: Function block for determining the ID number of a sensor.

A filter is specified by selecting the sensor type at the "bRF_TYPE" input so that only telegrams of this sensor type are identified by the function block. The value to be set at the "bRF_TYPE" input can be read from the EEPROM of the radio sensor to search for (see Figure 6).

There are three sensor types:

1. **bRF_TYPE** = 16#05 or 16#F6 is a sensor that transmits telegrams according to the EEPROM ORG number 16#05 (RORG number 16#F6). **Example**: Touch sensor with piezoelectric contact (e.g., PTM200)

2. **bRF_TYPE** = 16#06 or 16#D5 is a sensor that transmits telegrams according to the EEPROM ORG number 16#06 (RORG number 16#D5). **Example**: Window contact with solar power supply

3. **bRF_TYPE** = 16#07 or 16#A5 is a sensor that transmits telegrams according to the EEPROM ORG number 16#07 (RORG number 16#A5). **Example**: Thermokon SR04 room operating panel
5.3.1 Example: SR04 (Thermokon)

The ID number of an SR04 room operating panel (Thermokon) should be determined (see Figure 7). Because the room operating panel transmits the telegram according to the EEPROM ORG number 16#07, the `bRF_TYPE` input is set to “16#07”.

The “bNumberOfClicks” input is set to “2”. The sensor ID is then output with the same ID after receipt of two consecutive telegrams.

For the room operating panel to transmit two telegrams in succession, the learn or occupancy button (if available) must be pressed twice.

```c
PROGRAM PRG_ShowRoomOperatingPanel_ID // Description:
// Program to read the Identification Number (ID) of
// a room operating panel's telegram (4-Byte--Telegram, bRF_TYPE=16#07).
// by double clicking the occupancy button. (bNumberOfClicks=2).
VAR
  oFbEnOceanShowID: WagoAppEnOcean.FbEnOceanShowID;
  bRF_Type      : BYTE:=16#07;
  sStatus       : STRING;
  xUpdate       : BOOL;
  dwID          : DWORD;
END_VAR
```

Figure 7: Program for Determining the ID Number of the SR04 Room Operating Panel
5.4 Determining the Suitable EEP Function Block

The names of the function blocks identify the EEP used. There are two different EEP definitions (2.0 and 2.1), which, however, can be considered as synonymous. The WAGO function blocks are identified by the name from EEP2.1 and must be selected accordingly (see Figure 8).

**EEP2.0:** 05-02-01  
**EEP2.1:** F6-02-01

---

Information

Determining the EEP of a device

You can get the EEP of a device from the manufacturer directly.

If the information about the EEP of the device is unknown, the telegram can be received and evaluated in the raw data (see Section 5.5.4 on page 19).

---

Note

Selecting an EEP

The data will be evaluated incorrectly if the EEP is not selected correctly.
5.5 Receive Telegram

5.5.1 Evaluating the Signal of a Room Operating Panel

```
PROGRAM PRG_RoomOperatingPanel

// Description:
// Program to read a room operating panel’s telegram with EEP A5-10-01 or 07-10-01

VAR
  oFbEnoceanA51001_RoomOperatingPanel : WagoAppEnocean.FbEnoceanA51001_RoomOperatingPanel;
  dwID : DWORD := 16#00789;
```

![Figure 9: Reading a Room Operating Panel](image)

Figure 9 shows the program for reading a Thermokon SR04PST room operating panel. The room operating panel sends telegrams according to EEP 07-10-01 (A5-10-01). Accordingly, an instance of the `FbEnoceanA51001_RoomOperatingPanel` function block is called up. Module selection depends on the communication profile used by the device and must be customized for other room operating panels.

With the help of the `FbEnoceanShowID` function block, the ID number of the device can be determined (see Subsection 5.3 on page 13). The “`dwID`” variable must be initialized with the ID number determined or the ID number determined is entered as a constant number at the input of the function block.
5.5.2 Evaluating the Signal of a Touch Sensor

The telegram of an EnOcean PTM200 button should be evaluated. The PTM200 button transmits a telegram according to the EEP 05-02-01 (F6-02-01). To evaluate the telegram, an instance of the \texttt{FbEnoceanF60201\_RockerSwitch\_2\_Rocker} function block is used.

The output signals ("\texttt{xButton\textunderscore AO}" to "\texttt{xButton\textunderscore BI"}) correspond to the four contact grommets and are set to "TRUE" depending on the button pressed.

With the help of the \texttt{FbEnoceanShowID} function block, the ID number of the device can be determined (see Subsection 5.3 on page 13). The \texttt{dwID} variable must be initialized with the ID number determined or the ID number determined is entered as a constant number at the input of the function block.
5.5.3 Evaluating the Signal of a Window Contact

The telegram of an EnOcean STM250 window contact (Thermokon) should be evaluated.

The STM250 window contact transmits a telegram according to the EEP 06-00-01 (D5-00-01). To evaluate the telegram, an instance of the **FbD50001_SingleInputContact** function block is used.

With the help of the **FbEnoceanShowID** function block, the ID number of the device can be determined (see Subsection 5.3 on page 13). The **dwID** variable must be initialized with the ID number determined or the ID number determined is entered as a constant number at the input of the function block.
5.5.4 Receive Raw Data

If you do not know the EEP of the device, it is also possible to read the raw data of the device.

![Diagram of function block for reading 1-byte telegram](image1.png)

Figure 13: Reading the 1-byte Telegram

Figure 13 shows the program for reading a 1-byte telegram (RPS / 1-BS Telegram). An instance of the `FbEnocean_1BS_Receive` function block is called up.

![Diagram of function block for reading 4-byte telegram](image2.png)

Figure 14: Reading the 4-byte Telegram

Figure 14 shows the readout of a 4-byte telegram (4-BS Telegram). An instance of the `FbEnocean_4BS_Receive` function block is called up.
5.6 Transmit Telegram

5.6.1 Virtual EnOcean Button in CODESYS

Figure 15 shows the program for transmitting an EnOcean button signal. The `FbEnocean_RPS_Send` function block is used for this purpose. The function block transmits a telegram according to EEP F6-02-01.

```
PROGRAM PSG_Virtual_PushButton

// Description:
// PROGRAM TO send a push button's telegram with EEP F6-02-01 or 05-02-01.
// First the station address of the gateway must be queried.

VAR
  cFbEnocean_RPS_Send : WagoAppEnocean.FbEnocean_RPS_Send;
  dwOffset : DWORD := 1; // Offset
  x0       : BOOL;
  x1       : BOOL;
  x2       : BOOL;
  x3       : BOOL;
  xStatus  : STRING;
  xReady   : BOOL;
END_VAR

```

Figure 15: Transmit Button Signal

The inputs ("xButton_A0" to "xButton_B1") correspond to the four button signals of a switch. The function block allows simulation of a virtual button in the e!COCKPIT project.

The telegram to transmit requires a unique transmission ID number. The ID number of the telegram is calculated by adding the station address of the gateway (`Thermokon_STC.dwStationAddress`) and an offset (`dwOffset`). The value range of the offset goes from 1 to 127. Therefore, there are a total of 127 transmission ID numbers available.
5.6.2 Transmit Data Byte

Figure 16 shows the program for transmitting a 4-byte telegram (4-BS Telegram).

An instance of the **FbEnocean_4BS_Send** function block is called up.

```plaintext
PROGRAM Prs_Send_4_BYTE_Telegram

  // Description:
  // PROGRAM IO send 4-BS-Telegram
  // First the station address of the gateway must be queried.

VAR
  oFbEnocean_4BS_Send : WagoAppEnocean.FbEnocean_4BS_Send;
  dwOffset       : DWORD:=3;   //Offset 3
  bHysteresis    : BYTE;
  tTimeout       : TIME:=741S;
  xUpdate        : BOOL;
  bDB3           : BYTE;
  bDB2           : BYTE;
  bDB1           : BYTE;
  bDB0           : BYTE;
  sStatus        : STRING;
  xReady         : BOOL;
END_VAR
```

So that the STC-RS485-EVC gateway can transmit the telegram, the telegram to send requires a unique address, i.e. a so-called ID number. The ID number of the telegram is calculated by adding the station address of the gateway (**Thermokon_STC.dwStationAddress**) and the address offset (**dwOffset**). The value range of the offset goes from 1 to 127. Therefore, there are a total of 127 transmission ID numbers available.

![Figure 16: Transmitting the 4-byte telegram](image-url)
### 5.7 Bidirectional Communication

#### 5.7.1 Bidirectional Function Block

Some devices use an EnOcean profile for bidirectional communication. The WAGO library provides the function blocks for this purpose. The `FbEnoceanA52001_BatteryPoweredActuator` function block is an example.

#### Figure 17: Program for Bidirectional Communication

Figure 17 shows the program using bidirectional communication with an MD15-FtL-HE small actuator from Kieback+Peter. The small actuator uses EEP 07-20-01 (A5-20-01). For each small drive to address, an instance of the `FbEnoceanA52001_BatteryPoweredActuator` function block is required. The data of the small actuator to be addressed is read (“`typbatteryPoweredActuatorReadParameters`”) and the input values can be transferred to the device (“`typbatteryPoweredActuatorWriteParameters`”).

The module and the device must be “introduced” to one another as radio communication partner devices at the beginning of commissioning. The module must be set to learning mode by setting the “`xLearn`” input (`xLearn` = “TRUE”).

Then press the button on the small actuator. The device then transmits a radio telegram that is received by the module. The device ID that is received is indicated at the “`dwLearn_ID`” input and stored. The “`xLearn`” variable is reset when the module ID has been successfully received.