BECKHOFF New Automation Technology

Manual | EN

TS8035

TwinCAT 2 | FIAS Server

Supplement | Building Automation

Table of contents

BECKHOFF

Version: 1.0

1   Foreword

1.1   Notes on the documentation

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning the components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.
We reserve the right to revise and change the documentation at any time and without prior announcement.
No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH.
Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents:
EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702
with corresponding applications or registrations in various other countries.

EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.
The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.
Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.

1.2   Safety instructions

Safety regulations

Please note the following safety instructions and explanations!
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

Description of symbols

In this documentation the following symbols are used with an accompanying safety instruction or note. The safety instructions must be read carefully and followed without fail!

⚠ DANGER
Serious risk of injury!
Failure to follow the safety instructions associated with this symbol directly endangers the life and health of persons.

⚠ WARNING
Risk of injury!
Failure to follow the safety instructions associated with this symbol endangers the life and health of persons.

⚠ CAUTION
Personal injuries!
Failure to follow the safety instructions associated with this symbol can lead to injuries to persons.

NOTE
Damage to the environment or devices
Failure to follow the instructions associated with this symbol can lead to damage to the environment or equipment.

●
Tip or pointer
ℹ This symbol indicates information that contributes to better understanding.

1.3 Notes on information security

The products of Beckhoff Automation GmbH & Co. KG (Beckhoff), insofar as they can be accessed online, are equipped with security functions that support the secure operation of plants, systems, machines and networks. Despite the security functions, the creation, implementation and constant updating of a holistic security concept for the operation are necessary to protect the respective plant, system, machine and networks against cyber threats. The products sold by Beckhoff are only part of the overall security concept. The customer is responsible for preventing unauthorized access by third parties to its equipment, systems, machines and networks. The latter should be connected to the corporate network or the Internet only if appropriate protective measures have been set up.

In addition, the recommendations from Beckhoff regarding appropriate protective measures should be observed. Further information regarding information security and industrial security can be found in our https://www.beckhoff.com/secguide.

Beckhoff products and solutions undergo continuous further development. This also applies to security functions. In light of this continuous further development, Beckhoff expressly recommends that the products are kept up to date at all times and that updates are installed for the products once they have been made available. Using outdated or unsupported product versions can increase the risk of cyber threats.

To stay informed about information security for Beckhoff products, subscribe to the RSS feed at https://www.beckhoff.com/secinfo.

BECKHOFF

2 Introduction

Requirements

The Fidelio FIAS protocol (Fidelio Interface and Application Specification) has assumed a leading position worldwide for hotel management software and plays an ever larger part in applications in larger hotels. Since the demand for networked systems and room automation is growing, the TwinCAT FIAS Server with a direct FIAS interface enables trouble-free communication between hotel management system and PLC.

micros FIDELIO

The TwinCAT FIAS Server is a software package for accessing systems according to the FIAS standard (Fidelio Interface and Application Specification).
It enables communication between a TwinCAT PLC and a system with a communication interface that is compliant with the FIAS standard.
Communication takes place by TCP/IP.
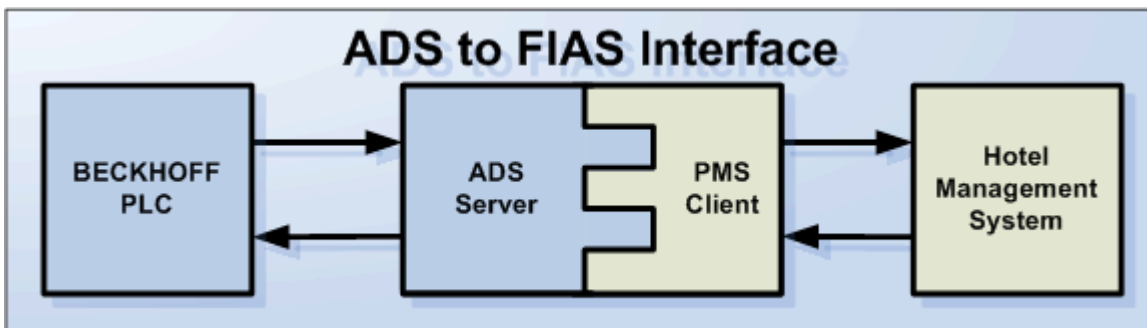
The connection of the hotel management software with the automation system helps to improve the efficiency of the system. Hence, the automation system can react automatically to the hotel management system's room occupancy plan. The air conditioning is adapted accordingly if the room is not occupied, or the sun shade is automatically activated in strong sunlight in the summer.
Apart from the possibility of optimizing energy consumption, customer service can also be improved. Messages between customers and the hotel management software can additionally be implemented using the TwinCAT FIAS Server for customer-orientated services.

3   Hardware requirements

The TwinCAT FIAS Server can be used on any PC controller that is compatible to TwinCAT (except for the CX9000).

The following are required / necessary for the installation of the TwinCAT FIAS Server on a CX x86 or CX ARM processor-based hardware platform:

- TwinCAT PLC Runtime
- Microsoft .NET Framework \ Compact .NET Framework.
- Ethernet interface

Please refer to the Beckhoff Information System as to which hardware with which image fulfils these requirements.

BECKHOFF

4 User requirements

The user of this ADS server / this library requires basic knowledge of the following:

- Handling the TwinCAT PLC Control
- Handling the TwinCAT System Manager
- PC and network knowledge
- Relevant safety regulations for building technical equipment

5 Installation / Commissioning

5.1 Installation under Windows CE (WEC)

Transfer of the installation to the Windows CE device

If the TwinCAT FIAS Server has been successfully installed on a Windows Standard device, two subfolders will be found in the "**..\TwinCAT\CE\FIAS Server**" folder.

- **\ARMV4I**
    - This folder contains the CE installation:
      TcFIASServerCe.ARMV4I.CAB for ARMV4 CPUs (e.g. CX900x)
- **\I586**
    - This folder contains the CE installation:
      TcFIASServerCe.I586.CAB for X86 CPUs (e.g. CX10xx)

Transfer the file "**TcFIASServerCe.ARMV4I.CAB**" or "**TcFIASServerCe.I586.CAB**" to the Windows CE device. This can take place via

- a shared folder (public folder)
- FTP
- ActiveSync
- Compact Flash (CF) adaptor

Installation on the Windows CE device

The CAB file installation can be executed with a normal double click. Confirm the preselected (default) directory "**\hard disk\System\TC-FIAS-Server**" with "OK".

Following the installation, the CAB file will be deleted automatically. Restart the device after the installation has finished.

5.2 Installation under Windows Embedded Standard (WES)

Starting the installation

Start the installation and follow the instructions in the dialogs.

After the installation.

After the installation has finished, you will find two new directories in your TwinCAT folder: **"..\TwinCAT\CE\FIAS Server"** and **"..\TwinCAT\FIAS Server".**
The installation files (.CAB) for the different Windows CE platforms ARM and X86 can be found in the directory **"..\TwinCAT\CE\FIAS Server Server"**.

5.3 Configuration

The TwinCAT FIAS Server is configured by means of a text file in XML format (**TcFIASServer.cfg**). The file is provided with the installation and is in the installation directory after completion. The file may neither be moved nor renamed. The table below describes the individual parameters required for the configuration of connections to a hotel management server. Furthermore, there is a description of the dependencies in which a backup file is created and how it is updated during service operation.

TcFIASServer.cfg (example)

```xml
<?xml version="1.0" encoding="utf-8" ?>
<TcFiasServerConfig>
   <Client Name="FIAS1" Port="5001" Ip="192.168.0.20" ResyncCycleTime="120" SaveCycleTime="120" Save
Type="2" LinkAliveCycleTime="5" />
</TcFiasServerConfig>
```

> **i** If you wish to establish a connection to more than one hotel management server, then a further FiasClient configuration must be entered in the TcFIASServer.cfg file. See the following example.

```xml
<?xml version="1.0" encoding="utf-8" ?>
<TcFiasServerConfig>
   <Client Name="FIAS1" Port="5001" Ip="192.168.0.21" ResyncCycleTime="120" SaveCycleTime="120" Save
Type="2" LinkAliveCycleTime="5" />
   <Client Name="FIAS2" Port="5001" Ip="192.168.0.22" ResyncCycleTime="120" SaveCycleTime="120" Save
Type="2" LinkAliveCycleTime="5" />
   <Client Name="FIAS3" Port="5001" Ip="192.168.0.23" ResyncCycleTime="120" SaveCycleTime="120" Save
Type="2" LinkAliveCycleTime="5" />
</TcFiasServerConfig>
```

Description of parameters

> **i** If a parameter cannot be read, or if it lies outside the permitted range, then each call of a PLC function block from the TcFIAS.Lib returns a specific error. This occurs until the error has been rectified.

Parameter	Description
Name	Name of the hotel management server. **Note The name specified here may not be used several times in the configuration.**
Port	The TCP port of the hotel management server. **Note The TCP port is variable and is specified by the integrator of the Fidelio management server.**
Ip	IP Address of the hotel management server. **Note The IP Address is variable and is specified by the integrator of the Fidelio management server.**
ResyncCycleTime	Specifies the time interval between resyncs of the database. The TwinCAT FIAS Server compares, based on a time stamp in the backup file, whether the time interval to the previous resync has been exceeded. If the ResyncCycleTime has been exceeded, a database resync is performed at the next establishment of a connection to the management server. After a successful resync of the database, the data received are always written to the backup file. The min. value is 120 minutes and the max. value is 65535 minutes. **Note If ResyncCycleTime set to an invalid value or a value outside the range, the default value of 120 minutes is used automatically.**
SaveType	The SaveType specifies the cases in which the backup file is to be written. • 1: In the event of a change. • 2: At periodic intervals. **Note If SaveType set to an invalid value, the default value 2 is used automatically.**
SaveCycleTime	Specifies the time interval between the interval for writing the backup file. The SaveCycleTime is expected in minutes. The min. value is 30 minutes, and the max. value is 65535 minutes. **Note A small value means that the backup file is often saved and the mass storage device of the controller is thus accessed more frequently. Therefore, select the parameter so that the number of accesses to the backup file is as small as possible.**

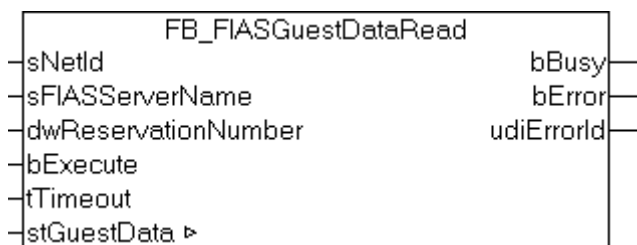| Parameter	Description
	Note **If SaveCycleTime set to an invalid value or a value outside the range, the default value of 30 minutes is used automatically.**
LinkAliveCycleTime	Specifies the time interval between the interval for sending a link alive telegram. The LinkAliveCycleTime is expected in minutes. The min. value is 5 minutes, and the max. value is 30 minutes. *Note* **If LinkAliveCycleTime set to an invalid value or a value outside the range, the default value of 5 minutes is used automatically.**

6   PLC API

Requirements

Name	Function
FB_FIASGuestDataRead [▶ 14]	This function block is used to read guest data.
FB_FIASRoomDataRead [▶ 15]	This function block is used to read room data.
FB_FIASRoomDataWriteRoomStatus [▶ 16]	This function block is used to write room status.
FB_FIASGetClientConnectionState [▶ 17]	This function block is used to read the status of the connection between the TwinCAT FIAS Server and the hotel management server.
FB_FIASGetDeviceInfo [▶ 18]	This function block is used to read the name and the version of the TwinCAT FIAS Server.

6.1   FB_FIASGuestDataRead

```
                FB_FIASGuestDataRead
─┤sNetId                              bBusy├─
─┤sFIASServerName                     bError├─
─┤dwReservationNumber             udiErrorId├─
─┤bExecute
─┤tTimeout
─┤stGuestData ▷
```

This block reads the guest data belonging to a reservation number.

Examples

Download demo project [▶ 21]

VAR_INPUT

```
sNetId                : T_AmsNetId;
sFIASServerName       : STRING;
dwReservationNumber   : DWORD;
bExecute              : BOOL;
tTimeout              : TIME := t#5s;
```

sNetId: A string containing the AMS Net ID of the target device to which the command is addressed.

sFIASServerName: A string containing the name of the hotel management server to which the TcFIASServer passes on the FIAS commands.

dw**ReservationNumber:** A dword containing the reservation number.

bExecute: The command is triggered by a rising edge at this input.

tTimeout: States the time before the function is cancelled.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
udiErrorId  : UDINT;
```

bBusy: This output remains TRUE until the block has executed a command request, but at the longest for the time period set at the *tTimeout* input. No new commands are accepted at the inputs if *bBusy* = TRUE.

bError: This output is switched to TRUE as soon as an error occurs during the execution of a command. The command-specific error code [▶ 20] is contained in *udiErrorId*. Is reset to FALSE by the execution of a command at the inputs.

udiErrorId: Contains the command-specific error code [▶ 20] of the most recently executed command. Is reset to 0 by the execution of a command at the inputs.
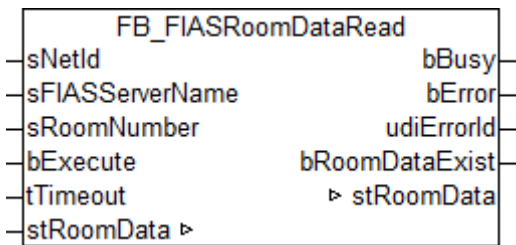
VAR_IN_OUT

```
stGuestData   : ST_FIASGuestData;
```

stGuestData : Containing the requested guest data [▶ 19] to the reservation number.

Requirements

Development environment	Target system	Required libraries
TwinCAT 2.11 R3/x64	PC/CX	TcFIAS library from V1.0.0

6.2 FB_FIASRoomDataRead

```
        FB_FIASRoomDataRead
—sNetId                          bBusy—
—sFIASServerName                 bError—
—sRoomNumber                  udiErrorId—
—bExecute               bRoomDataExist—
—tTimeout                 ▷ stRoomData—
—stRoomData ▷
```

This block reads all reservation numbers that exist for a room.

Examples

Download demo project [▶ 21]

VAR_INPUT

```
sNetId          : T_AmsNetId;
sFIASServerName : STRING;
sRoomNumber     : STRING(8);
bExecute        : BOOL;
tTimeout        : TIME := t#5s;
```

sNetId: A string containing the AMS Net ID of the target device to which the command is addressed.

sFIASServerName: A string containing the name of the hotel management server to which the TcFIASServer passes on the FIAS commands.

sRoomNumber: A string with max. 8 alphanumeric characters containing the room number.

bExecute: The command is triggered by a rising edge at this input.

tTimeout: States the time before the function is cancelled.

VAR_OUTPUT

```
bBusy           : BOOL;
bError          : BOOL;
udiErrorId      : UDINT;
bRoomDataExist  : BOOL;
```

bBusy: This output remains TRUE until the block has executed a command request, but at the longest for the time period set at the *tTimeout* input. No new commands are accepted at the inputs if *bBusy* = TRUE.

bError: This output is switched to TRUE as soon as an error occurs during the execution of a command. The command-specific error code [▶ 20] is contained in *udiErrorId*. Is reset to FALSE by the execution of a command at the inputs.

udiErrorId: Contains the command-specific error code [▶ 20] of the most recently executed command. Is reset to 0 by the execution of a command at the inputs.

bRoomDataExist: Output is switched to TRUE, if the TwinCAT FIAS Server has received room data to the specified room number of the hotel management server.
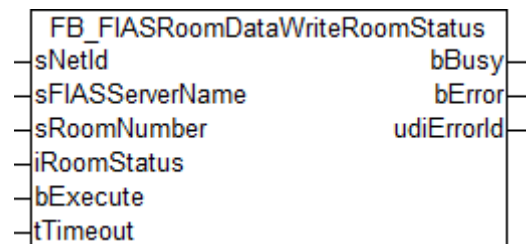
VAR_IN_OUT

```
stRoomData  : ST_FIASRoomData;
```

stRoomData: Containing the requested room data [▶ 19] to the room number.

Requirements

Development environment	Target system	Required libraries
TwinCAT 2.11 R3/x64	PC/CX	TcFIAS library from V1.0.0

6.3   FB_FIASRoomDataWriteRoomStatus

This block sends a room status to hotel management server.

Examples

Download demo project [▶ 21]

VAR_INPUT

```
sNetId             : T_AmsNetId;
sFIASServerName    : STRING;
sRoomNumber        : STRING(8);
iRoomStatus        : INT;
bExecute           : BOOL;
tTimeout           : TIME := t#5s;
```

sNetId: A string containing the AMS Net ID of the target device to which the command is addressed.

sFIASServerName: A string containing the name of the hotel management server to which the TcFIASServer passes on the FIAS commands.

sRoomNumber: A string with max. 8 alphanumeric characters containing the room number.

iRoomStatus: A int containing the room status.

bExecute: The command is triggered by a rising edge at this input.

tTimeout: States the time before the function is cancelled.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
udiErrorId : UDINT;
```

bBusy: This output remains TRUE until the block has executed a command request, but at the longest for the time period set at the *tTimeout* input. No new commands are accepted at the inputs if *bBusy* = TRUE.
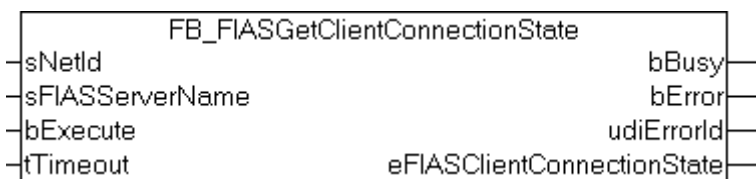
bError: This output is switched to TRUE as soon as an error occurs during the execution of a command. The command-specific <u>error code [▶ 20]</u> is contained in *udiErrorId*. Is reset to FALSE by the execution of a command at the inputs.

udiErrorId: Contains the command-specific <u>error code [▶ 20]</u> of the most recently executed command. Is reset to 0 by the execution of a command at the inputs.

Requirements

Development environment	Target system	Required libraries
TwinCAT 2.11 R3/x64	PC/CX	TcFIAS library from V1.0.0

6.4   FB_FIASGetClientConnectionState

```
                 FB_FIASGetClientConnectionState
  ─┤sNetId                                          bBusy├─
  ─┤sFIASServerName                                 bError├─
  ─┤bExecute                                       udiErrorId├─
  ─┤tTimeout                   eFIASClientConnectionState├─
```

This block reads the status of a connection to a hotel management server.

Examples

VAR_INPUT

```
sNetId             : T_AmsNetId;
sFIASServerName    : STRING(80);
bExecute           : BOOL;
tTimeout           : TIME := t#5s;
```

sNetId: A string containing the AMS Net ID of the target device to which the command is addressed.

sFIASServerName: A string containing the name of the hotel management server to which the TcFIASServer passes on the FIAS commands.

bExecute: The command is triggered by a rising edge at this input.

tTimeout: States the time before the function is cancelled.

VAR_OUTPUT

```
bBusy              : BOOL;
bError             : BOOL;
udiErrorId         : UDINT;
eConnectionState   : E_FIASClientConnectionState;
```

bBusy: This output remains TRUE until the block has executed a command request, but at the longest for the time period set at the *tTimeout* input. No new commands are accepted at the inputs if *bBusy* = TRUE.

bError: This output is switched to TRUE as soon as an error occurs during the execution of a command. The command-specific <u>error code [▶ 20]</u> is contained in *udiErrorId*. Is reset to FALSE by the execution of a command at the inputs.

udiErrorId: Contains the command-specific <u>error code [▶ 20]</u> of the most recently executed command. Is reset to 0 by the execution of a command at the inputs.

eConnectionState: This output is set to eConnected when the hotel management server is connected. Is there no connection [▶ 19], the output is set to eDisconnected.

Requirements

Development environment	Target system	Required libraries
TwinCAT 2.11 R3/x64	PC/CX	TcFIAS library from V1.0.0

6.5   FB_FIASGetDeviceInfo

```
           FB_FIASGetDeviceInfo
─sNetId                          bBusy─
─bExecute                        bError─
─tTimeout                     udiErrorId─
                         udiDeviceVersion─
                            sDeviceName─
```

This block reads the name and the version number of the TwinCAT FIAS Server.

Examples

Download demo project [▶ 21]

VAR_INPUT

```
sNetId          : T_AmsNetId;
bExecute        : BOOL;
tTimeout        : TIME := t#5s;
```

sNetId: A string containing the AMS Net ID of the target device to which the command is addressed.

bExecute: The command is triggered by a rising edge at this input.

tTimeout: States the time before the function is cancelled.

VAR_OUTPUT

```
bBusy             : BOOL;
bError            : BOOL;
udiErrorId        : UDINT;
udiDeviceVersion  : UDINT;
sDeviceName       : STRING;
```

bBusy: This output remains TRUE until the block has executed a command request, but at the longest for the time period set at the *tTimeout* input. No new commands are accepted at the inputs if *bBusy* = TRUE.

bError: This output is switched to TRUE as soon as an error occurs during the execution of a command. The command-specific error code [▶ 20] is contained in *udiErrorId*. Is reset to FALSE by the execution of a command at the inputs.

udiErrorId: Contains the command-specific error code [▶ 20] of the most recently executed command. Is reset to 0 by the execution of a command at the inputs.

udiDeviceVersion: Version of the TwinCAT FIAS Server.

sDeviceName: Name of the TwinCAT FIAS Server.

Requirements

Development environment	Target system	Required libraries
TwinCAT 2.11 R3/x64	PC/CX	TcFIAS library from V1.0.0

6.6　Enum / Structure

ST_FIASGuestData

```
TYPE ST_FIASGuestData :
STRUCT
   dwReservationNumber    : DWORD;
   dtDateTime             : DATE_AND_TIME;
   dGuestArrivalDate      : DATE;
   dGuestDepartureDate    : DATE;
   sRoomNumber            : STRING(8);
   sGuestFirstName        : STRING(20);
   sGuestName             : STRING(40);
   sGuestTitle            : STRING(20);
   eGuestLanguage         : E_FIASGuestLanguage := -1;
   sUserDefinableField00  : STRING(40);
   sUserDefinableField01  : STRING(40);
   sUserDefinableField02  : STRING(40);
   sUserDefinableField03  : STRING(40);
   sUserDefinableField04  : STRING(40);
   sUserDefinableField05  : STRING(40);
   sUserDefinableField06  : STRING(40);
   sUserDefinableField07  : STRING(40);
   sUserDefinableField08  : STRING(40);
   sUserDefinableField09  : STRING(40);
   bShareFlag             : BOOL;
END_STRUCT
END_TYPE
```

ST_FIASRoomData

```
TYPE ST_FIASRoomData :
STRUCT
   arrReservationNumbers  : ARRAY[1..10] OF DWORD;
   sClassOfService        : STRING(10);
END_STRUCT
END_TYPE
```

E_FIASGuestLanguage

```
TYPE E_FIASGuestLanguage :
(
   eFIASGuestLanguagesUndefined := -1,
   eFIASEnglishAmerican := 1,
   eFIASFrench,
   eFIASGerman,
   eFIASItalian,
   eFIASJapanese,
   eFIASSpanish
);
END_TYPE
```

E_FIASClientConnectionState

```
TYPE E_FIASClientConnectionState :
(
   eFIASClientUnknown := 0,
   eFIASClientInitialized,
   eFIASClientInitializing,
   eFIASClientConnected,
   eFIASClientConnecting,
   eFIASClientDisconnected,
   eFIASClientDisconnecting
);
END_TYPE
```

7 Error codes

Hex code	Dec code	Description
0x8501	34049	Input parameter error on a PLC function block.
0x8502	34050	The configuration file **TcFIASServer.cfg** was not found. See Configuration [▶ 11]
0x8503	34051	Internal server error while processing the TcFIASServer configuration file **TcFIASServer.cfg.**
0x8504	34052	Error in the configuration file **TcFIASServer.cfg**. The XML node **"Client"** was not found. See Configuration [▶ 11]
0x8505	34053	Error in the configuration file **TcFIASServer.cfg**. The XML attribute **"Ip"** was not found in the Client XML node, or it contains an incorrect value. See Configuration [▶ 11]
0x8506	34054	Error in the configuration file **TcFIASServer.cfg**. The XML attribute **"Port"** was not found in the Client XML node, or it contains an incorrect value. See Configuration [▶ 11]
0x8507	34055	Error in the configuration file **TcFIASServer.cfg**. The XML attribute **"Name"** was not found in the Client XML node, or it contains an incorrect value. See Configuration [▶ 11]
0x8508	34056	Error in the configuration file **TcFIASServer.cfg**. The XML attribute **"SaveType"** was not found in the Client XML node, or it contains an incorrect value. See Configuration [▶ 11]
0x8509	34057	Error in the configuration file **TcFIASServer.cfg**. The XML attribute **"SaveCycleTime"** was not found in the Client XML node, or it contains an incorrect value. See Configuration [▶ 11]
0x850A	34058	Error in the configuration file **TcFIASServer.cfg**. The XML attribute **"ResyncCycleTime"** was not found in the Client XML node, or it contains an incorrect value. See Configuration [▶ 11]
0x850B	34059	Error while reading a client status. The specified client was not found.
0x850C	34060	Internal server error while reading a client status.
0x850D	34061	Error whilst reading room data. The specified room was not found or there is no room information available for reading.
0x850E	34062	Internal server error while reading room data.
0x850F	34063	Error whilst reading guest data, guest data was not found.
0x8510	34064	Internal server error while reading guest data.
0x8511	34065	Error in the configuration file **TcFIASServer.cfg**. The XML attribute **"LinkAliveCycleTime"** was not found in the Client XML node, or it contains an incorrect value. See Configuration [▶ 11]
0x8512	34066	Internal server error while writing room data.
0x8513	34067	Internal server error while writing room data (room status).

8   Programming examples

Download demo project

https://infosys.beckhoff.com/content/1033/tcfiasserver/Resources/zip/11363343883.zip

More Information:
www.beckhoff.com/ts8035