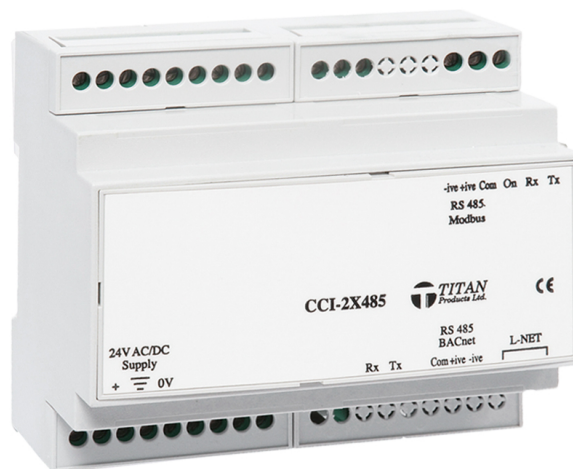


# CCI-2X485

## Modbus to BACnet Gateways



**BACnet Enabled**

## Features

- 24V AC/DC supply
- 2 x RS485 communications ports
- Native BACnet communications
- BACnet MS/TP to Modbus RTU Master
- 200 BACnet Object mapping
- Supports Modbus input registers
- PC programming (upload via FPT601)
- Din Rail Mount

## Specification

Power Supply	24VAC/DC
Power Consumption	5VA MAX
Communications	2 x RS485
Network	BACnet (MS/TP) and Modbus RTU
Baud Rates BACnet	9,600 19,200 38,400 or 76,800
Baud Rates Modbus	9,600 19,200 or 38,400
Indication	Tx/Rx comm.'s
Enclosure	Din Mounting (IP20)
Enclosure Rating	L94-VO
Size	106 mm wide 92 mm high 62 mm deep

## TITAN GATEWAY CONFIGURATION

This application sets up the Modbus to BACnet mapping objects through the use of a PC with software provided by Titan Products and the FPT601 Field Programming Tool. The application can support the following maximum object mapping of a Modbus types in any combination:

200 (150 with Intrinsic Reporting) Modbus Discrete Inputs to BACnet Binary Input Objects or  
200 (150 with Intrinsic Reporting) Modbus Coils to BACnet Binary Value Objects or  
200 (150 with Intrinsic Reporting) Modbus Input Registers to BACnet Analog Input Objects or  
200 (150 with Intrinsic Reporting) Modbus Holding Registers to BACnet Binary Value Objects or  
200 (150 with Intrinsic Reporting) Modbus Holding Registers to BACnet Analog Value Objects

The BACnet Device Object needs to be setup before any Modbus devices or mapping objects can be created. This is achieved by double clicking the Device Object or selecting *Tools ... Device Setup*. Having setup the Device Object, Modbus slave devices can be added by right clicking below the Device Object and selecting *Add Slave* (or *Tools ... Add ... Slave*). Mapping objects can be added to a Modbus device by first selecting a Modbus Device and selecting *Tools ... Add ... Mapping* or right clicking in the mapping window. The bottom left of the window details the number of Modbus Devices, mappings and the percentage of memory used.

Mapping objects can be removed by selecting the required mapping objects and selecting *Tools ... Remove ... Mapping(s)* or by right clicking in the mapping window. Slave devices and associated mappings can be removed by selecting *Tools ... Remove ... Slave* or by the right clicking in the Modbus device window.

The parameters for the Device Object, Modbus device or Mapping object can be displayed by double clicking the required item or by right clicking and selecting properties.

Having set a Modbus device to permit Automatic Device Mapping Copy, further devices can be copied from this by first selecting the base device and selecting *Tools ... Copy Mappings* or by right clicking on the device.

The current mapping data can be downloaded to the FPT601 Field Programming Tool by selecting *Program ... Write* or can be read from the FPT by selecting *Program ... Read*.

## DEVICE OBJECT SETUP

The following parameters can be setup:

### Instance

A unique device instance number throughout the entire BACnet network this is also known as the Device ID.

**Setting range 0 to 4194302**

### Name

A unique device object name throughout the entire BACnet network.

**Maximum 8 characters that identify the controller device**

### Location

The physical location of the gateway device.

**Maximum 8 characters that identify the controller location (This is not a mandatory setting)**

### Software Version

The version of the mapping configuration. This is automatically allocated in the controller configuration

### APDU Timeout

The BACnet APDU Timeout. **Range 0 to 6000 milli-seconds**

### APDU Retries

The number of BACnet APDU retries. **Range 1 to 10**

### MS/TP MAC (BACnet)

A unique MS/TP MAC address for BACnet. **Range 1 to 127**

### MS/TP Baud

The Baud rate of the MS/TP BACnet network. **Options: 9600/19200/38400/76800**

### Modbus Baud

The Baud rate of the Modbus network. **Options: 9600/19200/38400**

### Modbus Parity

The Parity of the Modbus network. **Options: Even/Odd/None.**

### Intrinsic Reporting

Select if Intrinsic Reporting is required for the BACnet Objects. The operation of Intrinsic Reporting reduces the maximum number of supported mapping,

## MODBUS SLAVE DEVICE SETUP

The following parameters can be setup:

### Slave Address

The MAC address of the Modbus device. **Range: 1 to 247**

### Timeout (mSec) Range: 1 to 1000 milli-secs

The time in milli seconds that the gateway will wait to receive a response for a message sent to the addressed Modbus slave device. This needs to be set large enough to allow time for the slave device to respond. However, it should not be set too large as the gateway will wait this time when a Modbus device is powered off etc.

### 32 bit Data Order

The Modbus register ordering for 32 bit data. As this requires x2 Modbus registers, the first or second register could contain the most significant data and this can be set accordingly.

### **Write Single Coil Supported**

This needs to be selected if the Modbus device supports the Write Single Coil Function code as some devices only support Multiple writes.

### **Write Single Register Supported**

This needs to be selected if the Modbus device supports the Write Multiple Registers Function code as some devices only support Multiple writes.

### **Automatic Device Mapping Copy**

This needs to be selected if the Modbus device will act as a base device from which other Modbus devices and associated mappings will be created. It can only be selected before adding any mapping objects. The mapping objects BACnet instance number are then restricted to the range of 0 to 99 and automatically has the Modbus address \* 100 added. For example, if the Modbus address is set to 7, the BACnet instance number can be in the range 700 to 799. When further devices are copied from this, the new Modbus address is used to create the mappings BACnet instance number. A similar scheme is used for the BACnet object name, and on a copied device, the new Modbus address is added to the front of the original object name. This insures that the instance number and object name are unique.

## **DISCRETE INPUT MAPPING**

The following parameters can be setup:

### **Data Address**

The Modbus data address for the discrete input.

### **Poll Rate (sec)**

The time in seconds between Modbus reads.

### **Object Type**

The BACnet object type that represents the mapping. A discrete input is represented as a Binary Input that is Read only.

### **Instance Number**

The BACnet Instance number of the mapping. This needs to be unique within the gateway for a particular data type.

### **Object Name**

The BACnet Object name of the mapping. This needs to be unique within the gateway and it is recommended to keep this short to conserve memory.

### **Description**

The BACnet Description of the mapping. It is recommended to keep this short to conserve memory.

### **Active State Text**

The BACnet Active State Text of the mapping. It is recommended to keep this short to conserve memory.

### **Inactive State Text**

The BACnet Inactive State Text of the mapping. It is recommended to keep this short to conserve memory.

### **Device Type**

The BACnet Device Type of the mapping. It is recommended to keep this short to conserve memory.

## **Polarity**

The BACnet Polarity of the mapping:

Normal Polarity : Modbus Discrete Input High -> BACnet Present Value Active

Reverse Polarity: Modbus Discrete Input High -> BACnet Present Value Inactive

## **COIL MAPPING**

The following parameters can be setup:

### **Data Address**

The Modbus data address for the coil.

### **Poll Rate (sec)**

The time in seconds between Modbus reads/writes.

### **Primary Function**

When set to Read, the coil will be constantly read from and only written to when changed by BACnet.

When set to Write, the coil will be constantly written to.

### **Power On Default**

When the Primary Function is set to Write, this value will be written to the coil on power up.

### **Object Type**

The BACnet object type that represents the mapping. A coil is represented as a Binary Value that can be Read/Write.

### **Instance Number**

The BACnet Instance number of the mapping. This needs to be unique within the gateway for a particular data type.

### **Object Name**

The BACnet Object name of the mapping. This needs to be unique within the gateway and it is recommended to keep this short to conserve memory.

### **Description**

The BACnet Description of the mapping. It is recommended to keep this short to conserve memory.

### **Active State Text**

The BACnet Active State Text of the mapping. It is recommended to keep this short to conserve memory.

### **Inactive State Text**

The BACnet Inactive State Text of the mapping. It is recommended to keep this short to conserve memory.

## **INPUT REGISTER MAPPING**

The following parameters can be setup:

### **Register Address**

The Modbus register address for the input.

### **Poll Rate (sec)**

The time in seconds between Modbus reads.

### **Data Format**

The storage of data within the Modbus Register(s).

Note that the Modbus Device 32 bit Data Order parameter needs to be setup for 32 bit data.

### **Slope**

#### **Intercept**

Data conversions to and from Modbus are performed as follows:

$BACnetValue = ModbusValue * Slope + Intercept$

$ModbusValue = (BACnetValue - Intercept) / Slope$

### **Object Type**

The BACnet object type that represents the mapping. An input register is represented as an Analog Input that is Read Only

### **Instance Number**

The BACnet Instance number of the mapping. This needs to be unique within the gateway for a particular data type.

### **Object Name**

The BACnet Object name of the mapping. This needs to be unique within the gateway and it is recommended to keep short.

### **Description**

The BACnet Description of the mapping. It is recommended to keep this short to conserve memory.

### **Device Type**

The BACnet Device Type of the mapping. It is recommended to keep this short to conserve memory.

### **Units**

The BACnet Units of the mapping.

### **Min Present Value**

The minimum BACnet writeable Present Value.

### **Max Present Value**

The maximum BACnet writeable Present Value.

## **HOLDING REGISTER MAPPING**

The following parameters can be setup:

### **Register Address**

The Modbus register address for the holding register.

### **Poll Rate (sec)**

The time in seconds between Modbus reads/writes.

### **Primary Function**

When set to Read, the holding register will be constantly read from and only written to when changed by BACnet. When set to Write, the holding register will be constantly written to.

### Power On Default

When the Primary Function is set to Write, this value will be written to the holding register on power up.

### Object Type

The BACnet object type that represents the mapping. A holding register is represented as an Analog Value or Binary Value. Set this to Binary Value if a single bit of a register is required.

### Instance Number

The BACnet Instance number of the mapping. This needs to be unique within the gateway for a particular data type.

### Object Name

The BACnet Object name of the mapping. This needs to be unique within the gateway and it is recommended to keep this short to conserve memory.

### Description

The BACnet Description of the mapping. It is recommended to keep this short to conserve memory.

*The following parameters can be setup if the Object Type is set to Analog Value:*

### Data Format

The storage of data within the Modbus Register(s). If a format is selected that is not a complete register (i.e. Signed 8 bits High Byte), the whole register will first be read, the section updated as required, and then written back. Note that the Modbus Device 32 bit Data Order parameter needs to be setup for 32 bit data.

### Slope

#### Intercept

Data conversions to and from Modbus are performed as follows:

$$\text{BACnetValue} = \text{ModbusValue} * \text{Slope} + \text{Intercept}$$

$$\text{ModbusValue} = (\text{BACnetValue} - \text{Intercept}) / \text{Slope}$$

### Units

The BACnet Units of the mapping.

### Min Present Value

The minimum BACnet writeable Present Value.

### Max Present Value

The maximum BACnet writeable Present Value.

*The following parameters can be setup if the Object Type is set to Binary Value:*

### Register Bit

The data bit in the register. On a write, the whole register will first be read, the bit updated as required, and then written back.

### Active State Text

The BACnet Active State Text of the mapping. It is recommended to keep this short to conserve memory.

### Inactive State Text

The BACnet Inactive State Text of the mapping. It is recommended to keep this short to conserve memory.

## BACNET RELIABILITY CODES

- 0 No fault detected
- 1 No sensor
- 2 Over range
- 3 Under range
- 4 Open loop
- 5 Shorted loop
- 6 No output
- 7 Unreliable other
- 8 Process error
- 9 Multi state fault
- 70 Modbus illegal function
- 71 Modbus illegal data address
- 72 Modbus illegal data value
- 73 Modbus slave device failure
- 74 Modbus acknowledge
- 75 Modbus slave device busy
- 76 Modbus parity error
- 77 Modbus gateway path unavailable
- 78 Modbus gateway target device failed to respond
- 79 Modbus receive error (CRC, Parity)
- 80 Modbus receive timeout
- 81 Modbus receive broadcast

## BACNET OUT OF SERVICE

Any mapping can be taken Out Of Service on BACnet to stop reading / writing to the Modbus device.