# Intesis®

**BY HMS NETWORKS**

# IN485MHI001R000 GATEWAY

## Mitsubishi Heavy Industries Residential and VRF Systems to BACnet MS/TP

**HMS**

# Table of Contents

# 1. Description, Compatible AC systems, and Order Codes

**BACnet MS/TP gateway for Mitsubishi Heavy Industries air conditioners.**

Compatible with residential and VRF air conditioning systems commercialized by Mitsubishi Heavy Industries.

Use the compatibility tool to get a complete list of compatible units: https://compatibility.intesis.com/

| ORDER CODE | LEGACY ORDER CODE |
| --- | --- |
| IN485MHI001R000 | INBACMHI001R100 |

# 2. General Information

## 2.1. Intended Use of the User Manual

This manual contains the main features of this Intesis gateway and the instructions for its appropriate installation, configuration, and operation.

Any person who installs, configures, or operates this gateway or any associated equipment should be aware of this manual's contents.

Keep this manual for future reference during the installation, configuration, and operation.

## 2.2. General Safety Information

> **!** **IMPORTANT**
> Follow these instructions carefully. Improper work may seriously harm your health and damage the gateway and/or any other equipment connected to it.

Only technical personnel, following these instructions and the country legislation for installing electrical equipment, can install and manipulate this gateway.

Install this gateway indoors, in a restricted access location, avoiding exposure to direct solar radiation, water, high relative humidity, or dust.

Preferably, mount this gateway on a DIN rail inside a grounded metallic cabinet, following the instructions in this manual.

If mounting on a wall, firmly fix this gateway on a non-vibrating surface, following the instructions in this manual.

All wires (for communication and power supply, if needed) must only be connected to networks with indoor wiring. All communication ports are considered for indoor use and must only be connected to SELV circuits.

Disconnect all systems from power before manipulating and connecting them to the gateway.

Use a circuit breaker between the gateway and the power supply. Rating: 250 V, 6 A.

Supply the correct voltage to power the gateway. The admitted range is detailed in the technical specifications table.

Respect the expected polarity of power and communication cables when connecting them to the gateway.

This Intesis gateway is designed for installation in an enclosure. When the device is mounted outside an enclosure, precautions should be taken to avoid electrostatic discharges to the unit in environments with static levels above 4 kV. When working in an enclosure (e.g., making adjustments, setting switches, etc.), typical anti-static precautions should be observed before touching the unit.

Binary inputs, if present, are potential-free contact. Do not connect any voltage.

Safety instructions in other languages can be found here.

## 2.3. Admonition Messages and Symbols

> **CAUTION**
> Instruction that must be followed to avoid a potentially hazardous situation that, if not avoided, could result in minor or moderate injury.

> **IMPORTANT**
> Instruction that must be followed to avoid a risk of reduced functionality and/or damage to the equipment or to avoid a network security risk.

> **NOTE**
> Additional information which may facilitate installation and/or operation.

> **TIP**
> Helpful advice and suggestions.

> **NOTICE**
> Remarkable Information.

# 3. Quickstart Guide for the IN485MHI001R000 Gateway

> ⚠️ **CAUTION**
> Disconnect all systems from power before connecting them to the gateway.

1. Mount the gateway in the desired installation site. This gateway can be mounted over a DIN rail or a wall. See details in Mounting (page 7).

> 📓 **NOTE**
> DIN rail mounting inside a grounded cabinet or metal enclosure is recommended.

2. Configure the gateway using the built-in DIP switches. See details in DIP switches (page 11).

3. Connect the gateway to the BACnet network via its **EIA-485** port.

4. Connect the gateway to the wired remote controller bus (XY). See details in Connection Procedure (page 8).

5. Check the communication performance between the BACnet bus and the AC system through the gateway's LED indicators. See details in LED Indicators (page 14).

6. The Intesis gateway is ready to be used in your system.

# 4. Overview

*Figure 1. Integration of Mitsubishi Heavy Industries AC units into a BACnet installation using the Intesis IN485MHI001R000 gateway*
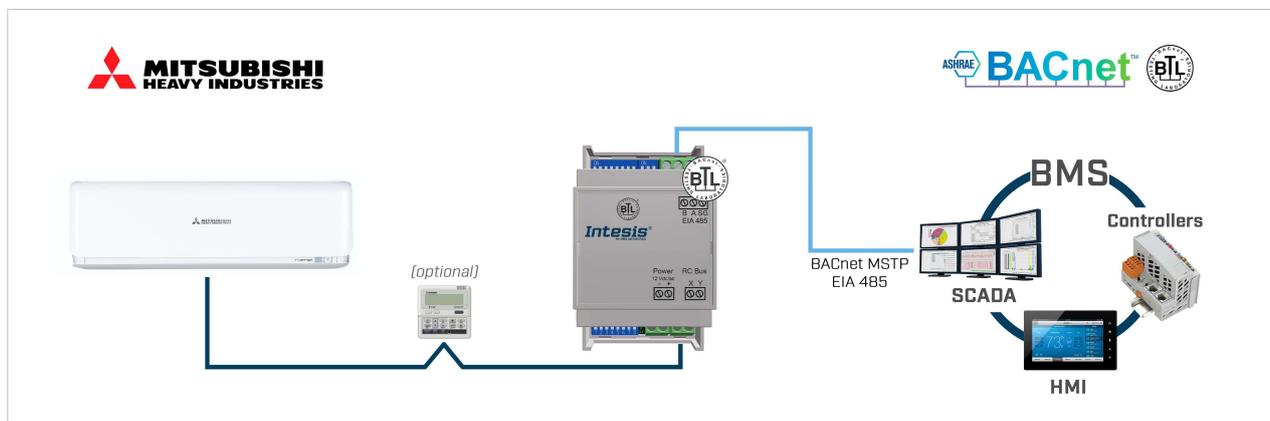


> **NOTE**
> This document assumes that the user is familiar with BACnet and Mitsubishi Heavy Industries technologies and their technical terms.

## 4.1. Inside the Package

Items included:

- Intesis IN485MHI001R000 gateway

- Installation guide

## 4.2. Main Features

- BTL mark ensures full interoperability with BACnet devices.

- Configuration with onboard DIP switches.

- Quick and easy installation: Set the DIP switches, plug, and play.

- External power supply not required.

- Simultaneous control of the AC unit via both the remote controller and the BMS (BACnet MS/TP).

- Reduced dimensions (HxWxD): 93 x 53 x 58 mm / 3.7 x 2.1 x 2.3".

- Mountable on DIN rail, wall, or even inside the indoor unit in some models of AC.

- Significant reduction of the HVAC system energy consumption.

- Three-year warranty.

## 4.3. Gateway Capacity

This Intesis gateway can integrate one single Mitsubishi Heavy Industries AC indoor unit and its associated elements.

## 4.4. General Functionality

With this Intesis IN485MHI001R000 gateway, you can easily integrate Mitsubishi Heavy Industries residential and VRF air conditioning systems into a system based on BACnet MS/TP. To do so, the gateway acts as a server device of the installation itself, accessing all signals from the AC indoor unit.

The gateway is continuously polling the AC unit, storing in its memory the current status of every signal you want to track and serving this data to the control system when requested. The gateway also sends the requested commands to the indoor unit.

# 5. Hardware

## 5.1. Mounting

Mount the gateway over a wall or over a DIN rail.

> **!** **IMPORTANT**
> Do not mount the gateway in air-handling units or conducts.

> **NOTE**
> DIN rail mounting inside a grounded metallic cabinet is recommended.

**Wall mounting**

1. Press the rear panel clips outwards until you hear a *click*.

2. Use the clip holes to screw the gateway to the wall.

3. Make sure the gateway is firmly fixed.
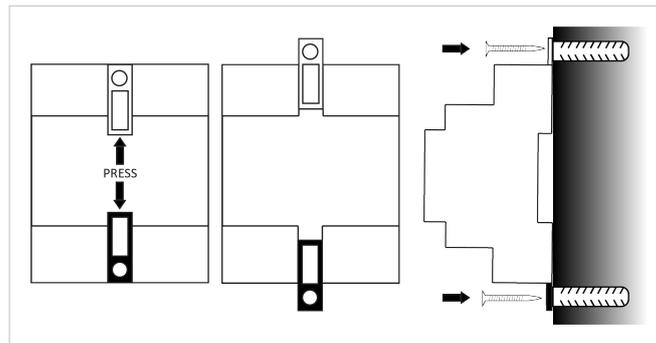


**DIN rail mounting**

Keep the clips in its original position.

1. Fit the gateway's top-side clip in the upper edge of the DIN rail.

2. Press the low side of the gateway gently to lock it in the DIN rail.

3. Make sure the gateway is firmly fixed.

> **NOTE**
> For some DIN rails, to complete step 2, you may need a small screwdriver or similar to pull the bottom clip down.
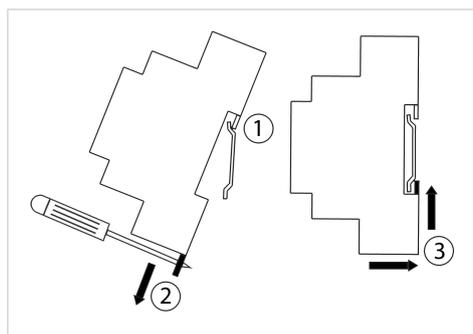
## 5.2. Connection Procedure

> ⚠️ **CAUTION**
> Disconnect all systems from power before connecting them to the gateway.

> ❗ **IMPORTANT**
> Keep communication cables as far away as possible from power and ground cables. Never bundle them together.

> 📝 **NOTE**
> Mount the gateway in the desired place before wiring it.

*Figure 2. Wiring diagram*
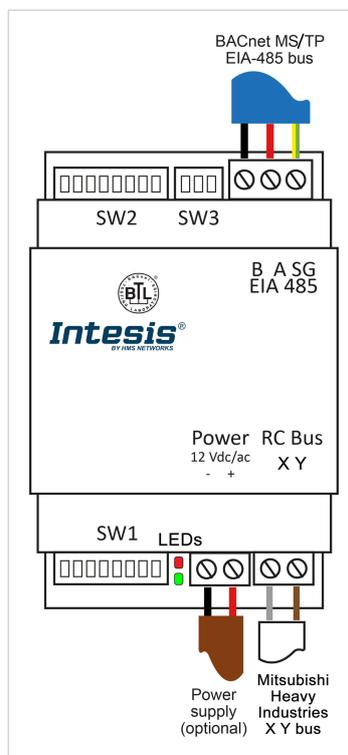


**Connection to the wired remote controller (RC) bus**

> 📝 **NOTE**
> Mount the gateway in the desired place before wiring it.

1.  Connect the gateway at any point of the XY bus.

    > 📝 **NOTE**
    > - The XY bus is a two-wire bus used to connect the indoor unit and a wired RC .
    > - This connection has no specific polarity.
    > - The maximum length for the RC bus is 600 m (1968 ft).

**Connection to the BACnet MS/TP bus**

2.   Connect the BACnet MS/TP bus to the EIA-485 port of the gateway.

> **(!) IMPORTANT**
> Observe polarity: B-, A+, and SG for signal ground.

> **(!) IMPORTANT**
> - EIA-485 bus doesn't allow loop or star topologies.
> - Maximum length for the EIA-485 bus is 1200 meters (4000 feet).

> **(!) EIA-485 BUS. TERMINATION RESISTORS AND FAIL-SAFE BIASING MECHANISM**
> The EIA-485 bus requires a 120Ω terminator resistor at each end of the bus to avoid signal reflections.
>
> In order to prevent fail status detections by the receivers, which are "listening" to the bus, when all the transmitters' outputs are in three-state (high impedance), a fail-safe biasing mechanism is required. This mechanism provides a safe status (a correct voltage level) in the bus when all the transmitters' outputs are in three-state.
>
> The IN485MHI001R000 gateway includes an on-board terminator resistor of 120Ω that can be connected to the EIA-485 bus by using DIP switch SW3.
>
> - **SW3, Position 1:**
>   ON: 120 Ω termination active.
>   OFF: 120 Ω termination inactive (default position).
>
> Some BACnet MS/TP EIA-485 Master devices can provide also internal 120Ω terminator resistor and/or fail-safe biasing. Consult the technical documentation of the Master device connected to the EIA-485 network in each case.
>
> If the termination resistor is enabled and you install the gateway at one of the ends of the bus, do not install an additional termination resistor at that end.

3.   Reconnect all systems to power.

## 5.3. Coexistence of the Gateway with a Remote Controller

If there is a wired remote controller (RC) connected to the indoor unit, you must set the roles for both the wired RC and the gateway:

- Set the wired RC as the header[1] and the gateway as the follower[1].

- Set the wired RC as the follower and the gateway as the header.

> **NOTE**
> [1] We use the terms header and follower to designate the roles also known as master and slave or main and sub.

> **NOTE**
> By default, the factory settings designate the gateway as the follower. Use the SW1-1 (DIP switch 1, position 1) to configure the gateway's role. See DIP switches (page 11).

> **NOTE**
> Refer to the documentation provided with the Mitsubishi Heavy Industries indoor unit to know the needed procedure to set the role of the wired RC.

> **IMPORTANT**
> If no wired RC is present in the installation, set the gateway as the header.

> **NOTE**
> Although it is not mandatory, we recommend connecting a wired RC in the bus since it may be necessary to establish proper communication between the gateway and some indoor units.

> **IMPORTANT**
> The roles of the gateway and the wired RC affect certain functions, such as using a temperature sensor from the BMS side or the Virtual Temperature function. To know more, see Ambient Temperature and Virtual Temperature Function (page 56).

## 5.4. Connection to an External Power Supply

> **IMPORTANT**
> In most cases, this gateway is powered through the remote controller bus itself, and there is no need to connect an external power supply. However, depending on the number and type of remote controllers installed, the bus could not supply the needed power.

> **TIP**
> Some signs indicating there is not enough power in the bus may include, for example, a malfunction of the remote controllers' display or in its performance.

If that's the case, connect a 12 VDC/AC SELV-rated NEC class 2 or Limited Power Source (LPS) power supply in the gateway's Power connector.

> **IMPORTANT**
> Respect polarity.

*Figure 3. Power connector*
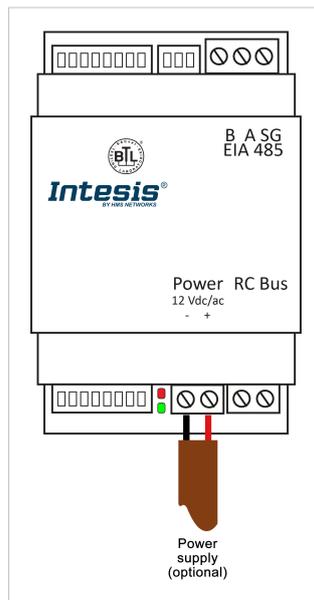


## 5.5. DIP switches

The gateway includes three DIP switches: SW1 (8 switches) at the bottom and SW2 (8 switches) and SW3 (3 switches) at the top.

> **IMPORTANT**
> The DIP switches configuration will only take effect after rebooting the gateway.
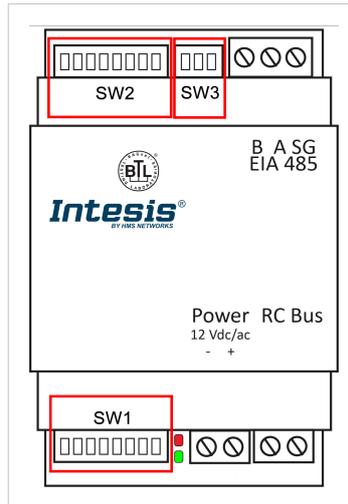
*Table 1. SW1 (P1): Gateway configuration; (P6 to P8): BACnet MS/TP baudrate*

| Binary value | Switches | | | | | | | | Description |
|---|---|---|---|---|---|---|---|---|---|
| b0 .. b7 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
| 0 X X X X X X X | OFF | X | X | X | X | X | X | X | Follower in XY bus (default) |
| 1 X X X X X X X | ON | X | X | X | X | X | X | X | Header in XY bus |
| X X X X X 0 0 0 | X | X | X | X | X | OFF | OFF | OFF | Autobaudrate (default value) |
| X X X X X 1 0 0 | X | X | X | X | X | ON | OFF | OFF | 9600 bps |
| X X X X X 0 1 0 | X | X | X | X | X | OFF | ON | OFF | 19200 bps |
| X X X X X 1 1 0 | X | X | X | X | X | ON | ON | OFF | 38400 bps |
| X X X X X 0 0 1 | X | X | X | X | X | OFF | OFF | ON | 57600 bps |
| X X X X X 1 0 1 | X | X | X | X | X | ON | OFF | ON | 76800 bps |
| X X X X X 0 1 1 | X | X | X | X | X | OFF | ON | ON | 115200 bps |
| X X X X X 1 1 1 | X | X | X | X | X | ON | ON | ON | Autobaudrate |

**NOTE**

If **Autobaudrate** is selected, the gateway will scan the network to find any other BACnet MS/TP device and will match its baudrate. Once detected, the baudrate will only be modified after a reset/reboot of the gateway.

*Table 2. SW2 (P1 to P7): BACnet MS/TP MAC address; (P8): Temperature unit (°C/°F)*

| Binary value | Switches | | | | | | | | MAC address | Description |
|---|---|---|---|---|---|---|---|---|---|---|
| b0 .. b7 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | |
| 0 0 0 0 0 0 0 X | OFF | OFF | OFF | OFF | OFF | OFF | OFF | X | 0 | - |
| 1 0 0 0 0 0 0 X | ON | OFF | OFF | OFF | OFF | OFF | OFF | X | 1 | - |
| 0 1 0 0 0 0 0 X | OFF | ON | OFF | OFF | OFF | OFF | OFF | X | 2 | - |
| 1 1 0 0 0 0 0 X | ON | ON | OFF | OFF | OFF | OFF | OFF | X | 3 | - |
| ... | ... | | | | | | ... | | ... | - |
| 1 0 1 1 1 1 1 X | ON | OFF | ON | ON | ON | ON | ON | X | 125 | - |
| 0 1 1 1 1 1 1 X | OFF | ON | ON | ON | ON | ON | ON | X | 126 | - |
| 1 1 1 1 1 1 1 X | ON | ON | ON | ON | ON | ON | ON | X | 127 | - |
| X X X X X X X 0 | X | X | X | X | X | X | X | OFF | - | Temperature values in BACnet are represented in Celsius degrees (default value) |
| X X X X X X X 1 | X | X | X | X | X | X | X | ON | - | Temperature values in BACnet are represented in Fahrenheit degrees |

**NOTE**

By default, the **Device instance** base is 246000. Setting the SW2 switches, you can add from 0 to 127, so the final Device instance address can be from 246000 to 246127.

**IMPORTANT**

If you overwrite the **Device instance** object from the BMS side (for example, using the control terminal), this function to set up the **Device instance** with the SW2 switches will be deactivated. To activate this function again, a factory reset is needed.

*Table 3. SW3 (P1 to P3): BACnet polarization and termination resistor*

| Binary value | Switches | | | Description |
|---|---|---|---|---|
| b0 .. b2 | 1 | 2 | 3 | |
| 0 X X | OFF | X | X | EIA-485 bus without termination resistor. The gateway is not at one end of the EIA-485 bus (default value) |
| 1 X X | ON | X | X | 120 Ω termination resistor active. The gateway is at one end of the EIA-485 bus |
| X 0 0 | X | OFF | OFF | No bus polarization (default value) |
| X 1 1 | X | ON | ON | Bus polarization active |

# 5.6. LED Indicators

There are two LEDs at the lower side of the gateway, between the DIP switch block SW1 and the **Power** connector.
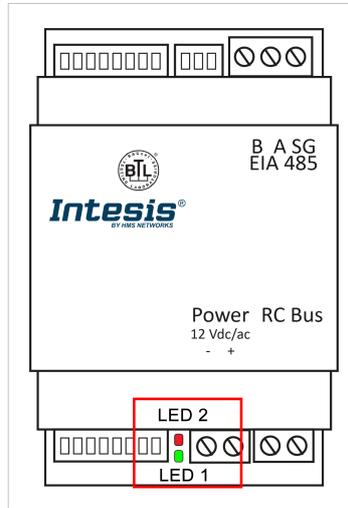


When powering the gateway up, both LEDs blink once and then turn off. After that, LEDs will behave as described in the table below:

*Table 4. LED status table*

| LED | Status | Description |
|-----|--------|-------------|
| L1 Green BACnet | ON | BACnet MS/TP link performed |
| | Flickering | Activity on the BACnet MS/TP bus |
| | OFF | BACnet MS/TP link not performed |
| L2 Red AC unit | ON | AC communication error |
| | Blinking | AC unit error |
| | Flashing | AC communication OK |

**LED PATTERNS**

- **ON**: 100% on

- **Flickering**: irregular cycle (90% on - 10% off approx)

- **Blinking**: 50% on - 50% off

- **Flashing**: 10% on - 90% off

- **OFF**: 100% off

## 5.7. Technical Specifications

| | |
|---|---|
| **Housing** | Plastic, type PC (UL 94 V-0) |
| | Net dimensions (HxWxD): 93 x 53 x 58 mm / 3.7 x 2.1 x 2.3" |
| | Color: Light grey. RAL 7035 |
| **Weight** | 85 g (3 oz) |
| **Terminal wiring** | Wire cross-section/gauge per terminal: |
| | One core: 0.2 .. 2.5 mm$^2$ (24 .. 11 AWG) |
| | Two cores: 0.2 .. 1.5mm$^2$ (24 .. 15 AWG) |
| | Three cores: Not permitted |
| | Use solid or stranded wires (twisted or with ferrule). |
| **External power supply (optional)** | SELV-rated NEC class 2 or limited power source (LPS) power supply |
| | 12 VDC/AC; 0.1 A |
| **Mounting** | Wall, DIN rail, or inside the indoor unit |
| **EIA-485 port** **BACnet MS/TP** | 1 x pluggable terminal block (3 poles: B, A, and SG) |
| **AC unit port** | 1 x RC bus pluggable terminal block (2 poles) |
| **LED indicators** | 2 x Communication status |
| **DIP switches** | SW1: Gateway and baudrate configuration |
| | SW2: MAC address and temperature unit |
| | SW3: Bus polarization and termination |
| **Operational and storage temperature** | Celsius: Op: 0 to +70°C; St: -20 to 85°C |
| | Fahrenheit: Op: 32 to 158°F; St: -4 to 185°F |
| **Operational and storage humidity** | 5% to 95% RH non-condensing |
| **Isolation Voltage** | 1500 VDC |
| **Isolation resistance** | 1000 MΩ |

## 5.8. Dimensions

**Net dimensions (HxWxD):**

Millimeters: 93 x 53 x 58 mm

Inches: 3.66 x 2.08 x 2.28"

> **❗ IMPORTANT**
> Leave enough clear space to wire the gateway easily and for the subsequent manipulation of elements such as connectors, DIP switches, etc.
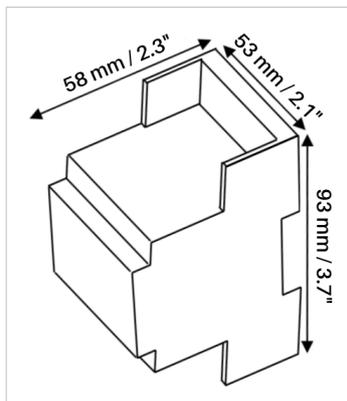
# 6. Restore the Factory Settings

To restore the gateway to its factory settings, proceed as follows:

1. Set all switches from DIP switches SW1 and SW2 to **ON**.

2. Reboot the gateway:

   a. Power it OFF.

   b. Power it ON.

   > **NOTE**
   > To turn the gateway OFF, disconnect it from the AC unit and from the power supply, if there is one. To turn the gateway ON, reestablish those connections.

3. After the reboot, LEDs will blink with the SOS Morse sequence:

   a. Three short blinks

   b. Three longer blinks

   c. Three short blinks

4. Set all switches from DIP switches SW1 and SW2 to **OFF** within 30 seconds.

   > **IMPORTANT**
   > If you take longer than 30 seconds, all LEDs will turn off, meaning that the procedure has failed. To retry it, go to step 1 and begin the procedure again.

After this procedure, the LED will flash red, meaning that the gateway has been restored to the factory settings.

# 7. Objects

## 7.1. Supported Object Types

| Object type | ID |
|---|---|
| Analog-Input | 0 |
| Analog-Output | 1 |
| Analog-Value | 2 |
| Binary-Input | 3 |
| Binary-Output | 4 |
| Binary-Value | 5 |
| Device | 8 |
| Multistate-Input | 13 |
| Multistate-Output | 14 |
| Multistate-Value | 19 |

## 7.2. Member Objects

### 7.2.1. Type: Gateway

| Object name | Description | Object type | Object instance |
|---|---|---|---|
| IN485MHI001R000 | Mitsubishi Heavy Industries RC gateway | 8: Device | 246000 (default) |

### 7.2.2. Type: Indoor Unit

| Object name | Object type | Object instance |
|---|---|---|
| OnOff_status | BI | 0 |
| OnOff_command | BO | 0 |
| Mode_status | MI | 0 |
| Mode_command | MO | 0 |
| Setpoint_status | AI | 0 |
| User_Setpoint_status | AI | 17 |
| Setpoint_command | AO | 0 |
| VirtualTempActive | BI | 14 |
| FanSpeed_status | MI | 1 |
| FanSpeed_command | MO | 1 |
| AirDirectionUD_status | MI | 2 |
| AirDirectionUD_command | MO | 2 |
| RoomTemperature_status | AI | 1 |
| RoomTemperature_command | AO | 1 |
| ErrorCode | AI | 2 |
| ErrorCodeM | MI | 4 |
| ErrorActive | BI | 1 |
| ErrorAddress | AI | 2 |
| ErrorReset | BO | 5 |
| OnTimeCounter | AV | 0 |
| FilterSign | BI | 6 |
| FilterReset | BO | 4 |

| Object name | Object type | Object instance |
|---|---|---|
| Occupancy | MV | 0 |
| OccupiedCoolSetpoint | AV | 1 |
| OccupiedHeatSetpoint | AV | 2 |
| UnoccupiedCoolSetpoint | AV | 3 |
| UnoccupiedHeatSetpoint | AV | 4 |
| OccupancyContinuousCheck | BV | 0 |
| UnoccupiedDeadBandAction | BV | 1 |
| LockRemoteControl | BV | 2 |
| OutdoorTemp_status | AI | 8 |
| CenterRemote | MI | 9 |
| DIP_SW_S1_status | AI | 9 |
| DIP_SW_S2_status | AI | 10 |
| SerialNumber | AI | 11 |

# 7.3. Objects and Properties

## 7.3.1. Mitsubishi Heavy Industries AC Gateway (Device Object Type)

**Object_Identifier**: The gateway can be identified in the BACnet network either manually or automatically:

- **Automatic address (default)**: The gateway address is generated by combining the base address + the local address configured in the DIP switch block SW2 (P1 to P7). See DIP switches (page 11).

- **Manual address**: The gateway will switch to manual addressing mode when this property receives a value from the BACnet side.

> **❗ IMPORTANT**
> During the manual addressing mode, the address configured with the DIP switch SW2 is not considered.

> **❗ IMPORTANT**
> If **Object_Identifier** is overwritten from BACnet, the DIP switch SW2 configuration will not be considered for the device instance calculation until a **Restore factory settings** is performed. See details in Restore the Factory Settings (page 16).

The **Object_Identifier**, **Object_Name**, and **Description** properties can be edited by writing on their **Value**.

> **🗒 NOTE**
> The **Description** value permits up to 63 characters.

| Property Identifier | Property Datatype | Value | ASHRAE | Intesis |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | Device, 246000 (default value) | R | W |
| Object_Name | CharacterString | IN485MHI001R000 | R | W |
| Object_Type | BACnetObjectType | DEVICE (8) (Device Object Type) | R | R |
| System_Status | BACnetDeviceStatus | OPERATIONAL (0) | R | R |
| Vendor_Name | CharacterString | HMS Industrial Networks SLU | R | R |
| Vendor_Identifier | Unsigned16 | 246 | R | R |
| Model_Name | CharacterString | IN485MHI001R000 | R | R |
| Firmware_Revision | CharacterString | 1.0.0.0 | R | R |
| Application_Software_ Version | CharacterString | 1.0.0.0 | R | R |
| Location | CharacterString | "" | O | - |

| Property Identifier | Property Datatype | Value | ASHRAE | Intesis |
|---|---|---|---|---|
| Description | CharacterString | Mitsubishi Heavy Industries AC interface | O | W |
| Protocol_Version | Unsigned | 1 | R | R |
| Protocol_Revision | Unsigned | 12 | R | R |
| Protocol_Services_ Supported | BACnetServiceSupported | Refer to the PICS document | R | R |
| Protocol_Object_Types_ Supported | BACnetObjectTypes Supported | Refer to section Objects and Properties (page 18) | R | R |
| Object_List | BACnetArray[N] of BACnetObjectIdentifier | BACnetARRAY[N] | R | R |
| Structured_Object_List | BACnetArray[N] of BACnetObjectIdentifier | - | O | - |
| Max_APDU_Length_ Accepted | Unsigned | 480 for BACnet MS/TP 1476 for BACnet/IP | R | R |
| Segmentation_Supported | BACnetSegmentation | SEGMENTED-BOTH (0) | R | R |
| Max_Segments_accepted | Unsigned | 16 | O | R |
| VT_Classes_Supported | List of BACnetVTClass | - | O | - |
| Active_VT_Sessions | List of BACnetVTSession | - | O | - |
| Local_Date | Date | - | O | - |
| Local_Time | Time | - | O | - |
| UTC_Offset | INTEGER | - | O | - |
| Daylight_Savings_Status | BOOLEAN | - | O | - |
| APDU_Segment_Timeout | Unsigned | 3000 | R | R |
| APDU_Timeout | Unsigned | 3000 | R | R |
| Number_of_APDU_ Retries | Unsigned | 3 | R | R |
| List_Of_Session_Keys | List of BACnetSessionKey | - | O | - |
| Time_Synchronization_ Recipients | List of BACnetRecipient | - | O | - |
| Max_Master | Unsigned | 127 | R | W |
| Max_Info_Frames | Unsigned | 1 | O | R |
| Device_Address_Binding | List of BACnetAddressBinding | NULL (empty) | R | R |
| Database_Revision | Unsigned | 0 | R | R |
| Configuration_Files | BACnetArray[N] of BACnetObjectIdentifier | - | O | - |
| Last_Restore_Time | BACnetTimeStamp | - | O | - |
| Backup_Failure_Timeout | Unsigned16 | - | O | - |
| Active_COV_ Subscriptions | List of BACnetCOVSubscription | List of BACnetCOVSubscription | O | R |
| Slave_Proxy_Enable | BACnetArray[N] of BOOLEAN | - | O | - |
| Manual_Slave_Address_ Binding | List of BACnetAddressBinding | - | O | - |
| Auto_Slave_Discovery | BACnetArray[N] of BOOLEAN | - | O | - |
| Slave_Address_Binding | BACnetAddressBinding | - | O | - |
| Last_Restart_Reason | BACnetRestartReason | - | O | - |
| Time_Of_Device_Restart | BACnetTimeStamp | - | O | - |
| Restart_Notification_ Recipients | List of BACnetRecipient | - | O | - |
| UTC_Time_ Synchronization_ Recipients | List of BACnetRecipient | - | O | - |
| Time_Synchronization_ Interval | Unsigned | - | O | - |
| Align_Intervals | BOOLEAN | - | O | - |
| Interval_Offset | Unsigned | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.2. OnOff_status (Binary Input Object Type)

It indicates if the indoor unit is turned on or off.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Input, 0) | R | R |
| Object_Name | CharacterString | OnOff_status | R | R |
| Object_Type | BACnetObjectType | BINARY_INPUT (3) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0), UNRELIABLE_OTHER (7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Polarity | BACnetPolarity | NORMAL (0) | R | R |
| Inactive_Text | CharacterString | Off | O | R |
| Active_Text | CharacterString | On | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

### 7.3.3. OnOff_command (Binary Output Object Type)

It turns the indoor unit on or off.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Output, 0) | R | R |
| Object_Name | CharacterString | OnOff_command | R | R |
| Object_Type | BACnetObjectType | BINARY_OUTPUT (4) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | R | W |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Polarity | BACnetPolarity | NORMAL (0) | R | R |
| Inactive_Text | CharacterString | Off | O | R |
| Active_Text | CharacterString | On | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Minimum_Off_Time | Unsigned32 | - | O | - |
| Minimum_On_Time | Unsigned32 | - | O | - |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | R |
| Relinquish_Default | BACnetBinaryPV | INACTIVE (0) | R | R |
| Time Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Feedback_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.4. Mode_status (Multistate Input Object Type)

It indicates the indoor unit's current mode.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Multistate Input, 0) | R | R |
| Object_Name | CharacterString | Mode_status | R | R |
| Object_Type | BACnetObjectType | MULTISTATE_INPUT (13) | R | R |
| Present_Value | Unsigned | 1 .. 5 | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0), UNRELIABLE_OTHER(7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Number_Of_States | Unsigned | 5 | R | R |
| State_Text | BACnetArray[N] of CharacterString | Check the **Mode status table** below. | O | R |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Values | List of Unsigned | - | O | - |
| Fault_Values | List of Unsigned | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

*Table 5. Mode status*

| Present_Value | State_Text |
|---|---|
| 1 | Heat |
| 2 | Cool |
| 3 | Fan |
| 4 | Dry |
| 5 | Auto |
| 6 | AutoHeat |
| 7 | AutoCool |

> **NOTE**
> When setting the control mode in Auto, the AC unit itself decides the appropriate mode.

## 7.3.5. Mode_command (Multistate Output Object Type)

It sets the AC indoor unit's mode.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Multistate Output,0) | R | R |
| Object_Name | CharacterString | Mode_command | R | R |
| Object_Type | BACnetObjectType | MULTISTATE_OUTPUT (14) | R | R |
| Present_Value | Unsigned | 1 .. 5 | R | W |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Number_Of_States | Unsigned | 5 | R | R |
| State_Text | BACnetArray[N] of CharacterString | Check the **Mode command table** below | O | R |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | R |
| Relinquish_Default | Unsigned | 1 | R | R |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Feedback_Value | Unsigned | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

*Table 6. Mode command*

| Present_Value | State_Text |
|---|---|
| 1 | Heat |
| 2 | Cool |
| 3 | Fan |
| 4 | Dry |
| 5 | Auto |

> **NOTE**
> When setting the control mode in Auto, the AC unit itself decides the appropriate mode.

## 7.3.6. Setpoint_status (Analog Input Object Type)

It reports the temperature setpoint requested to the indoor unit.

> **NOTE**
> To know more, see Considerations on Temperature Signals (page 60).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Input, 0) | R | R |
| Object_Name | CharacterString | SetPoint_status | R | R |
| Object_Type | BACnetObjectType | ANALOG_INPUT (0) | R | R |
| Present_Value | REAL | 16 .. 30°C / 61 .. 86°F | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0), UNRELIABLE_OTHER (7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Celsius degrees (62) Fahrenheit degrees (64) | R | R |
| Min_Pres_Value | REAL | 16°C / 61°F | O | R |
| Max_Pres_Value | REAL | 30°C / 86°F | O | R |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTE**
> You can set the temperature scale in Celsius or Fahrenheit via the DIP switches. More information in DIP switches (page 11).

## 7.3.7. Setpoint_command (Analog Output Object Type)

It is used to request a temperature setpoint from the BACnet side.

> **NOTE**
> To know more, see Considerations on Temperature Signals (page 60).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Output, 0) | R | R |
| Object_Name | CharacterString | SetPoint_command | R | R |
| Object_Type | BACnetObjectType | ANALOG_OUTPUT (1) | R | R |
| Present_Value | REAL | 16 .. 30°C / 61 .. 86°F | W | W |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Celsius degrees (62) <br> Fahrenheit degrees (64) | R | R |
| Min_Pres_Value | REAL | 16°C / 61°F | O | R |
| Max_Pres_Value | REAL | 30°C / 86°F | O | R |
| Resolution | R | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | R |
| Relinquish_Default | Unsigned | 22 | R | R |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTE**
> You can set the temperature scale in Celsius or Fahrenheit via the DIP switches. More information in DIP switches (page 11).

## 7.3.8. VirtualTempActive (Binary Input Object Type)

It indicates if the Virtual Temperature function is active or inactive.

> **NOTE**
>
> The Virtual Temperature function allows the gateway to set the reference temperature using the value reported by a sensor connected to the BMS.
>
> For more information, see Ambient Temperature and Virtual Temperature Function (page 56).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Input, 14) | R | R |
| Object_Name | CharacterString | VirtualTempActive | R | R |
| Object_Type | BACnetObjectType | BINARY_INPUT (3) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) UNRELIABLE_OTHER (7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Polarity | BACnetPolarity | NORMAL (0) | R | R |
| Inactive_Text | CharacterString | No | O | R |
| Active_Text | CharacterString | Yes | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.9. FanSpeed_status (Multistate Input Object Type)

It indicates the indoor unit's fan speed.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Multistate Input, 1) | R | R |
| Object_Name | CharacterString | FanSpeed_status | R | R |
| Object_Type | BACnetObjectType | MULTISTATE_INPUT (13) | R | R |
| Present_Value | Unsigned | 1 .. 5 | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0), UNRELIABLE_OTHER (7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE/TRUE | R | R |
| Number_Of_States | Unsigned | 5 | R | R |
| State_Text | BACnetArray[N] of CharacterString | Check the **Fan speed status table** below. | O | R |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Values | List of Unsigned | - | O | - |
| Fault_Values | List of Unsigned | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

*Table 7. Fan speed status*

| Present_Value | State_Text |
|---|---|
| 1 | Auto[1] |
| 2 | Fan Speed 1 |
| 3 | Fan Speed 2 |
| 4 | Fan Speed 3 |
| 5 | Fan Speed 4 |

[1]Older indoor units may lack this position.

## 7.3.10. FanSpeed_command (Multistate Output Object Type)

It sets the indoor unit's fan speed.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Multistate Output, 1) | R | R |
| Object_Name | CharacterString | FanSpeed_command | R | R |
| Object_Type | BACnetObjectType | MULTISTATE_OUTPUT (14) | R | R |
| Present_Value | Unsigned | 1 .. 5 | R | W |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Number_Of_States | Unsigned | 5 | R | R |
| State_Text | BACnetArray[N] of CharacterString | Check the **Fan speed command table** below | O | R |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | R |
| Relinquish_Default | Unsigned | 1 | R | R |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Feedback_Value | Unsigned | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

*Table 8. Fan speed command*

| Present_Value | State_Text |
|---|---|
| 1 | Auto[1] |
| 2 | Fan Speed 1 |
| 3 | Fan Speed 2 |
| 4 | Fan Speed 3 |
| 5 | Fan Speed 4 |

[1]Older indoor units may lack this position.

### 7.3.11. AirDirectionUD_status (Multistate Input Object Type)

It indicates the indoor unit's vertical air direction (up-down) status.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Multistate Input, 2) | R | R |
| Object_Name | CharacterString | AirDirectionUD_status | R | R |
| Object_Type | BACnetObjectType | MULTISTATE_INPUT(13) | R | R |
| Present_Value | Unsigned | 1 .. 5 | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0), UNRELIABLE_OTHER (7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE/TRUE | R | R |
| Number_Of_States | Unsigned | 5 | R | R |
| State_Text | BACnetArray[N] of CharacterString | Check the **Air direction status table** below. | O | R |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Values | List of Unsigned | - | O | - |
| Fault_Values | List of Unsigned | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

*Table 9. Air direction up-down status*

| Present_Value | State_Text |
|---|---|
| 1 | Pos1 |
| 2 | Pos2 |
| 3 | Pos3 |
| 4 | Pos4 |
| 5 | Swing |

## 7.3.12. AirDirectionUD_command (Multistate Output Object Type)

It sets the indoor unit's vertical air direction (up-down).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Multistate Output, 2) | R | R |
| Object_Name | CharacterString | AirDirectionUD_command | R | R |
| Object_Type | BACnetObjectType | MULTISTATE_OUTPUT (14) | R | R |
| Present_Value | Unsigned | 1 .. 5 | R | W |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Number_Of_States | Unsigned | 5 | R | R |
| State_Text | BACnetArray[N] of CharacterString | Check the **Air direction command table** below | O | R |
| Priority_Array | BACnetPriorityArray | - | R | R |
| Relinquish_Default | Unsigned | - | R | R |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Feedback_Value | Unsigned | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

*Table 10. Air direction up-down command*

| Present_Value | State_Text |
|---|---|
| 1 | Pos1 |
| 2 | Pos2 |
| 3 | Pos3 |
| 4 | Pos4 |
| 5 | Swing |

## 7.3.13. RoomTemperature_status (Analog Input Object Type)

It reports the ambient temperature measured by the sensor from the AC system side. It can be reported by the indoor unit or the wired remote controller's sensor.

> **NOTE**
> To know more, see Ambient Temperature and Virtual Temperature Function (page 56).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Input, 1) | R | R |
| Object_Name | CharacterString | RoomTemperature_status | R | R |
| Object_Type | BACnetObjectType | ANALOG_INPUT (0) | R | R |
| Present_Value | REAL | -10 .. 50°C / 14 .. 122°F | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0), UNRELIABLE_OTHER (7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Celsius degrees (62) Fahrenheit degrees (64) | R | R |
| Min_Pres_Value | REAL | -10°C / 14°F | O | - |
| Max_Pres_Value | REAL | 50°C / 122°F | O | - |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTE**
> You can set the temperature scale in Celsius or Fahrenheit scale via DIP switches. More information in DIP switches (page 11).

## 7.3.14. RoomTemperature_command (Analog Output Object Type)

It is used to write the ambient temperature measured by a sensor from the BACnet side.

> **NOTE**
> To know more, see Considerations on Temperature Signals (page 60).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Output, 1) | R | R |
| Object_Name | CharacterString | RoomTemperature_command | R | R |
| Object_Type | BACnetObjectType | ANALOG_OUTPUT (1) | R | R |
| Present_Value | REAL | -10 .. 50°C / 14 .. 122°F | R | W |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Celsius degrees (62) <br> Fahrenheit degrees (64) | R | R |
| Min_Pres_Value | REAL | -10°C / 14°F | O | - |
| Max_Pres_Value | REAL | 50°C / 122°F | O | - |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | R |
| Relinquish_Default | Unsigned | -32768 | R | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTE**
> You can set the temperature scale in Celsius or Fahrenheit scale via DIP switches. More information in DIP switches (page 11).

## 7.3.15. ErrorCode (Analog Input Object Type)

It indicates the AC system's current error.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Input, 2) | R | R |
| Object_Name | CharacterString | ErrorCode | R | R |
| Object_Type | BACnetObjectType | ANALOG_INPUT (0) | R | R |
| Present_Value | REAL | 0 .. 90, 65535 | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | 300 | O | - |
| Units | BACnetEngineeringUnits | NO_UNITS (95) | R | R |
| Min_Pres_Value | REAL | 0 | O | - |
| Max_Pres_Value | REAL | 65535 | O | - |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTICE**
>
> For more information on each error code, see Error Codes (page 62).

## 7.3.16. ErrorCodeM (Multistate Input Object Type)

It indicates the AC system's current error.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Multistate Input, 4) | R | R |
| Object_Name | CharacterString | ErrorCodeM | R | R |
| Object_Type | BACnetObjectType | MULTISTATE_INPUT(13) | R | R |
| Present_Value | Unsigned | 0 .. 90, 65535 | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Number_Of_States | Unsigned | 92 | R | R |
| State_Text | BACnetArray[N] of CharacterString | Check Error Codes (page 62) | O | R |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Values | List of Unsigned | - | O | - |
| Fault_Values | List of Unsigned | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.17. ErrorActive (Binary Input Object Type)

It indicates if there is an active error in the AC system.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Input, 1) | R | R |
| Object_Name | CharacterString | ErrorActive | R | R |
| Object_Type | BACnetObjectType | BINARY_INPUT (3) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Polarity | BACnetPolarity | NORMAL (0) | R | R |
| Inactive_Text | CharacterString | No | O | R |
| Active_Text | CharacterString | Error | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.18. ErrorAddress (Analog Input Object Type)

It indicates the address of the indoor unit which is reporting the error.

| Property Identifier | Property Datatype | Value | ASHRAE | Intesis |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Input, 4) | R | R |
| Object_Name | CharacterString | ErrorAddress | R | R |
| Object_Type | BACnetObjectType | ANALOG_INPUT (0) | R | R |
| Present_Value | REAL | - | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) UNRELIABLE_OTHER (7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE/TRUE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | NO_UNITS (95) | R | R |
| Min_Pres_Value | REAL | - | O | - |
| Max_Pres_Value | REAL | - | O | - |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.19. ErrorReset (Binary Output Object Type)

It resets the error signal.

| Property Identifier | Property Datatype | Value | ASHRAE | Intesis |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Output, 5) | R | R |
| Object_Name | CharacterString | ErrorReset | R | R |
| Object_Type | BACnetObjectType | BINARY_OUTPUT (4) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | R | W |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Polarity | BACnetPolarity | NORMAL (0) | R | R |
| Inactive_Text | CharacterString | Normal | O | R |
| Active_Text | CharacterString | Reset | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Minimum_Off_Time | Unsigned32 | - | O | - |
| Minimum_On_Time | Unsigned32 | - | O | - |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | R |
| Relinquish_Default | BACnetBinaryPV | INACTIVE (0) | R | R |
| Time Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Feedback_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.20. OnTimeCounter (Analog Value Object Type)

It indicates the AC unit running time.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Value, 0) | R | R |
| Object_Name | CharacterString | OnTimeCounter | R | R |
| Object_Type | BACnetObjectType | ANALOG_VALUE (2) | R | R |
| Present_Value | REAL | 0 .. 65535 | R | R/W |
| Description | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Hours (71) | R | R |
| Min_Pres_Value | REAL | 0 | O | - |
| Max_Pres_Value | REAL | 65535 | O | - |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.21. FilterSign (Binary Input Object Type)

It indicates the status of the filter.

| Property Identifier | Property Datatype | Value | ASHRAE | Intesis |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Input, 1) | R | R |
| Object_Name | CharacterString | FilterSign | R | R |
| Object_Type | BACnetObjectType | BINARY_OUTPUT (4) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) UNRELIABLE_OTHER (7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Polarity | BACnetPolarity | NORMAL (0) | R | R |
| Inactive_Text | CharacterString | OK | O | R |
| Active_Text | CharacterString | Dirty | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Feedback_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

### 7.3.22. FilterReset (Binary Output Object Type)

It resets the filter signal.

| Property Identifier | Property Datatype | Value | ASHRAE | Intesis |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Output, 4) | R | R |
| Object_Name | CharacterString | FilterReset | R | R |
| Object_Type | BACnetObjectType | BINARY_OUTPUT (4) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | W | W |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Polarity | BACnetPolarity | NORMAL (0) | R | R |
| Inactive_Text | CharacterString | Normal | O | R |
| Active_Text | CharacterString | Reset | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Minimum_Off_Time | Unsigned32 | - | O | - |
| Minimum_On_Time | Unsigned32 | - | O | - |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | R |
| Relinquish_Default | BACnetBinaryPV | INACTIVE (0) | R | R |
| Time Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Feedback_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

### 7.3.23. Occupancy (Multistate Value Object Type)

It indicates the current occupancy status.

> **NOTE**
> To know more, see Occupancy Function (page 54).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Multistate Value, 0) | R | R |
| Object_Name | CharacterString | Occupancy | R | R |
| Object_Type | BACnetObjectType | MULTISTATE_VALUE (19) | R | R |
| Present_Value | BACnetBinaryPV | 1 .. 3 | R | R/W |
| Description | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Number_Of_States | Unsigned | 3 | R | R |
| State_Text | BACnetArray[N] of CharacterString | Check the **Occupancy values table** below | O | R |
| Priority_Array | BACnetPriorityArray | - | R | - |
| Relinquish_Default | Unsigned | - | R | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | Unsigned | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

*Table 11. Occupancy values*

| Present_Value | State_Text |
|---|---|
| 1 | Occupied |
| 2 | Unoccupied |
| 3 | Disabled |

## 7.3.24. OccupiedCoolSetPoint (Analog Value Object Type)

It indicates the temperature setpoint when the room is occupied, the cool mode is selected, and the occupancy object is enabled.

> **NOTE**
> To know more, see Occupancy Function (page 54).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Value, 1) | R | R |
| Object_Name | CharacterString | OccupiedCoolSetPoint | R | R |
| Object_Type | BACnetObjectType | ANALOG_VALUE (2) | R | R |
| Present_Value | REAL | 0 .. 65535 | R | R/W |
| Description | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Degrees Celsius (62) Degrees Fahrenheit (64) | R | R |
| Min_Pres_Value | REAL | 0 | O | - |
| Max_Pres_Value | REAL | 65535 | O | - |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTE**
> You can set the temperature scale in Celsius or Fahrenheit scale via DIP switches. More information in DIP switches (page 11).

## 7.3.25. OccupiedHeatSetPoint (Analog Value Object Type)

It indicates the temperature setpoint when the room is occupied, the heat mode is selected, and the occupancy object is enabled.

> **NOTE**
> To know more, see Occupancy Function (page 54).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Value, 2) | R | R |
| Object_Name | CharacterString | OccupiedHeatSetPoint | R | R |
| Object_Type | BACnetObjectType | ANALOG_VALUE (2) | R | R |
| Present_Value | REAL | | R | R/W |
| Description | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Degrees Celsius (62) Degrees Fahrenheit (64) | R | R |
| Min_Pres_Value | REAL | | O | - |
| Max_Pres_Value | REAL | | O | - |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTE**
> You can set the temperature scale in Celsius or Fahrenheit scale via DIP switches. More information in DIP switches (page 11).

## 7.3.26. UnoccupiedCoolSetPoint (Analog Value Object Type)

It indicates the setpoint when the room is unoccupied, the cool mode is selected, and the occupancy object is enabled.

> **NOTE**
> To know more, see Occupancy Function (page 54).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Value, 3) | R | R |
| Object_Name | CharacterString | UnoccupiedCoolSetPoint | R | R |
| Object_Type | BACnetObjectType | ANALOG_VALUE (2) | R | R |
| Present_Value | REAL | | R | R/W |
| Description | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Degrees Celsius (62) Degrees Fahrenheit (64) | R | R |
| Min_Pres_Value | REAL | | O | - |
| Max_Pres_Value | REAL | | O | - |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTE**
> You can set the temperature scale in Celsius or Fahrenheit scale via DIP switches. More information in DIP switches (page 11).

## 7.3.27. UnoccupiedHeatSetPoint (Analog Value Object Type)

It indicates the setpoint temperature when the room is unoccupied, the heat mode is selected, and the occupancy object is enabled.

> **NOTE**
> To know more, see Occupancy Function (page 54).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Value, 4) | R | R |
| Object_Name | CharacterString | UnoccupiedHeatSetPoint | R | R |
| Object_Type | BACnetObjectType | ANALOG_VALUE (2) | R | R |
| Present_Value | REAL | 0 .. 65535 | R | R/W |
| Description | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Degrees Celsius (62) Degrees Fahrenheit (64) | R | R |
| Min_Pres_Value | REAL |  | O | - |
| Max_Pres_Value | REAL |  | O | - |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTE**
> You can set the temperature scale in Celsius or Fahrenheit scale via DIP switches. More information in DIP switches (page 11).

## 7.3.28. OccupancyContinuousCheck (Binary Value Object Type)

It indicates if the system is continuously checking the setpoint and occupancy conditions.

> **NOTE**
> To know more, see Occupancy Function (page 54).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Value, 0) | R | R |
| Object_Name | CharacterString | OccupancyContinuousCheck | R | R |
| Object_Type | BACnetObjectType | BINARY_VALUE (5) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | R | R/W |
| Description | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Inactive_Text | CharacterString | Disabled | O | R |
| Active_Text | CharacterString | Enabled | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Minimum_Off_Time | Unsigned32 | - | O | - |
| Minimum_On_Time | Unsigned32 | - | O | - |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | - |
| Relinquish_Default | BACnetBinaryPV | INACTIVE (0) | R | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.29. UnoccupiedDeadbandAction (Binary Value Object Type)

It indicates the action to be performed when Unoccupancy is enabled, and Room Temperature is within the deadband.

> **NOTE**
> To know more, see Occupancy Function (page 54).

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Value, 1) | R | R |
| Object_Name | CharacterString | UnoccupiedDeadbandAction | R | R |
| Object_Type | BACnetObjectType | BINARY_VALUE (5) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | R | R/W |
| Description | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Inactive_Text | CharacterString | Off | O | R |
| Active_Text | CharacterString | CurrentMode | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Minimum_Off_Time | Unsigned32 | - | O | - |
| Minimum_On_Time | Unsigned32 | - | O | - |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | - |
| Relinquish_Default | BACnetBinaryPV | INACTIVE (0) | R | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

### 7.3.30. LockRemoteControl (Binary Value Object Type)

It is used to lock or unlock the AC remote controller.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Binary Value, 2) | R | R |
| Object_Name | CharacterString | LockRemoteControl | R | R |
| Object_Type | BACnetObjectType | BINARY_VALUE (5) | R | R |
| Present_Value | BACnetBinaryPV | INACTIVE (0) / ACTIVE (1) | R | R/W |
| Description | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Inactive_Text | CharacterString | Unlocked | O | R |
| Active_Text | CharacterString | Locked | O | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Minimum_Off_Time | Unsigned32 | - | O | - |
| Minimum_On_Time | Unsigned32 | - | O | - |
| Priority_Array | BACnetPriorityArray | BACnetPriorityArray | R | - |
| Relinquish_Default | BACnetBinaryPV | INACTIVE (0) | R | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

## 7.3.31. OutdoorTemp_status (Analog Input Object Type)

It indicates the outdoor temperature from the outdoor unit sensor.

| Property Identifier | Property Datatype | Value | ASHRAE | GATEWAY |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Input, 8) | R | R |
| Object_Name | CharacterString | OutdoorTemp_status | R | R |
| Object_Type | BACnetObjectType | ANALOG_INPUT (0) | R | R |
| Present_Value | REAL | -25 .. 43°C / -13 .. 109.4°F | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0), UNRELIABLE_OTHER (7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Update_Interval | Unsigned | - | O | - |
| Units | BACnetEngineeringUnits | Celsius degrees (62) Fahrenheit degrees (64) | R | R |
| Min_Pres_Value | REAL | -25°C / -13°F | O | R |
| Max_Pres_Value | REAL | 43°C / 109.4°F | O | R |
| Resolution | REAL | - | O | - |
| COV_Increment | REAL | 0 | O | W |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| High_Limit | REAL | - | O | - |
| Low_Limit | REAL | - | O | - |
| Deadband | REAL | - | O | - |
| Limit_Enable | BACnetLimitEnable | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

> **NOTE**
> You can set the temperature scale in Celsius or Fahrenheit via the DIP switches. More information in DIP switches (page 11).

### 7.3.32. CenterRemote (Multistate Input Object Type)

This object indicates whether only the BMS (Center I, Center II), the wired remote controller (Remote), or both (Center/Remote) can manage the indoor unit.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Multi-state Input, 0) | R | R |
| Object_Name | CharacterString | CenterRemote | R | R |
| Object_Type | BACnetObjectType | MULTISTATE_INPUT (13) | R | R |
| Present_Value | Unsigned | 1 .. 4 | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE/TRUE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0), UNRELIABLE_OTHER(7) | O | R |
| Out_Of_Service | BOOLEAN | FALSE | R | R |
| Number_Of_States | Unsigned | 4 | R | R |
| State_Text | BACnetArray[N] of CharacterString | Check the **CenterRemote table** below | O | R |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Values | List of Unsigned | - | O | - |
| Fault_Values | List of Unsigned | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |

*Table 12. CenterRemote Table*

| Present_Value | State_Text |
|---|---|
| 1 | Remote Enabled (Center/Remote) |
| 2 | Remote ON/OFF Lock (Center I) |
| 3 | Remote ALL Lock (Center II) |
| 4 | Just Remote (Remote) |

### 7.3.33. DIP_SW_S1_status (Analog Input Object Type)

It indicates the status of the DIP switch block SW1 in decimal value. To get the status of each individual switch of SW1, just convert it to binary. The gateway reads this value only when booting up.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Input, 9) | R | R |
| Object_Name | CharacterString | DIP_SW_S1_status | R | R |
| Object_Type | BACnetObjectType | ANALOG_INPUT (0) | R | R |
| Present_Value | BACnetBinaryPV | 0 .. 255 | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE / TRUE | R | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |
| Units | BACnetEngineeringUnits | No units (95) | R | R |
| COV_Increment | REAL | 0 | O | W |

### 7.3.34. DIP_SW_S2_status (Analog Input Object Type)

It indicates the status of the DIP switch block SW2 in decimal value. To get the status of each individual switch of SW2, just convert it to binary. The gateway reads this value only when booting up.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Input, 10) | R | R |
| Object_Name | CharacterString | DIP_SW_S2_status | R | R |
| Object_Type | BACnetObjectType | ANALOG_INPUT (0) | R | R |
| Present_Value | BACnetBinaryPV | 0 .. 255 | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Out_Of_Service | BOOLEAN | FALSE / TRUE | R | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |
| Units | BACnetEngineeringUnits | No units (95) | R | R |
| COV_Increment | REAL | 0 | O | W |

### 7.3.35. SerialNumber (Analog Input Object Type)

It indicates the serial number of the gateway with the pattern **000EXXXXX**, where:

- **000E** is constant and not included in the Present Value property.

- **XXXXX** is the unique device serial number. This is the information provided by the Present Value.

| Property Identifier | Property Datatype | Value | ASHRAE | Gateway |
|---|---|---|---|---|
| Object_Identifier | BACnetObjectIdentifier | (Analog Input, 11) | R | R |
| Object_Name | CharacterString | SerialNumber | R | R |
| Object_Type | BACnetObjectType | ANALOG_INPUT (0) | R | R |
| Present_Value | BACnetBinaryPV | 00000 .. 99999 | R | R |
| Description | CharacterString | - | O | - |
| Device_Type | CharacterString | - | O | - |
| Status_Flags | BACnetStatusFlags | {FALSE, FALSE, FALSE, FALSE} | R | R |
| Event_State | BACnetEventState | STATE_NORMAL (0) | R | R |
| Reliability | BACnetReliability | NO_FAULT_DETECTED (0) | O | R |
| Polarity | BACnetPolarity | NORMAL (0) | R | R |
| Change_Of_State_Time | BACnetDatetime | - | O | - |
| Change_Of_State_Count | Unsigned | - | O | - |
| Time_Of_State_Count_Reset | BACnetDatetime | - | O | - |
| Elapsed_Active_Time | Unsigned | - | O | - |
| Time_Of_Active_Time_Reset | BACnetDatetime | - | O | - |
| Time_Delay | Unsigned | - | O | - |
| Notification_Class | Unsigned | - | O | - |
| Alarm_Value | BACnetBinaryPV | - | O | - |
| Event_Enable | BACnetEventTransitionBits | - | O | - |
| Acked_Transitions | BACnetEventTransitionBits | - | O | - |
| Notify_Type | BACnetNotifyType | - | O | - |
| Event_Time_Stamps | BACnetArray[N] of BACnetTimeStamp | - | O | - |
| Profile_Name | CharacterString | - | O | - |
| Units | BACnetEngineeringUnits | No units (95) | R | R |
| COV_Increment | REAL | 0 | O | W |

# 8. Occupancy Function

The Occupancy function determines the AC unit's behavior depending on the presence or absence of people in the room. This signal is processed directly in the Intesis gateway and has the capacity to modify three parameters of the AC system: Setpoint, Mode, and On/Off.

> **!** **IMPORTANT**
>
> This function requires a presence sensor on the control system (BMS) side, which determines the state of the **Occupancy** object:
>
> - **Occupied**: Someone is in the room.
>
> - **Unoccupied**: No one is in the room.
>
> - **Disabled**: The function is disabled

Besides the Occupancy object, and to adjust the settings of the Occupancy function, the gateway offers these BACnet objects:

- **Occupancy Cool Setpoint (OCS):** Setpoint temperature when the AC unit is in cool mode and someone is present in the room.

- **Unoccupancy Cool SetPoint (UCS):** Setpoint temperature when the AC unit is in cool mode and no one is in the room.

- **Occupancy Heat Setpoint (OHS):** Setpoint temperature when the AC unit is in heat mode and someone is present in the room.

- **Unoccupancy Heat SetPoint (UHS):** Setpoint temperature when the AC unit is in heat mode and no one is in the room.

*Figure 4. Temperature setpoint objects related to the room's occupancy status and the AC unit's mode*
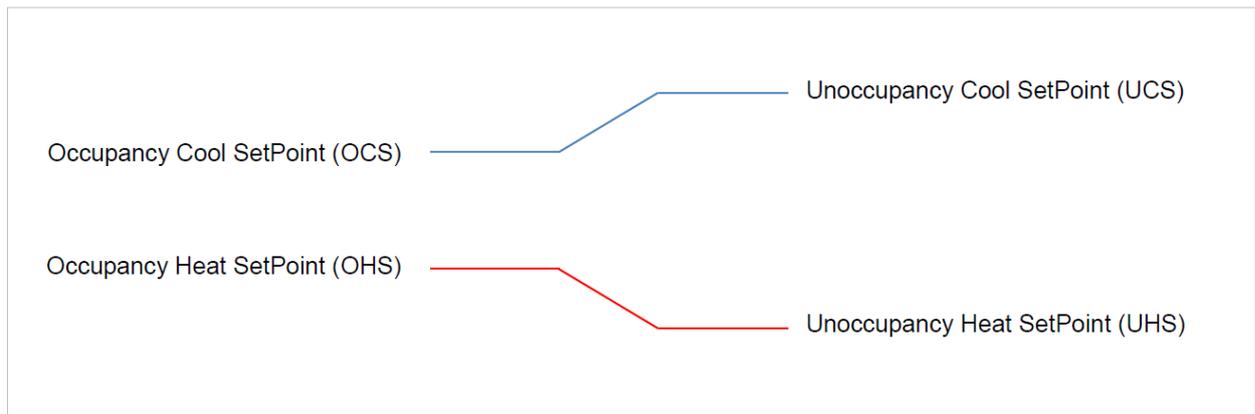


> **i** **NOTICE**
>
> The minimum difference between Cool and Heat setpoints must be 2°C / 4°F.

- **Occupancy Continuous check:** It determines when the gateway checks the room occupancy:

  - If this object's value is 1 (active), the gateway checks the occupancy when the occupancy status and the room temperature change.

- If this object's value is 0 (inactive), the gateway only checks the occupancy when the occupancy status changes.

- **Unoccupied Deadband Action:** It determines the AC unit's behavior when the room is unoccupied and the ambient temperature is within the deadband.

  - If this object's value is 1 (active), the indoor unit will remain on.

  - If this object's value is 0 (inactive), the indoor unit will turn off.

When there is presence in the room, and according to the current room temperature, the AC unit's **mode, setpoint,** and **on/off** will be set to:

| Condition | Setpoint | Mode | On/Off |
|---|---|---|---|
| Room temperature > OCS | Current OCS value | Cool | On |
| Room temperature < OHS | Current OCS value | Heat | On |
| OCS < Room temperature > OHS | OCS/OHS depending on the current mode (If Fan or Dry mode is active, no setpoint is sent) | Current mode | On |

When there is no presence in the room, and according to the current room temperature, the AC unit's **mode, setpoint** and **on/of**f will be set to:

| Condition | Setpoint | Mode | On/Off |
|---|---|---|---|
| Room temperature > OCS | Current UCS value | Cool | On |
| Room temperature < OHS | Current UHS value | Heat | On |
| OCS < Room temperature > OHS | UCS/UHS depending on the current mode (If Fan or Dry mode is active, no setpoint is sent) | Current mode | On (Deadband action=1) |
|  |  |  | Off (Deadband action=0) |

> **NOTICE**
>
> Any local change (for example with the remote control) in the Setpoint, Mode, or the On/Off signal will disable the Occupancy function.

# 9. Ambient Temperature and Virtual Temperature Function

The IN485MHI001R000 gateway enables the use of a temperature sensor from the BACnet-based BMS.

The reason for using this sensor could be that it is better positioned than the sensor in the indoor unit or the wired remote controller. If an indoor unit or wired remote controller is mounted far away, it can lead to a significant difference between the temperature perceived by their sensors and the actual temperature in the occupied space.

*Figure 5. In this case, the room temperature is four degrees Celsius less than the temperature perceived by the indoor unit's sensor.*
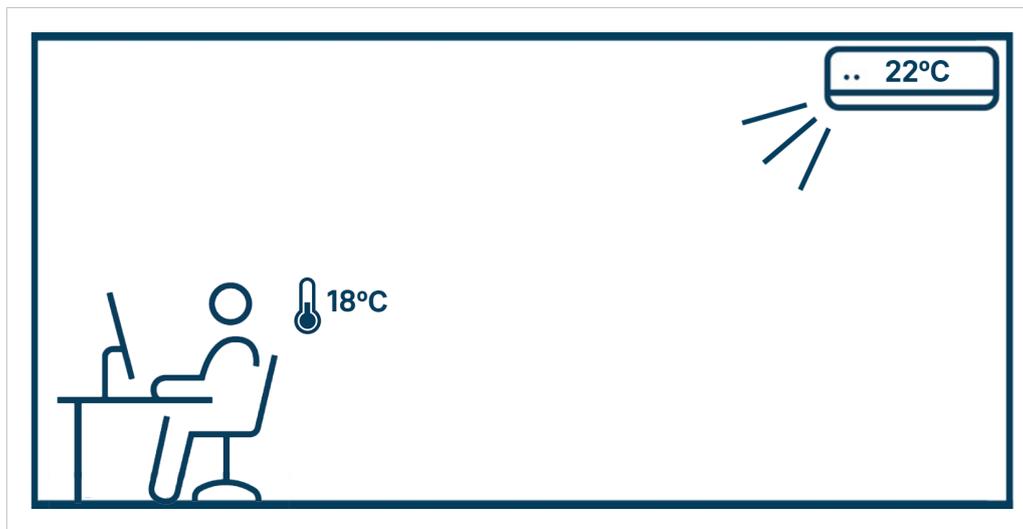


When using the temperature sensor from the BMS side, the IN485MHI001R000 gateway allows two different options:

• Direct overwriting of the AC system reference temperature.

• Activation of the Virtual Temperature function.

As explained below, the choice between these options depends on the role of both the gateway and the Mitsubishi Heavy Industries wired RC. It also depends on the thermostat used by the AC system to determine its reference temperature, which can be either the sensor inside the indoor unit or the sensor inside the wired RC.

**DIRECT OVERWRITING OF THE INDOOR UNIT'S REFERENCE TEMPERATURE**

1.  Set the IN485MHI001R000 gateway as the header of the bus by setting the SW1-1 (DIP switch 1, position 1) ON. See DIP switches (page 11).

2.  Set the IN485MHI001R000 gateway to read the temperature provided by the wired RC by setting the SW1-2 (DIP switch 1, position 2) ON. See DIP switches (page 11).

3.  Set the Mitsubishi Heavy Industries wired RC as the follower of the bus.

> **NOTE**
> Refer to the documentation provided with the Mitsubishi Heavy Industries system to know the needed procedure to set the role of the wired RC.

4.  Set the Mitsubishi Heavy Industries AC system to read the temperature provided by the sensor inside the wired RC.

> **NOTE**
> This configuration must be performed by a Mitsubishi Heavy Industries authorized installer through the remote controller.

5.  At this point, the indoor unit could report an error warning that there is an abnormality in the wired RC's thermistor.

> **NOTICE**
> This error is reported because no valid value has been written yet as the ambient temperature. Take into consideration that, after commissioning the gateway for the first time, the default value for the BACnet object **RoomTemperature_command (Analog Output)** is -32768, which is an invalid value.

6.  Write the value reported by the BMS sensor using the BACnet object **RoomTemperature_command (Analog Output)**.

> **NOTE**
> At this point, any error reported by the indoor unit will disappear. From now on, the AC system will use the value reported by the BACnet temperature sensor as its reference temperature.

**ACTIVATION OF THE VIRTUAL TEMPERATURE FUNCTION**

> **NOTE**
> Before explaining how to activate the Virtual Temperature function, it is important to understand how this function works.

**How the Virtual Temperature function works:**

The Virtual Temperature function uses the value reported by the temperature sensor connected to the BMS side to apply a formula that establishes the setpoint temperature for the indoor unit. This recalculated setpoint offsets the difference between the real temperature in the room and the temperature at which the indoor unit is operating.

The formula used by the Virtual Temperature function is the following:

$S_{AC} = T_{AC} - (T_{BMS} - S_{BMS})$

Where:

- $S_{AC}$: Recalculated temperature setpoint sent to the indoor unit after the gateway applies the formula.

- $T_{AC}$: Indoor unit's reference temperature.

- $T_{BMS}$: Ambient temperature reported by the sensor connected to the BMS side.

- $S_{BMS}$: Temperature setpoint requested from the BMS side.

Once activated, the Virtual Temperature function recalculates the setpoint when any of these values changes.

In practical terms, when you write a value in the BACnet object **RoomTemperature_command (Analog Output)**, the gateway activates the Virtual Temperature function, using this value to recalculate the indoor unit's reference temperature.

**How to activate the Virtual Temperature function:**

The Virtual Temperature function can be activated with the gateway acting either as the header or the follower of the wired RC bus.

- **Procedure with the gateway acting as the header of the RC bus:**

    1. Set the IN485MHI001R000 gateway as the header of the bus by setting the SW1-1 (DIP switch 1, position 1) ON. See DIP switches (page 11).

    2. Set the IN485MHI001R000 gateway to read the temperature provided by the indoor unit by setting the SW1-2 (DIP switch 1, position 2) OFF. See DIP switches (page 11).

    3. Set the Mitsubishi Heavy Industries wired RC as the follower of the bus.

        > **NOTE**
        > Refer to the documentation provided with the Mitsubishi Heavy Industries system to know the needed procedure to set the role of the wired RC.

    4. Set the Mitsubishi Heavy Industries AC system to read the temperature provided by the sensor inside the indoor unit.

        > **NOTE**
        > This configuration must be performed by a Mitsubishi Heavy Industries authorized installer through the remote controller.

- **Procedure with the gateway acting as the follower of the RC bus:**

    1. Set the IN485MHI001R000 gateway as the follower of the bus by setting the SW1-1 (DIP switch 1, position 1) OFF. See DIP switches (page 11).

    2. Set the Mitsubishi Heavy Industries wired RC as the header of the bus.

        > **NOTE**
        > Refer to the documentation provided with the Mitsubishi Heavy Industries system to know the needed procedure to set the role of the wired RC.

    3. Set both the gateway and the AC system to read the temperature from the same source, either from the thermistor inside the indoor unit or the wired RC.

        a. For the gateway, this is done via the SW1-2 (DIP switch 1, position 2):

            – ON: The gateway reads the temperature from the thermistor of the wired RC.

            – OFF: The gateway reads the temperature from the thermistor of the indoor unit.
            To know more, see DIP switches (page 11).

        b. For the AC system, this configuration is done via the wired RC must and must be performed by a Mitsubishi Heavy Industries authorized installer.

Once the roles and reading sources of both the gateway and the wired RC are set, proceed as follows:

1. Write the desired setpoint temperature in the BACnet object **Setpoint_command (Analog Output)**.

2. Read the temperature value reported by a temperature sensor from the BMS side.

3. Write this value in the BACnet object **RoomTemperature_command (Analog Output)**.

> **NOTE**
> At this point, the Virtual Temperature function is automatically activated. From now on, the AC system will use the value reported by the formula described above as its reference temperature.

> **NOTICE**
> Due to the formula applied by the Virtual Temperature function, the temperature reported by the AC system and the actual ambient temperature may differ. This must not be taken as an error, but as the normal behavior when the Virtual Temperature function is working.
>
> To know more about the BACnet objects dedicated to the temperature signals and to the Virtual Temperature function, see Considerations on Temperature Signals (page 60).

> **NOTE**
> When starting up the gateway, the **RoomTemperature_command (Analog Output)**, which is the dedicated object to write the ambient temperature perceived from the BMS side, reports a value of -32768 (0x8000).

> **NOTICE**
> When starting up the gateway, the Present_Value property for the RoomTemperature_command object is 0, and the Reliability property displays **UNRELIABLE_OTHER (7)**. This means that no external temperature reference has been provided to the object, so the system is not applying the Virtual Temperature function. However, after receiving the first value, the Reliability property changes to **NO_FAULT_DETECTED (0)**. After that, any value can be used in the temperature range, including 0.

**TROUBLESHOOTING**

Typical malfunctions with the ambient temperature and the Virtual Temperature function are related to mistakes in the configuration of the gateway's or the wired RC's role, or in the values written in the BACnet objects. If an error occurs, consider the following:

- **The gateway and the wired RC have the correct role.**

  Remember that the gateway can overwrite the reference temperature of the AC system using a temperature sensor from the BMS side only when it is set as the header and the wired RC is set as the follower.

  To set the gateway as the header of the bus, set the SW1-1 (DIP switch 1, position 1) ON. To know more, see Coexistence of the Gateway with a Remote Controller (page 10).

  The role of the wired RC is usually set through a DIP switch mounted on its board, but you should read the documentation delivered with the AC system to ensure the necessary procedure.

- **The gateway and the wired RC are reading the temperature from the same source.**

  Both the gateway and the AC system can be set to read the temperature from the indoor unit or the wired RC. Only when both the gateway and the AC system are set to read the temperature from the wired RC can the gateway overwrite the reference temperature of the AC system using a temperature sensor from the BMS side.

  Set the gateway to read the temperature of the wired RC thermistor by setting the SW1-2 (DIP switch 1, position 2) ON. To know more, see DIP switches (page 11).

- **Valid values are written in the BACnet objects dedicated to the temperature.**[2]

  Pay special attention to the BACnet object **RoomTemperature_command (Analog Output)**, which is dedicated to write the value reported by the BMS sensor.

> **NOTE**
> [2] The following section provides an extended explanation of all the BACnet objects dedicated to the temperature signals and to the Virtual Temperature function.

## 9.1. Considerations on Temperature Signals

> **NOTE**
> You can set the temperature signals in degrees Celsius or Fahrenheit via the DIP switch SW2. More information in DIP switches (page 11).

*Table 13. Objects dedicated to temperature signals and to the Virtual Temperature function*

| BACnet object | Function when the Virtual Temperature function is inactive | Function when the Virtual Temperature function is active |
|---|---|---|
| Setpoint_status (Analog Input) | It indicates the temperature setpoint sent to the indoor unit.<br><br>It will report the same value as the value introduced in the Setpoint_command object. | It indicates the recalculated temperature setpoint sent to the indoor unit after the gateway applies the $S_{AC} = T_{AC} - (T_{BMS} - S_{BMS})$ formula.<br><br>It may report a value different from that introduced in the Setpoint_command object. |
| Setpoint_command (Analog Output) | It is used to request the temperature setpoint from the BMS side.<br><br>It will report the same value as a wired remote controller connected to the indoor unit (if available). | It is used to request the temperature setpoint from the BMS side.<br><br>It may report a value different from that reported by a wired remote controller connected to the indoor unit (if available). |
| RoomTemperature_status (Analog Input) | It indicates the ambient temperature perceived by the sensor from the AC system side (the sensor inside the indoor unit or inside the wired remote controller, if available). | It indicates the ambient temperature perceived by the sensor from the AC system side (the sensor inside the indoor unit or inside the wired remote controller, if available). |
| RoomTemperature_command (Analog Output) | It is used to activate the Virtual Temperature function by writing the value reported from a BMS side sensor. | It indicates the temperature reported from a BMS side sensor. |
| VirtualTemperatureActive (Binary Input) | It reports a value of 0 | It reports a value of 1 |
| UserSetpoint_status (Analog Input) | It indicates the temperature setpoint requested from the BMS side. | It indicates the original temperature setpoint requested from the BMS side. |

**VIRTUAL TEMPERATURE FUNCTION EXAMPLE CASE**

Imagine a very cold room with a temperature of 10°C. There's an AC indoor unit mounted in the ceiling, which is very high.

The technician responsible for the BMS wants to raise the room temperature to 20°C. To achieve this, she accesses the console and sets this value in the BACnet object **Setpoint_command (Analog Output)**. The same value of "20" is also reflected in the BACnet object **Setpoint_status (Analog Input)**.

A few minutes later, the technician checks the BACnet object **RoomTemperature_status (Analog Input)** to determine the temperature reported by the indoor unit's sensor, which reads 17°C. However, the BMS sensor in the room reports a temperature of 13°C. The technician knows that the BMS sensor is better positioned than the indoor unit's sensor. Unfortunately, the indoor unit does not allow the gateway a direct overwriting of the value reported by its sensor temperature. To address this, the technician activates the Virtual Temperature function.

To activate it, she writes the temperature currently perceived by the BMS sensor (13°C) into the BACnet object **RoomTemperature_command (Analog Output)**. As soon as she inputs "13" into that object/register, the BACnet object **VirtualTemperatureActive (Binary Input)** transitions from "0" to "1." This indicates that the Virtual Temperature function is now active and will continuously apply the formula to recalculate the setpoint temperature sent to the indoor unit.

At this moment, the formula values are: 17-(13-20)=24. Therefore, the Virtual Temperature function is currently sending a setpoint of 24°C to the indoor unit, and this value is reflected in the BACnet object **Setpoint_status (Analog Input)**.

After a few minutes, the technician checks the BACnet object **RoomTemperature_command (Analog Output)** to find the temperature perceived by the BMS sensor: 19°C. Then, she looks at the BACnet object **RoomTemperature_status (Analog Input)**, which reports the temperature sensed by the indoor unit: 24°C.

At this point, the formula applied by the Virtual Temperature function is based on these numbers: 24-(19-20)=25. The technician observes the BACnet object **Setpoint_status (Analog Input)** and realizes that the Virtual Temperature function has established the setpoint at 25°C.

# 10. Error Codes

> **NOTE**
> The error codes displayed in the remote controller may vary depending on the AC unit model. Read the documentation of your unit to know its specific error codes.

| Error code | Code in the wired RC | Description |
|---|---|---|
| 0 | N/A | No active error |
| 1 | E1 | Transmission error between the indoor unit and the remote controller |
| 2 | E2 | Indoor unit address duplication |
| 3 | E3 | Indoor unit, outdoor unit addressing error |
| 4 | E4 | Invalid indoor unit address |
| 5 | E5 | Communication error between indoor unit and outdoor unit |
| 6 | E6 | Indoor unit heat exchanger sensor disconnection |
| 7 | E7 | Indoor unit return air sensor disconnection |
| 8 | E8 | Abnormal indoor unit heat exchanger temperature |
| 9 | E9 | Blocked drain error, malfunction of optional parts |
| 10 | E10 | Excessive number of indoor units connected to one remote controller |
| 11 | E11 | Remote control addressing setting (RAS) is done when plural remote controllers are connected |
| 12 | E12 | Invalid indoor unit address setting |
| 13 | E13 | SAF malfunction |
| 14 | E14 | Communication error between master and slave indoor units |
| 15 | E15 | Indoor unit supply air sensor disconnection |
| 16 | E16 | Indoor unit fan motor malfunction |
| 17 | E17 | Gas sensor malfunction |
| 19 | E19 | Setting of operation check |
| 20 | E20 | Indoor unit fan motor rotation |
| 28 | E28 | Return air sensor on remote controller disconnection |
| 30 | E30 | Cross connection between piping and wiring |
| 31 | E31 | Outdoor unit address duplication |
| 32 | E32 | Reverse phase (missing phase on primary side) |
| 33 | E33 | Over current on the compressor |
| 34 | E34 | Missing phase |
| 35 | E35 | Outdoor unit abnormal heat exchanger temperature |
| 36 | E36 | Abnormal discharge pipe temperature |
| 37 | E37 | Outdoor heat exchanger sensor disconnection |
| 38 | E38 | Outdoor unit return air sensor disconnection |
| 39 | E39 | Discharge pipe temperature sensor disconnection |
| 40 | E40 | Abnormal high pressure |
| 41 | E41 | Overheat of the power transistor |
| 42 | E42 | Current cut |
| 43 | E43 | Excessive number of connected indoor units |
| 44 | E44 | Abnormal compressor bottom temperature |
| 45 | E45 | Communication error between inverter and outdoor unit main PCB |
| 46 | E46 | Mixed automatic address setting (AAS) and remote control addressing setting (RAS) |
| 47 | E47 | Abnormal compressor oil pressure/voltage |

| Error code | Code in the wired RC | Description |
|---|---|---|
| 48 | E48 | Outdoor unit fan motor error |
| 49 | E49 | Low pressure error/Service valve closed |
| 50 | E50 | Ice storage unit error |
| 51 | E51 | Abnormal freezing, insufficient water supply/Inverter error |
| 52 | E52 | Water temperature sensor disconnection |
| 53 | E53 | Suction pipe temperature sensor disconnection |
| 54 | E54 | High/Low pressure sensor anomaly |
| 55 | E55 | Underneath temperature sensor disconnection |
| 56 | E56 | Power transistor temperature sensor disconnection |
| 57 | E57 | Insufficient gas charge |
| 58 | E58 | Current safe |
| 59 | E59 | Outdoor unit failure/Compressor startup failure |
| 60 | E60 | Compressor rotor lock |
| 61 | E61 | Communication error among outdoor unit |
| 62 | E62 | Outdoor unit address error |
| 63 | E63 | Emergency stop |
| 64 | E64 | Coolant failure |
| 65 | E65 | System malfunction when PAC I/F is used |
| 66 | E66 | I/F address duplication |
| 67 | E67 | V-multi excess number of indoor units |
| 68 | E68 | Communication error between V-multi and the indoor unit |
| 70 | E70 | Communication error between CC1 - indoor unit |
| 71 | E71 | Communication error between NR and the indoor unit |
| 72 | E72 | Communication error between CC2 and the indoor unit |
| 73 | E73 | Communication error between CC3 and the indoor unit |
| 74 | E74 | Address duplication for network equipment |
| 75 | E75 | Communication circuit failure in network optional equipment |
| 80 | E80 | Abnormal engine water temp |
| 81 | E81 | Abnormal oil pressure of engine |
| 82 | E82 | Too much revolution of engine |
| 83 | E83 | Too less revolution of engine |
| 84 | E84 | Engine startup failure |
| 85 | E85 | Engine stop failure |
| 86 | E86 | Oil pressure sensor disconnection |
| 87 | E87 | Water temperature sensor disconnection |
| 88 | E88 | Engine oil failure |
| 89 | E89 | Other engine failure |
| 90 | E90 | Actuator failure |
| 65535 | - | Error communication between the gateway and the AC unit |

**NOTE**
If you detect a non-listed error code, please contact Mitsubishi Heavy Industries technical support.