Shizuku 2

(VRF System Emulator)

Reference Manual

2023/11/08

目次

第1章 はじめに

1.1 Shizuku2 とは

Shizuku2 は VRF（Variable Refrigerant Flow）が導入された建物の熱環境システムを模擬するためのエミュレータソフトウェア（以下、単に「エミュレータ」と呼ぶ）です。

空調システムとして VRF を導入する建物は増え続けており、その性能を正しく予測することには大きな価値があります。しかし、一般に VRF は中央熱源システムに比べて空調システムと執務者との相互作用が大きく、性能を正しく予想することは難しいとされています。これは、執務者がリモートコントローラを直接に操作できて室内環境に影響を与えることができるということや、冷媒と空気が直接に熱交換するために空気の温度が極端な値になりやすいということが原因です。

このため、本エミュレータは、VRF に対する様々な制御が熱環境システムのエネルギー性能と熱的快適性にどのように影響するのかを予想することを目的に開発されました。これらの 2 つの性能の間にあるトレードオフの関係を正しく評価するために、建物、VRF、執務者がそれぞれ精緻にモデル化され、現実が模擬されます。エミュレータのユーザーは BACnet という現実の建物でも使われる汎用の通信方式を使って、あたかも現実に VRF が存在するかのように、VRF の制御を試みることができます。

本資料は、エミュレータの使い方の解説書です。本章の以降の節では、模擬する建物、VRF、執務者について説明します。第 2 章では、Shizuku2 のインストール方法、ディレクトリ構成、簡単な実行例について解説します。第 3 章では、Excel を使ってエミュレータの中にある VRF を制御する方法について解説します。第 4 章では、自作のプログラムによって VRF を制御する方法について解説します。第 5 章では運用を最適化する上で注意すべき点を列挙します。

1.2 模擬対象の熱環境システム

1) 建物

模擬対象の建物の平面図を Fig. 1.1 に示します。北西と南西に面した 2 つの事務室が模擬対象です。それぞれ別のテナントが入居しています。床面積はいずれも 273 m^2 です。細かな間仕切りはありません。

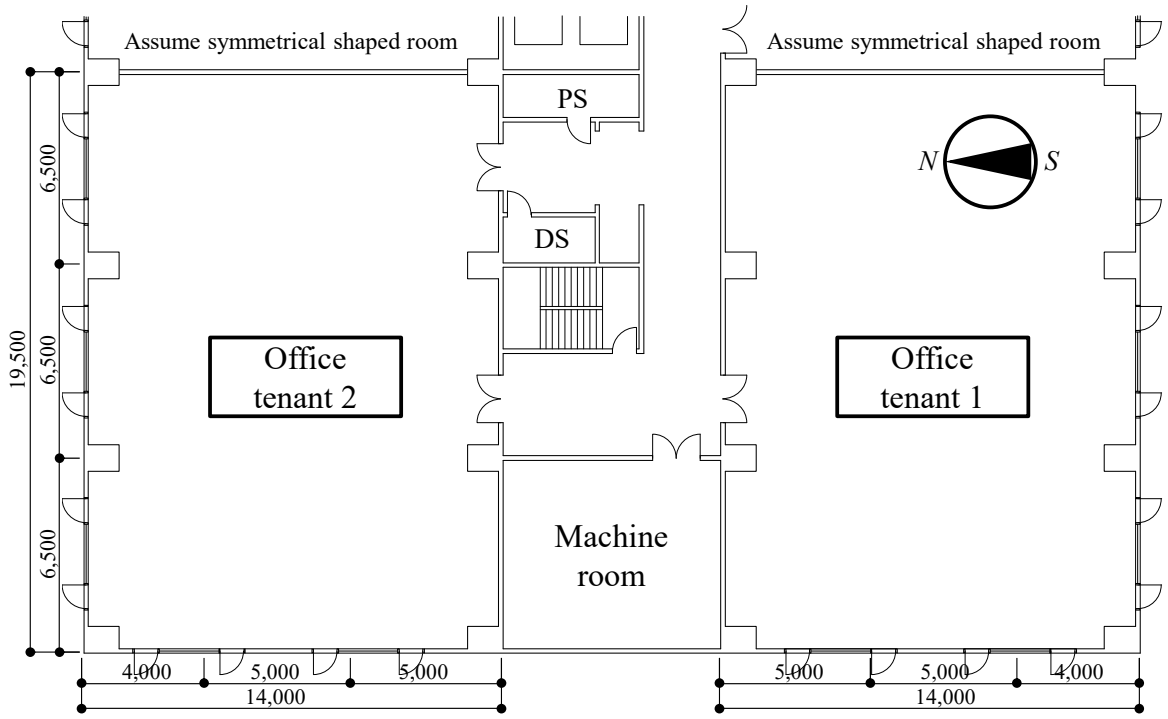


Fig. 1.1 平面図

外壁の断面図を Fig. 1.2 に示します。窓の合計面積は南北が 15.96 m^2、西側が 10.64 m^2 です。

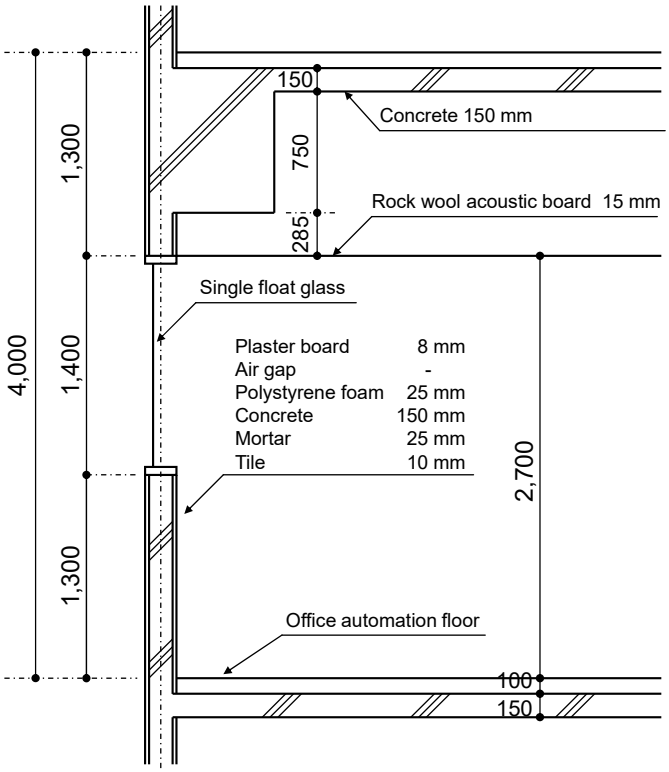


Fig. 1.2 断面図

この建築は日本の東京に建つと仮定します。東京の夏と冬の典型的な気象データを Fig. 1.3 に示します。

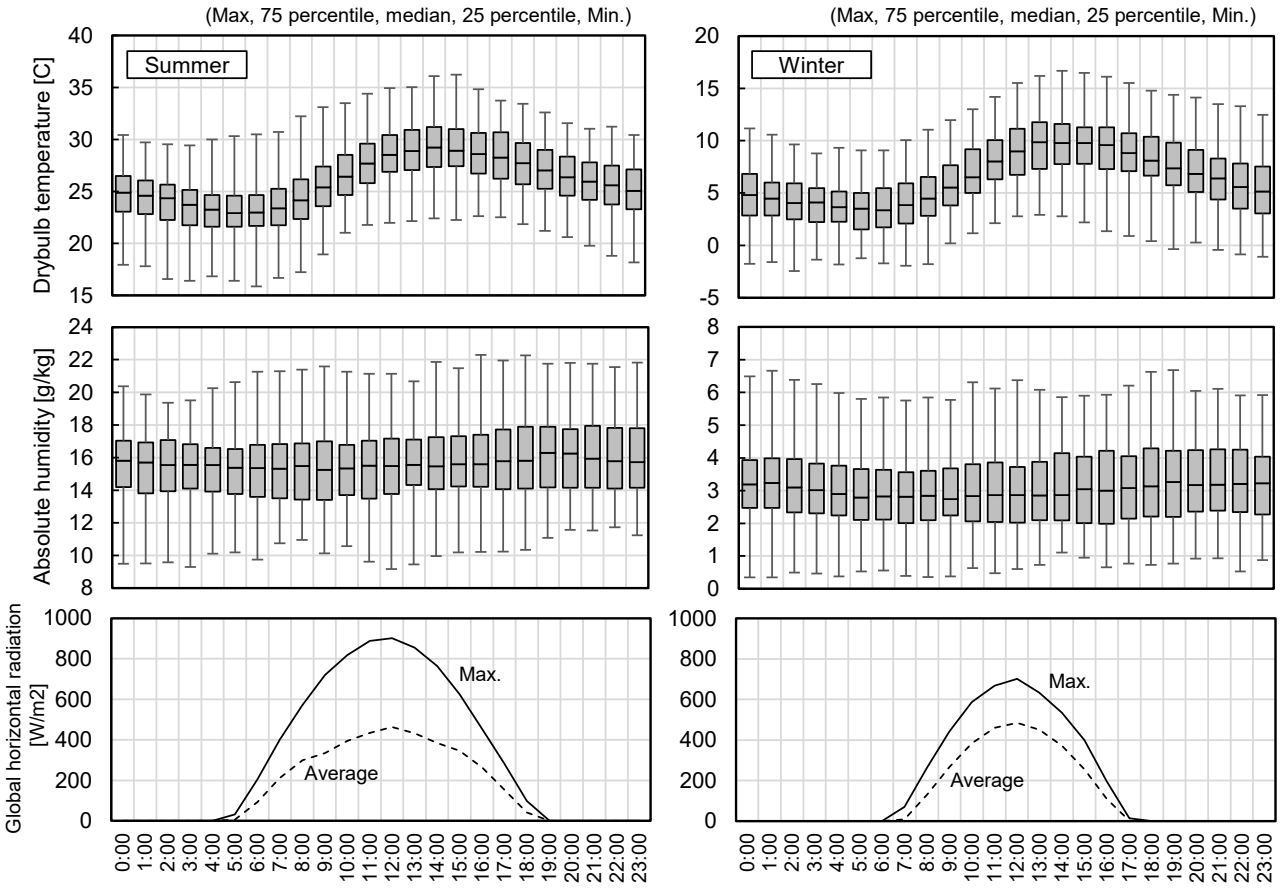


Fig. 1.3 東京の夏季と冬季の気象特性

エミュレータは夏として 7 月 20 日からの 1 週間、または冬として 2 月 10 日からの 1 週間を計算します。Fig. 1.3 はそれぞれの計算の 1 日目にあたる 7 月 20 日と 2 月 10 日の気象データをランダムに 100 回生成し、その統計量を求めた結果です。

2) VRF システム

VRF は北側と南側事務室のそれぞれでインテリア空調用とペリメータ空調用に分け、全部で 4 系統です。Fig. 1.4 にそれぞれの室内機ユニットが空調するゾーンを示します。これらのゾーンのそれぞれに、換気用に小型の全熱交換器が設置されています。

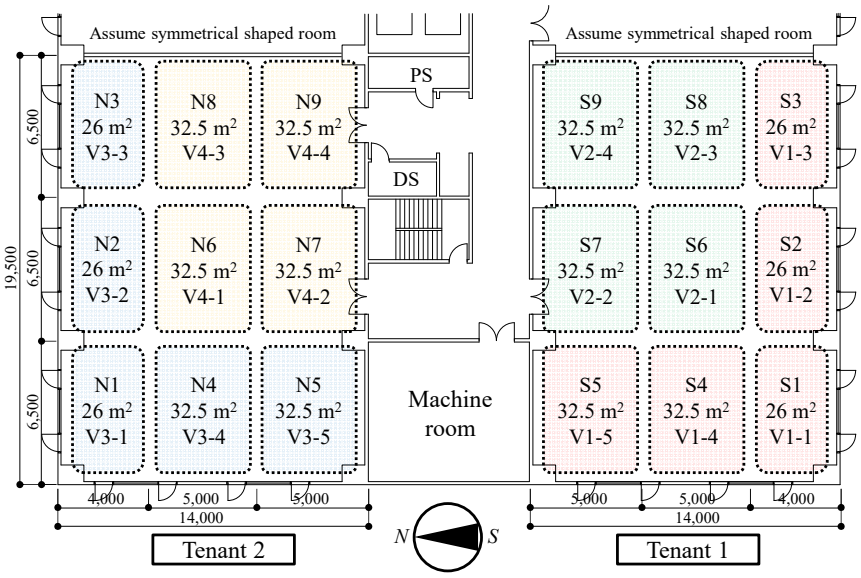


Fig. 1.4 室内機ユニットの空調ゾーン

室外機の仕様を Table 1.1 に示します。いずれも冷暖切替型の機種です。ぞれぞれのゾーンの室内機の仕様を Table 1.2 と Table 1.3 に示します。

Table 1.1 Outdoor unit specifications

-	VRF1	VRF2	VRF3	VRF4
Cooling capacity [kW]	40.0	22.4	33.5	22.4
Cooling electricity [kW]	12.5	6.07	9.74	6.07
Heating capacity [kW]	45.0	25.0	37.5	25.0
Heating electricity [kW]	13.1	6.32	10.0	6.32
Air flow rate [m^3/min]	210	218	187	218
Electricity [kW]	0.58	0.52	0.42	0.52

Table 1.2 Indoor unit specifications

Indoor unit type	C56	C71
Nominal cooling capacity [kW]	5.6	7.1
Nominal heating capacity [kW]	6.3	8.0
Air flow rate [m^3/min]	15.5	22.0
Electricity [kW]	0.043	0.072

Table 1.3 Type of indoor units in each zone

Zone name	N1	N2	N3	N4	N5	N6	N7	N8	N9
I/U name	V3-1	V3-2	V3-3	V3-4	V3-5	V4-1	V4-2	V4-3	V4-4
I/U type	C71	C56	C56	C71	C71	C56	C56	C56	C56
Zone name	S1	S2	S3	S4	S5	S6	S7	S8	S9
I/U name	V1-1	V1-2	V1-3	V1-4	V1-5	V2-1	V2-2	V2-3	V2-4
I/U type	C71	C71	C71	C71	C71	C56	C56	C56	C56

3) 執務者

設定された乱数シードによって人数は変わりますが、事務室には概ね 80 人程度の執務者がいます。執務者は一人ひとりが別々にモデル化されており、異なった行動パターンと温冷感を持っています。執務者一覧は Appendix 2 に整理されています。

Fig. 1.5 はある 1 週間の事務室にいる執務者の数の推移です。それぞれの執務者がどのように入退室するかは確率的に決まるため、人員密度は毎日変わります。また、残業をして夜まで残る執務者や、土日に出勤して働く執務者もいます。

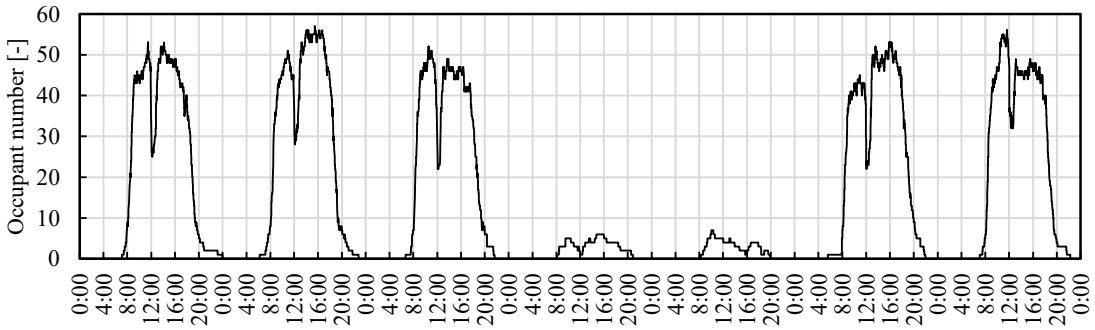


Fig. 1.5 Change in the number of weekly office workers

執務者は室内環境条件によって確率的に不満を表します。その条件は以下の 4 つです。

1）熱環境が暑かったり、寒かったりする
2）冷たい気流が体に直接にあたる
3）上下方向の温度分布が大きい
4）換気不足で空気が汚れる

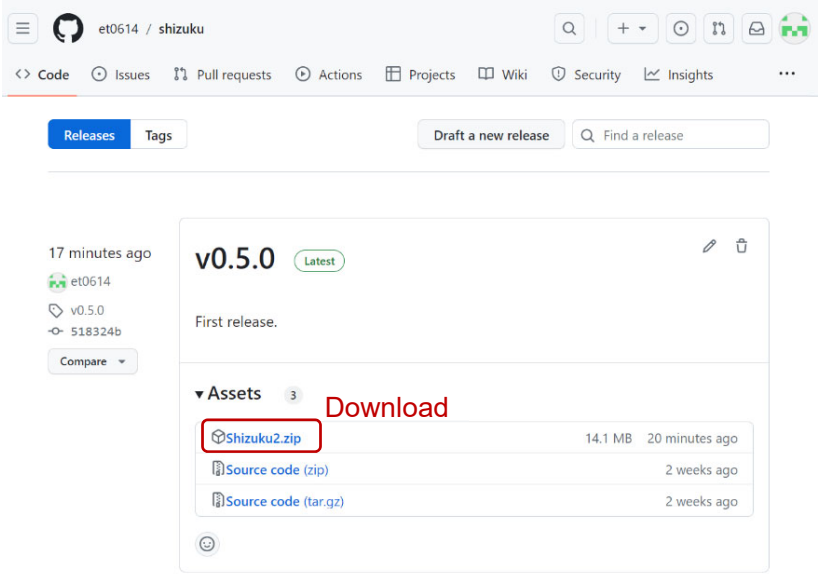
これらの環境条件は、VRF をどのように運用するかによって変化します。

第2章　エミュレータの導入

2.1　インストール

以下の Web サイトから最新のソフトウェアの圧縮ファイル（Shizuku2.zip）をダウンロードします。

https://github.com/et0614/shizuku/releases



エミュレータを実行するためには.NET 6.0 以上が必要です。以下の Web サイトからダウンロードしてインストールします。

https://dotnet.microsoft.com/download

2.2 ディレクトリ

ダウンロードした圧縮ファイルを解凍すると、Fig. 2.1 に示すディレクトリとなります。

```
Shizuku2
    ├── Shizuku2.exe                (1)
    ├── setting.ini                 (2)
    ├── data (Directory)            (3)
    ├── ExcelController.exe         (4)
    ├── schedule.xlsx               (4a)
    ├── schedule_samples.xlsx       (4b)
    ├── CaseStudyProcessor.exe      (4c)
    ├── schedules (Directory)       (4d)
    ├── DummyDeivceController.exe   (5)
    ├── Libraries                   (6)
    └── Other files
```

Fig. 2.1 Shizuku2 directory

「Shizuku2.exe (1)」がエミュレータの実行ファイルです。

「setting.ini (2)」はエミュレータの動きを変えるための初期設定ファイルです。

「data (3)」はエミュレータの計算結果が書き出されるディレクトリです。

エミュレータ内の VRF は外部から BACnet 通信を使って制御します。「ExcelController.exe (4)」を使う方法は最も簡単で、このソフトウェアは予め Excel のファイルに記入されたスケジュールを読み取り、エミュレータと同期を取りながら VRF を制御します。4a~d は関連するファイルとディレクトリで詳細は第 3 章で解説します。

「DummyDeviceController.exe (5)」はエミュレータの中に用意したダミーの BACnet Device と通信するためのサンプルプログラムで、次節で解説します。

「Libraries (6)」は python 言語または C#言語などでエミュレータと通信する際に使うライブラリが収められたディレクトリです。

2.3 エミュレータの起動と BACnet 通信のテスト

Shizuku2.exe をダブルクリックして実行すると、Fig. 2.2 に示すような起動画面が表れます。

エミュレータの中には VRF などの設備のモデルがありますが、エミュレータが起動した時には停止しており、外部から BACnet 通信で操作しなければ動きません。このような BACnet で通信する機器を BACnet controller と呼びます。

外部の BACnet controller がエミュレータに接続するための時間を用意するため、エミュレータは起動して準備計算が終わると一旦、待機状態になります。Fig. 2.2 は、この状態です。ここで、キーボードから「Enter」キーを入力すると待機状態から抜け出し、計算が始まります。

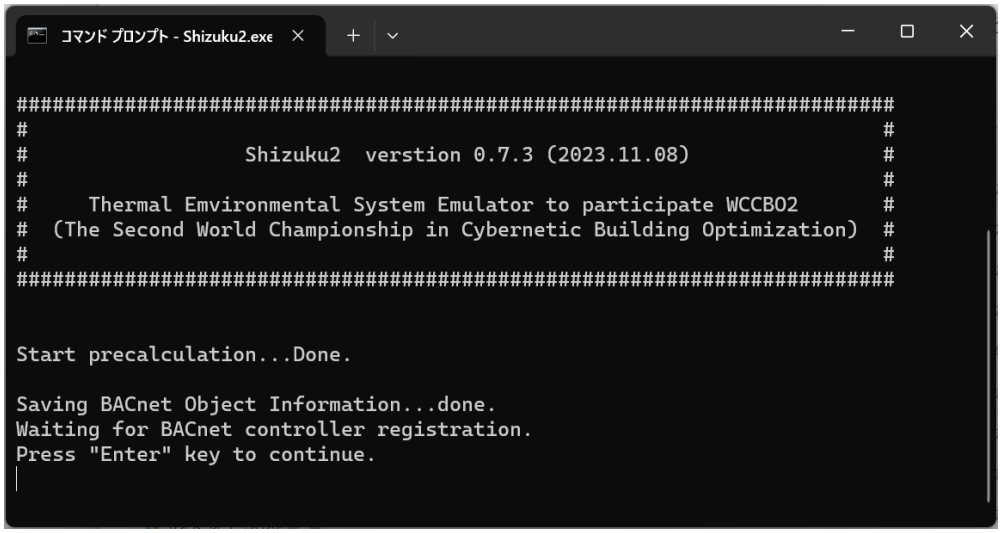


Fig. 2.2 Emulator startup screen

Fig. 2.2 の待機状態であっても BACnet 通信には対応できます。エミュレータの中には、正常に BACnet で通信できるかどうかを試験するための、ダミーの BACnet Device が用意されています。このダミーのデバイスと通信するため、「DummyDeivceController.exe」をダブルクリックで起動します。

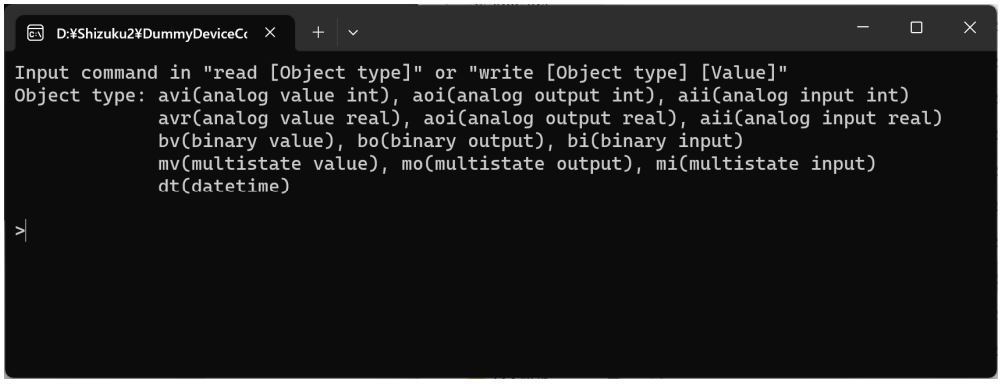


Fig. 2.3 Dummy device controller

エミュレータの内部の様々な状態値は、BACnet オブジェクトとして管理されています。エミュレータが持っている BACnet オブジェクトの一覧は Appendix 1 に記載されています。

代表的な BACnet オブジェクトのタイプと使用例を Table 2.1 に示します。Analog value, input, output は整数や実数値を管理するためのオブジェクトです。Binary value, input, output は真偽値を管理するためのオブジェクトです。Multistate value, input, output は離散的な整数値を管理するためのオブジェクトです。BACnet date time は日付と時刻を管理するためのオブジェクトです。

Value または Output は外部から値を書き込むことができ、主に機械を制御するときに使います。Input は読み取り専用で、主にシステムの状態を監視するときに使います。

Table 2.1 Value and use example of object types

Object types	Value	Use example
Analog value, output	integer or real	Setting setpoint temperature of indoor unit
Analog input	integer or real	Monitor room temperature
Binary value, output	boolean	Setting on/off status of VRF
Binary input	boolean	Monitor on/off status of VRF
Multistate value, output	unsigned integer	Setting fan speed of indoor unit
Multistate input	unsigned integer	Monitor air flow direction of indoor unit
BACnet date time	date and time	Get current date and time in the emulator

ダミーのデバイスの中には、これらのタイプの BACnet オブジェクトが 1 つずつ、用意されています。例えば、整数型の Analog value の値を読み取ってみましょう。コンソールで「read avi」と入力して Enter を押すと Fig. 2.4 となり、現在の状態値として「1」を読み取ることができます。

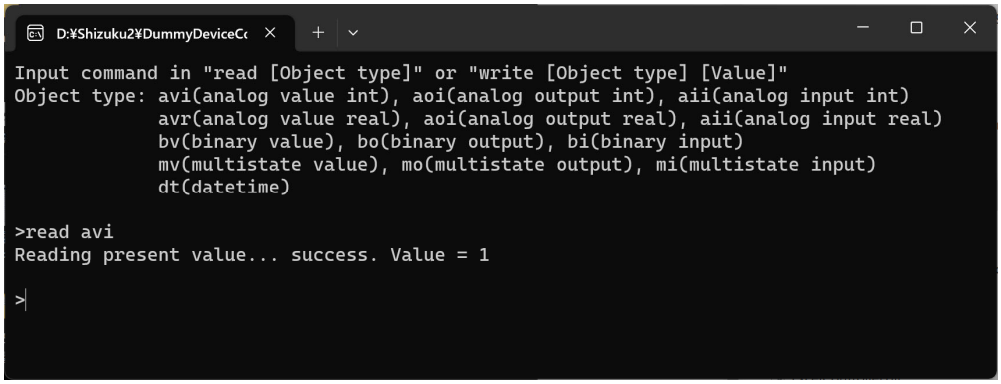


Fig. 2.4 Reading analog value (integer) from the emulator

このとき、エミュレータの画面では Fig. 2.5 に示すように、外部から状態値を読み出す要求が来たことが表示されます。なお、この状態表示は BACnet 通信の試験のため、ダミーデバイスのみで有効にしています。

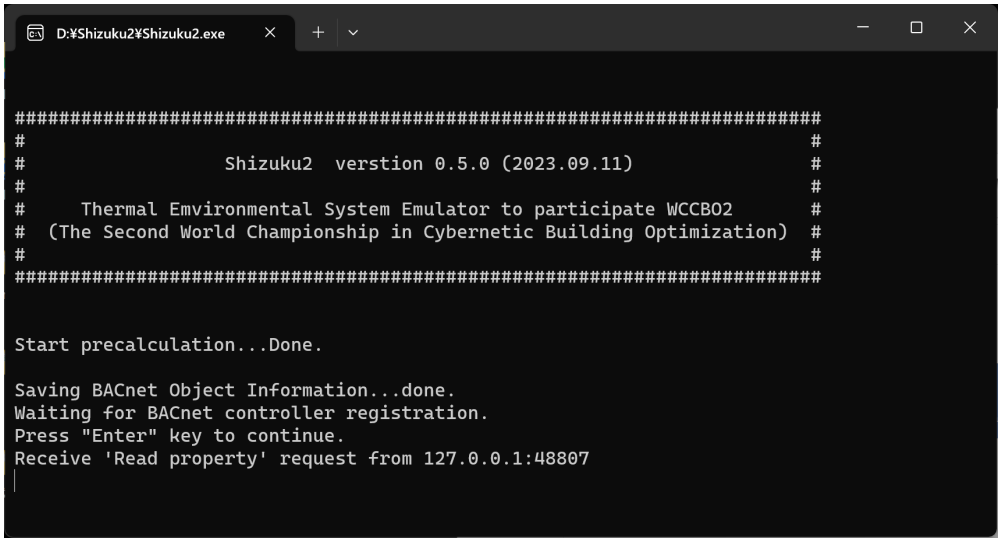


Fig. 2.5 Response of the emulator

Analog value は値を書き換えることもできます。「write avi 5」と入力して Enter を押すと、値が 5 で上書きされます。再び「read avi」と入力すると、Fig. 2.6 が得られ、値が 5 で上書きされたことが確認できます。

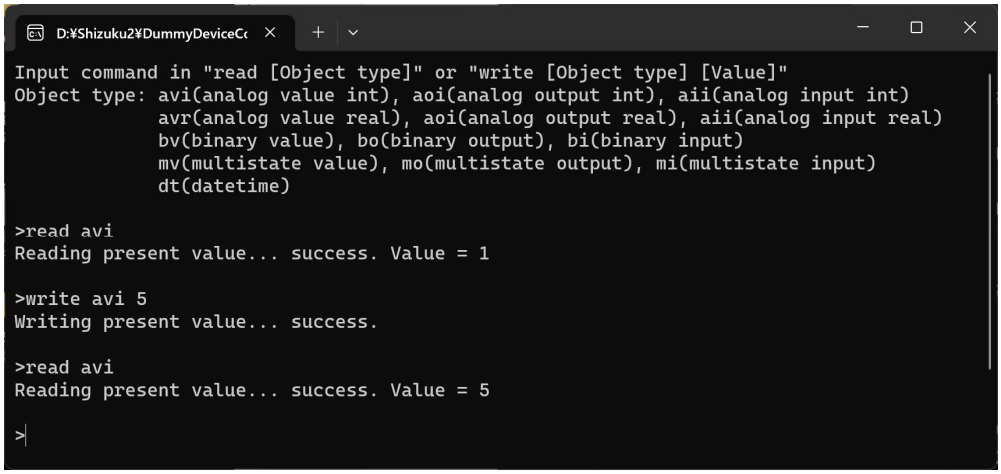


Fig. 2.6 Overwriting the analog value of the emulator

DummyDeivceController がエミュレータとは別のウィンドウで立ち上がる点は非常に重要です。エミュレータはエミュレータの中に用意された制御システムではなく、BACnet 通信を使って外部から任意の制御システムを使って操作されるということです。

従って、VRF をどのような手段で操作するのかはユーザーが自由に決めることができます。自分の得意な言語で制御プログラムを書いても良いし、複数のプログラムを作って分散して制御することもできます。このような仕組みは現実の建物と全く同じです。

2.4 計算の実行

エミュレータのウィンドウで Enter キーを入力すると、Fig. 2.7 に示すように計算が始まります。

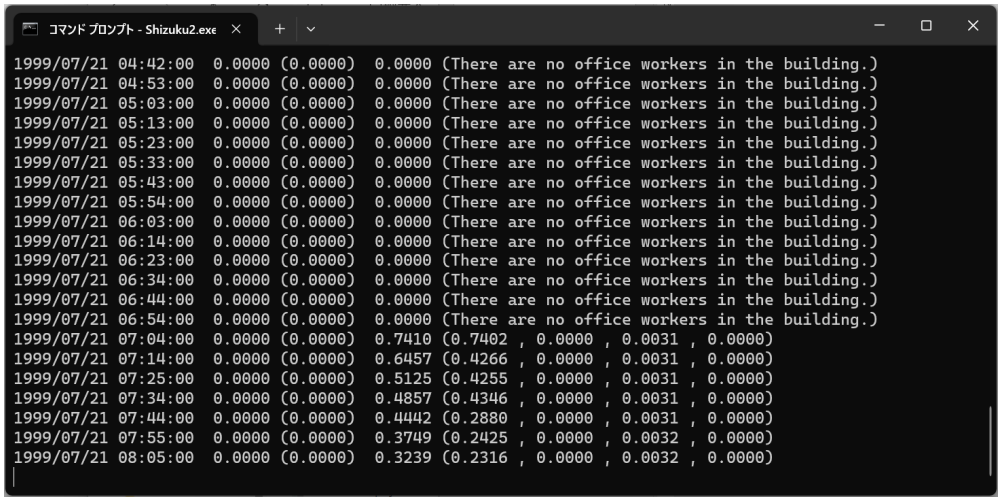


Fig. 2.7 Start emulation

標準では加速度は 600 倍に設定されています。エミュレータは夏または冬の 1 週間を模擬します。現実の 1 秒間にエミュレータは 600 秒進むため、約 17 分で計算が終わります。

計算中は、日時に続いて 7 つの数値が表示されます。

左の 2 つはエネルギーに関わる数値で、1 つ目は積算のエネルギー消費量 [GJ]、2 つ目の括弧内の数字は瞬時のエネルギー消費[GJ/h]です。この計算では VRF や換気システムを制御しておらず、すべての機器は停止しているため、0 が表示され続けます。

右の 5 つは快適性に関わる数値で、1 つ目は平均の不満足者率 [-]、括弧内の 4 つは瞬時の不満足者率で、左から順に、温冷感による不満足者率、ドラフトによる不満足者率、上下温度分布による不満足者率、空気の汚染による不満足者率、です。瞬時の不満足者率は執務者が建物内にいるときのみ、表示されます。

計算が終わると Fig. 2.8 に示すように「data」ディレクトリの下に計算結果が書き出されます。

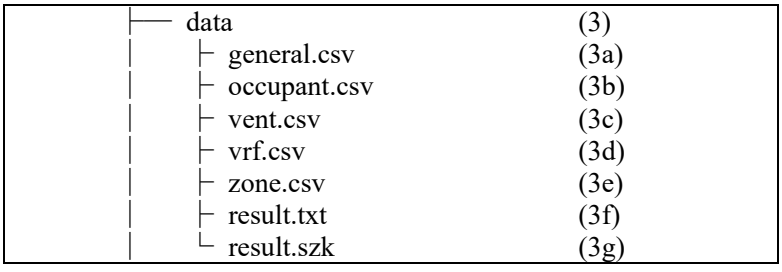


Fig. 2.8 data directory

「general.csv (3a)」には、外気条件、エネルギー消費、不満足者率など、全体に関わる情報が書き出されます。「occupant.csv (3b)」には、執務者の温冷感申告値や着衣量などが書き出されます。「vent.csv (3d)」には、部屋の CO2 濃度と換気ファンのエネルギー消費が書き出されます。「vrf.csv (3d)」には、VRF のエ

ネルギー消費や運転状態などが書き出されます。「zone.csv (3e)」には、部屋の温度と湿度が書き出されます。「result.txt (3f)」には、計算条件や成績が書き出されます。これを暗号化したファイルが「result.szk (3g)」です。

2.5 計算条件の設定

計算条件を変えるためには「setting.ini」の内容を書き換えます。内容を Fig. 2.9 に示します。

```
use_rso=1;              //Use random seed for determine occupants' behavior or not. (0:false, 1:true)
rseed_obhv=1;           //Random seed for determine occupants' behaviour randomly.
use_rsw=1;              //Use random seed for generating weather data or not. (0:false, 1:true)
rseed_w=1;              //Random seed for generating weather data.
rseed_oprm=1;           //Random seed for generating parameters of occupants' behaviour model.
timestep=60;            //Time step[sec] (0~3600)
scheduller=0;           //VRF scheduller enabled (0:disabled, 1:enabled)
controller=0;           //VRF controller type (0:Original, 1:Daikin, 2:Mitubishi Electric, 3:Toshiba, 4:Hitachi, 5:Panasonic)
weather=3;              //Weather data type (0:Load csv file, 1:Sapporo, 2:Sendai, 3:Tokyo, 4:Osaka, 5:Fukuoka, 6:Naha)
period=0;               //Simulation period (0:Summer, 1:Winter)
accelerationRate=600; //Default acceleration rate (1~)
userid=0;               //Unique ID to identify results data file
outputSpan=60;          //Time interval[sec] outputing results.
```

Fig. 2.9 Initialization file

特に重要な設定項目は「period」と「accelerationRate」です。

「period」は計算の期間を設定するための項目で、0 の場合には夏の 1 週間、1 の場合には冬の 1 週間が計算対象になります。

「accelerationRate」は計算の加速度です。デフォルトでは 600 倍としていますが、計算機の能力が高い場合にはもっと大きな数値にしても良いです。逆に計算機の能力が低くて指定の速度での計算ができない場合には、Fig. 2.10 のように「DELAYED」と表示されます。この表示が続く場合には時刻が同期できなくなるため、加速度を小さくする必要があります。

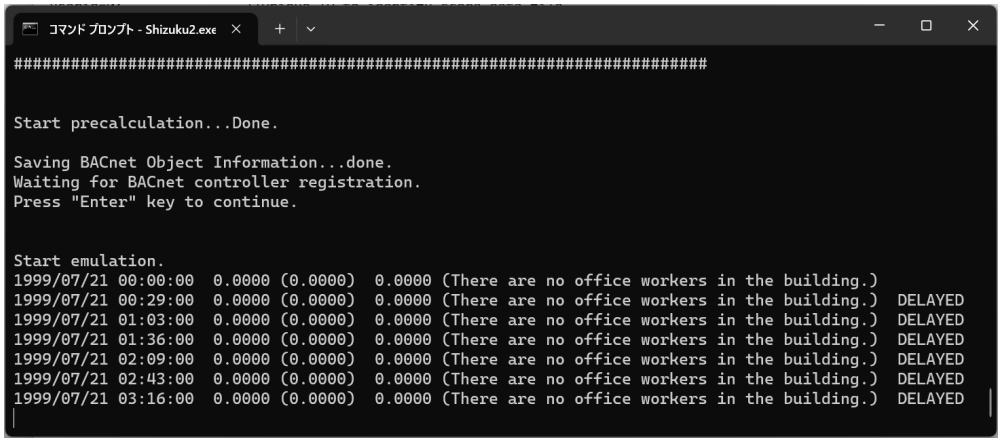


Fig. 2.10 Indication if calculation is not completed in time

第3章 Excel を使った VRF の制御

3.1 ソフトウェアの説明

エミュレータは、BACnet 通信を使って操作しますが、多くのユーザーにとって BACnet 通信プログラムの開発は経験が無いでしょう。このため、一般の期間シミュレーションプログラムと同様の方法で操作する方法が用意されています。

Fig. 3.1 にエミュレータのディレクトリを再掲します。

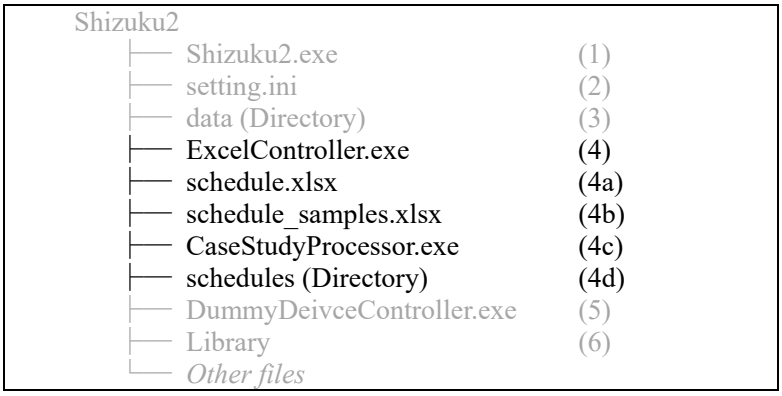


Fig. 3.1 Shizuku2 directory

「ExcelController.exe (4)」を使えば、Excel で作成したスケジュールに従って、BACnet で制御信号を送ることができます。Fig. 3.2 に ExcelController の計算処理を示します。

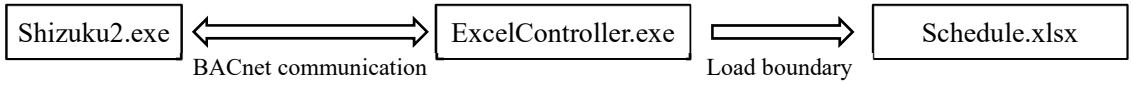
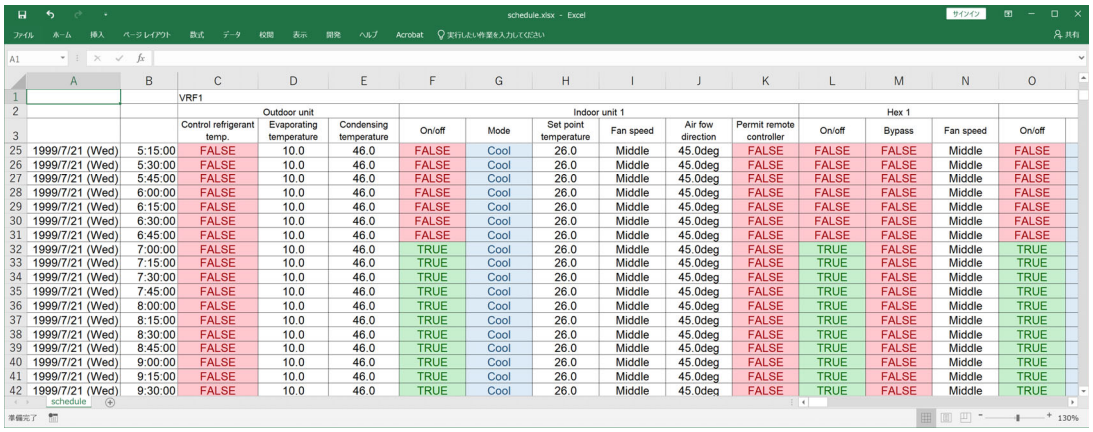


Fig. 3.2 Calculation process of ExcelController

ExcelController を起動すると、「schedule.xlsx (4a)」に記入されたスケジュールが境界条件として読み込まれます。その後、エミュレータ（Shizuku2）の日時が進み始めると、その進む速度に合わせて ExcelController から、読み込んだスケジュールに従って、制御内容がエミュレータに BACnet で伝えられます。

schedule.xlsx の内容を Fig. 3.3 に示します。



			VRF1												
			Outdoor unit			Indoor unit 1						Hex 1			
			Control refrigerant temp.	Evaporating temperature	Condensing temperature	On/off	Mode	Set point temperature	Fan speed	Air fow direction	Permit remote controller	On/off	Bypass	Fan speed	On/off
25	1999/7/21 (Wed)	5:15:00	FALSE	10.0	46.0	FALSE	Cool	26.0	Middle	45.0deg	FALSE	FALSE	FALSE	Middle	FALSE
26	1999/7/21 (Wed)	5:30:00	FALSE	10.0	46.0	FALSE	Cool	26.0	Middle	45.0deg	FALSE	FALSE	FALSE	Middle	FALSE
27	1999/7/21 (Wed)	5:45:00	FALSE	10.0	46.0	FALSE	Cool	26.0	Middle	45.0deg	FALSE	FALSE	FALSE	Middle	FALSE
28	1999/7/21 (Wed)	6:00:00	FALSE	10.0	46.0	FALSE	Cool	26.0	Middle	45.0deg	FALSE	FALSE	FALSE	Middle	FALSE
29	1999/7/21 (Wed)	6:15:00	FALSE	10.0	46.0	FALSE	Cool	26.0	Middle	45.0deg	FALSE	FALSE	FALSE	Middle	FALSE
30	1999/7/21 (Wed)	6:30:00	FALSE	10.0	46.0	FALSE	Cool	26.0	Middle	45.0deg	FALSE	FALSE	FALSE	Middle	FALSE
31	1999/7/21 (Wed)	6:45:00	FALSE	10.0	46.0	FALSE	Cool	26.0	Middle	45.0deg	FALSE	FALSE	FALSE	Middle	FALSE
32	1999/7/21 (Wed)	7:00:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
33	1999/7/21 (Wed)	7:15:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
34	1999/7/21 (Wed)	7:30:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
35	1999/7/21 (Wed)	7:45:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
36	1999/7/21 (Wed)	8:00:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
37	1999/7/21 (Wed)	8:15:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
38	1999/7/21 (Wed)	8:30:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
39	1999/7/21 (Wed)	8:45:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
40	1999/7/21 (Wed)	9:00:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
41	1999/7/21 (Wed)	9:15:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE
42	1999/7/21 (Wed)	9:30:00	FALSE	10.0	46.0	TRUE	Cool	26.0	Middle	45.0deg	FALSE	TRUE	FALSE	Middle	TRUE

Fig. 3.3 Content of the ExcelContoller

縦方向に 15 分ごとの設定値が並びます。この 15 分という時間間隔は固定値で、変更することはできません。横方向には、室外機、室内機、換気、という順序で設定項目が並びます。設定項目一覧を Table 3.1 に示します。

Table 3.1 Setting items of the ExcelController

	Name	Description	Value
Outdoor unit	Control refrigerant temp.	Whether or not the machine attempts to control the temperature of the refrigerant at a constant level.	True / False
	Evaporating temperature	The setpoint of the evaporating temperature when the temperature of the refrigerant is controlled to be constant.	Integer
	Condensing temperature	The setpoint of the condensing temperature when the temperature of the refrigerant is controlled to be constant.	Integer
Indoor unit	On/Off	On off status of the indoor unit.	True / False
	Mode	Operating mode of the indoor unit.	Cool / Heat / Fan
	Set point temperature	Room set point temperature of the indoor unit.	Real
	Fan speed	Fan speed of the indoor unit.	Low / Middle / High
	Air direction	Air direction of the indoor unit.	Horizontal ~ Vertical
	Permit remote controller	Whether or not to allow office workers to manipulate the room temperature setpoint	True / False
HEX	On/Off	On off status of the heat recovery ventilation.	True / False
	Bypass	Whether or not to supply outdoor air bypassing the heat exchanger.	True / False
	Fan speed	Fan speed of the heat recovery ventilation.	True / False

「schedule_samples.xlsx (4b)」にはいくつかのスケジュールの設定例が記載されています。

用意したスケジュールの設定例の一覧を Table 3.2 に示します。暖房運転が H1~H8、冷房運転が C1~C8、全部で 16 の設定例があります。設定の違いは、凝縮・蒸発温度を固定するかどうか、室温設定値、ファン風量、吹き出し風向、居住者にリモコン操作を許可するか否か、インテリアゾーンの室内機の運転を間引くか否か、です。

Table 3.2 Conditions of simulation cases

Case	-	Condensing / Evaporating temperature [°C]	Setpoint temperature [°C]	Fan speed†	Airflow direction [degree]	Remote control permission	Stop VRF in the interior zone
H1	heating	46.0	22.0	Middle	45.0	false	false
H2		40.0	22.0	Middle	45.0	false	false
H3		46.0	26.0	Middle	45.0	false	false
H4		46.0	22.0	Low	45.0	false	false
H5		46.0	22.0	Middle	5.0	false	false
H6		46.0	22.0	Middle	90.0	false	false
H7		46.0	22.0	Middle	45.0	true	false
H8		46.0	22.0	Middle	45.0	false	true
C1	cooling	10.0	26.0	Middle	45.0	false	false
C2		15.0	26.0	Middle	45.0	false	false
C3		10.0	22.0	Middle	45.0	false	false
C4		10.0	26.0	Low	45.0	false	false
C5		10.0	26.0	Middle	5.0	false	false
C6		10.0	26.0	Middle	90.0	false	false
C7		10.0	26.0	Middle	45.0	true	false
C8		10.0	26.0	Middle	45.0	false	true

複数のスケジュールを作って様々なケースで計算をしたいときに、手作業でスケジュールを入れ替えて計算を繰り返すのは煩雑です。この場合には、「CaseStudyProcessor.exe (4c)」を使えば、自動で複数のスケジュールファイルに対して計算ができます。Fig. 3.4 に示すように、「schedules (4d)」ディレクトリの中に 1 つ以上のスケジュールファイルを入れて「CaseStudyProcessor.exe (4c)」を実行すると、ディレクトリの中にあるスケジュールの 1 つずつを使って次々と計算が実行されます。

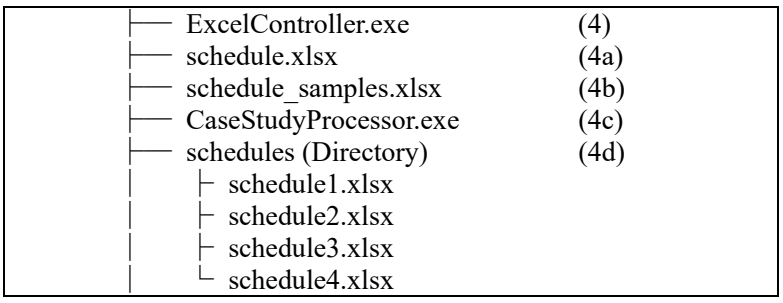


Fig. 3.4 Batch calculation method

3.2 実行例

エミュレータを起動して Fig. 2.2 の待機画面にします。この状態で ExcelController を起動すると、Schedule.xlsx が読み込まれ、Fig. 3.5 が表示されます。

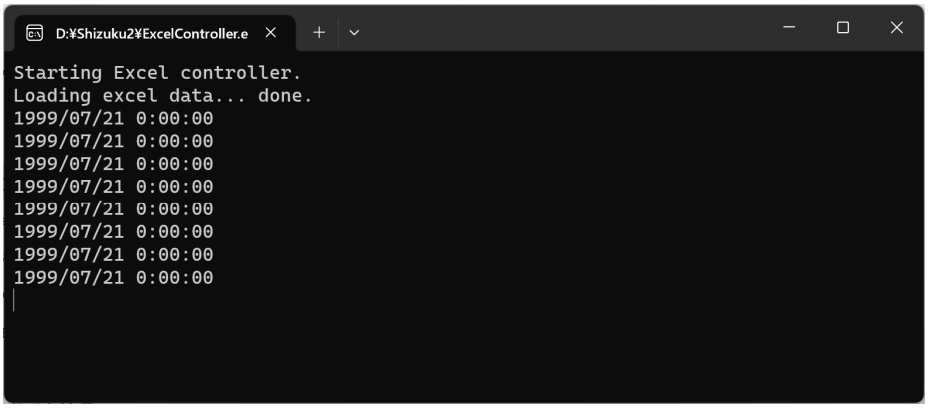


Fig. 3.5 Start up the ExcelContoller

ExcelController は毎秒、シミュレーション内の現在の日時を表示します。エミュレータはまだ待機中で、時間を進めていないため、初期値である「1999/07/21 0:00:00」が繰り返し表示されます。

エミュレータのウィンドウで Enter キーを入力すると時間が進み始めます。ExcelContoller のウィンドウで Fig. 3.6 が表示されます。いくつかの設定がエミュレータに送信されるとともに、時間が動き出したことがわかります。

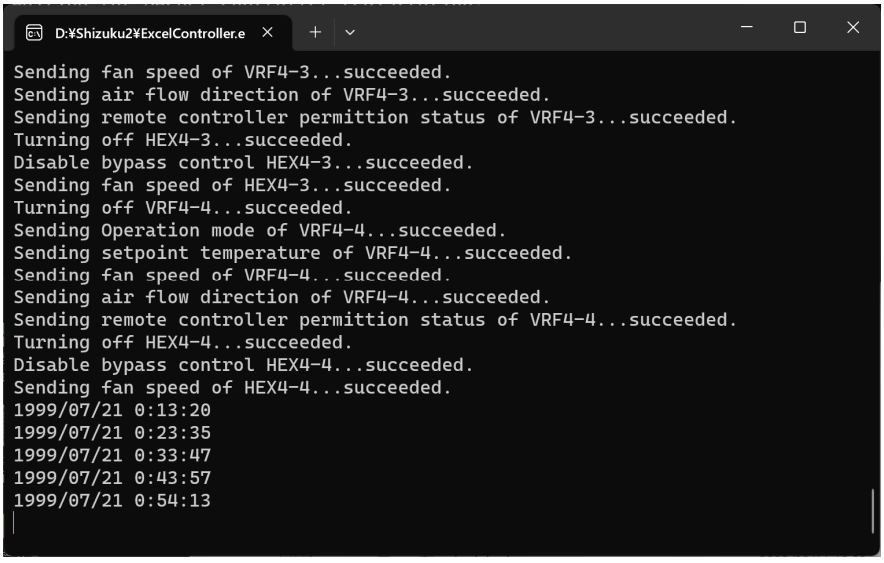


Fig. 3.6 Change settings according to schedule

しばらく放置して 7 時頃まで計算を進めます。エミュレータのウィンドウの表示が Fig. 3.7 となります。ExcelController を起動しなかった場合とは異なり、7 時頃に VRF や換気システムが起動するため、エネルギーが消費されています。室内の温湿度は制御され、換気も有効になったため、何も制御しない場合に比べると、温冷感による不満と空気の汚染による不満は小さくなります。

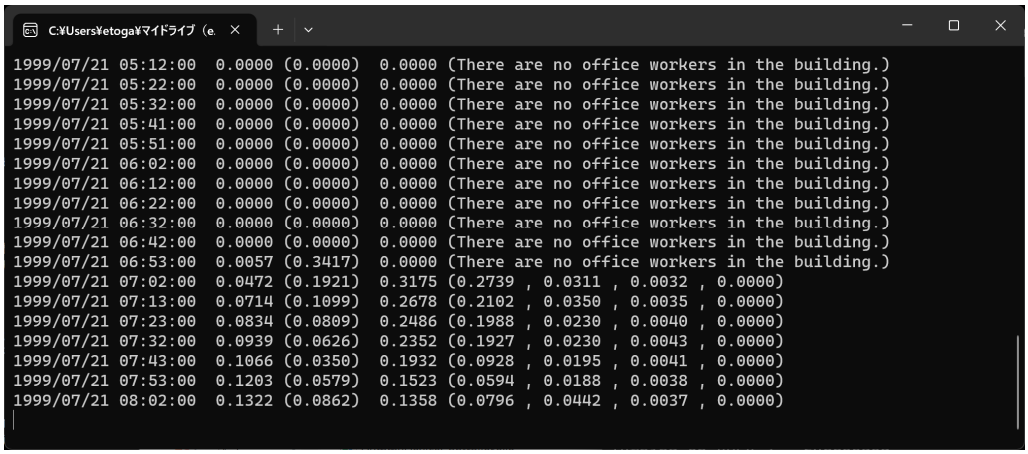


Fig. 3.7　Output of the emulator

第4章　プログラムによる VRF の制御

4.1　共通事項

BACnet 通信の仕様は ASHRAE Standard 135-2020 に規定されています。しかし、この仕様に則って 0 からプログラムを作り上げることは現実的ではないでしょう。以下に挙げるように、多くの言語で BACnet 通信のためのライブラリが開発されており、これらを使えば作業が楽になります。

C#:	BACsharp BACnet Stack	(https://bacsharp.sourceforge.net)
Java:	BACnet4J	(https://github.com/MangoAutomation/BACnet4J)
Python:	BACpypes	(https://bacpypes.readthedocs.io)
C:	BACnet Protocol Stack	(https://sourceforge.net/projects/bacnet)

BACnet ネットワークには多くの BACnet Device が繋がり、様々なデータはこれらの Device の中に分散されて保存されます。本エミュレータには Table 4.1 に示す BACnet Device が用意されています。

Table 4.1 BACnet devices in the emulator

Name	ID	PORT	Description
DateTimeController	1	47809	Manage simulation date, time, and acceleration speed.
VRFController	2	47810	Operate VRF and manage current operating conditions.
VRFScheduller	3	47811	Manage VRF operations on a schedule. Whether or not to activate this device is optional.
EnvironmentMonitor	4	47812	Monitor outdoor weather conditions and indoor temperature and humidity.
OccupantMonitor	5	47813	Monitor information related to the occupants.
VentilationController	6	47814	Operate ventilation system and manage current operating conditions.
DummyDevice	9	47817	Dummy device to try BACnet communication.

それぞれの BACnet Device は識別するために ID を持っています。一般にはそれぞれの BACnet Device は異なる IP アドレスを持つのですが、エミュレータのように複数の Device が同一の IP アドレスに存在する場合には、異なるポート番号で通信します。

BACnet Deivce の中には、複数の Object があり、Device に関連する情報は Object の中に保存されています。それは例えば VRFController であれば、室内機の発停状況、ファンの風量、消費電力、などです。これらの Object は、それぞれにインスタンス番号と Type を持ち、これらの組み合わせは重複のない ID となります。例えば VRF1 の電力消費量に関わる情報は 1021 番というインスタンス番号、Analog Input という Type として管理されています。エミュレータが持つ BACnet Device とそれぞれの Device が持つオブジェクトの一覧は Appendix 1 に整理されています。

DateTimeController はシミュレーション内の日時を管理します。現実の建物とは違い、加速度に関わる情報を持っており、この値を操作すればシミュレーションが進む速度を変えることができます。

VRFScheduller は予め書かれたプログラムに従って、標準のスケジュールで設備を運転させるための Device です。この Device を有効とするか否かは自由で、デフォルトでは無効にされています。

DummyDevice は BACnet 通信ができるかどうかを確認するためのダミーで、シミュレーション結果（快適性と省エネルギー性）には影響は与えません。

VRFController と VentilationController はそれぞれ VRF システムと換気システムの状態を監視し、運転を

変更するための Device です。また、EnvironmentMonitor と OccupantMonitor は、屋外と室内の空気の状態や執務者の温冷感を監視するための Device です。従って、これらの 4 つの Device を使って、建物の熱環境と執務者の反応を監視しつつ、VRFController と VentilationController を使って空調設備の運転方法を改善することが求められます。

上記の通り、BACnet で通信するためにはインスタンス番号と Type を特定する必要がありますが、このようなプログラムを書くのはかなり煩雑です。このため、本エミュレータ専用の BACnet 通信ライブラリを開発しました。言語は Python および.NET（C#または Basic）を使うことができます。これらのライブラリは Fig. 4.1 に示すように「Libraries」ディレクトリの中に収められています。以降の節では、これらのライブラリを使って通信する方法について解説します。

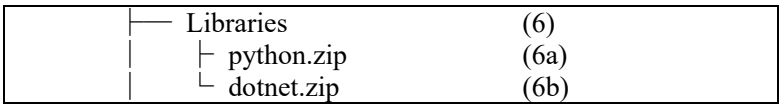


Fig. 4.1 Python and .NET library to communicate with the emulator

4.2 Python によるプログラム作成

「python.zip」を解凍して、エミュレータと通信するためのいくつかの Python プログラムのファイルを用意します。

Fig. 4.2 にライブラリに定義したクラスの関係を UML で示します。Python による BACnet 通信ライブラリとしては、bacpypes があり、PresentValueReadWriter クラスはこれを使って任意の BACnet Device が持つ現在値の読み込みと書き込み処理を実装しています。また、PresentValueReadWriter クラスはエミュレータと時刻を同期するための処理も実装しています。

この PresentValueReadWriter クラスを継承することで、エミュレータの中にある具体的な BACnet Device と通信するために 4 つのクラスが定義されています。

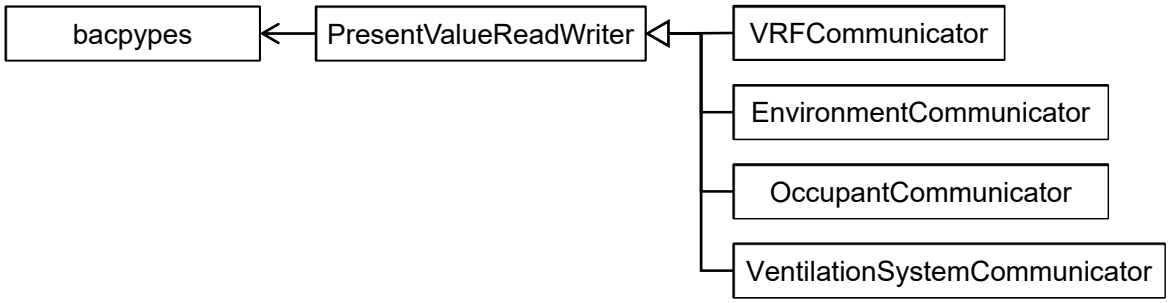


Fig. 4.2 UML of python classes to communicate with the emulator

これらのクラスを使って具体的にエミュレータを操作するプログラムを開発した例について以下で解説します。クラスに定義されたメソッドなどは以下の Web サイトでドキュメントを公開しています。

http://www.wccbo.org/lib/python

上述の通り、bacpypes を使っているため、以下のコマンドで bacpypes をインストールする必要があります。python や pip のインストール方法については解説を省きます。

```
$ pip install bacpypes
```

1) 時刻の同期

エミュレータと時刻を同期させるためのプログラムを Code 4.1 に示します。

Code 4.1 How to synchronize the time with the emulator (python)

sample1.py

```
1 import time
2 import PresentValueReadWriter
3
4 pvrw = PresentValueReadWriter.PresentValueReadWriter(10)
5 print('Subscribe COV...',end='')
6 while not pvrw.subscribe_date_time_cov():
```

```
 7       time.sleep(0.1)
 8 print('success')
 9
10 while True:
11       dt = pvrw.current_date_time()
12       print(dt.strftime('%Y/%m/%d %H:%M:%S'))
13       time.sleep(1.0)
```

4 行で、エミュレータと時刻を同期する機能を持つ PresentValueReadWriter クラスのインスタンスを作成しています。コンストラクタの引数は BACnet Device の ID です。BACnet ネットワークでは Device 同士で通信するため、エミュレータの中の Device と通信するために別の Device が必要です。この Device の ID を引数で与えます。ID の値は自由ですがネットワーク内で重複は許されないため、Table 4.1 で使われていない数値にします。1, 2, 3, 4, 5, 6, 9 は使われているため、10 としました。

6 行の「subscribe_date_time_cov」が時刻を同期するためのメソッドです。エミュレータにこの Device を登録し、エミュレータの加速度が変わったときに通知を受けるようにしています。ネットワークの状況により、この登録処理は失敗する可能性もあるため、6 行と 7 行でループさせて、成功するまで 0.1 秒間隔で登録処理を繰り返します。

登録に成功すると 11 行に示す「current_date_time」メソッドで現在の日時（datetime 型）を取得できるようになります。ここでは 12 行と 13 行で、現在の日時を 1 秒間隔で書き出しています。

Code 4.1 の実行結果を以下に示します。最初はエミュレータの時間が止まっているため日時の表示も変わりませんが、エミュレータを動かすと時間が進み始めます。

```
Subscribe COV...success
1999/07/21 00:00:00
1999/07/21 00:00:00
1999/07/21 00:00:00
1999/07/21 00:00:00
1999/07/21 00:09:13
1999/07/21 00:19:17
1999/07/21 00:29:17
1999/07/21 00:39:19
...
```

上述のように、ループで現在日時を確認し続け、適当なタイミングで設備を起動停止させることがスケジュールに従った制御の基本になります。

なお、PresentValueReadWriter クラス以外のクラスはいずれも PresentValueReadWriter クラスを継承しているため、上で解説した方法と全く同じ方法で時刻を同期することができます。

2)　室内外の環境の取得

室内外の環境を取得するためには EnvironmentCommunicator クラスを使います。プログラムを Code 4.2 に示します。4 行がコンストラクタで、引数は通信につかう Device の ID です。

8 行の「get_drybulb_temperature」は外気の乾球温度を取得するためのメソッドで、戻り値の 1 つ目は通信が成功したか否か、2 つ目は乾球温度です。通信が成功したか否かに応じて、9 行で結果を書き出しています。12 行と 16 行は外気の相対湿度と全天日射の取得処理です。

室内のゾーンごとの乾球温度を知りたい場合には、当該ゾーンを空調している VRF の室外機番号と室

内機番号を引数に使って、20 行のように「get_zone_drybulb_temperature」メソッドを呼び出します。ここでは VRF2-4 が空調しているゾーンの乾球温度を取得しています。相対湿度の場合も同様で、24 行に示すように「get_zone_relative_humidity」を使います。

Code 4.2 Monitoring indoor and outdoor environment of the emulator (python)

sample2.py

```
1  import time
2  import EnvironmentCommunicator
3
4  eCom = EnvironmentCommunicator.EnvironmentCommunicator(14)
5
6  while True:
7      print('Reading outdoor air temperature... ',end='')
8      val = eCom.get_drybulb_temperature()
9      print('{:.1f}'.format(val[1]) + ' C' if val[0] else ' Failed')
10
11     print('Reading outdoor relative humidity... ',end='')
12     val = eCom.get_relative_humidity()
13     print('{:.1f}'.format(val[1]) + ' %' if val[0] else ' Failed')
14
15     print('Reading global horizontal radiation... ',end='')
16     val = eCom.get_global_horizontal_radiation()
17     print('{:.1f}'.format(val[1]) + ' W/m2' if val[0] else ' Failed')
18
19     print('Reading drybulb temperature of zone at VRF2-4... ',end='')
20     val = eCom.get_zone_drybulb_temperature(2,4)
21     print('{:.1f}'.format(val[1]) + ' C' if val[0] else ' Failed')
22
23     print('Reading relative humidity of zone at VRF2-4... ',end='')
24     val = eCom.get_zone_relative_humidity(2,4)
25     print('{:.1f}'.format(val[1]) + ' %' if val[0] else ' Failed')
26
27     print('')
28     time.sleep(1)
```

Code 4.2 の実行結果を以下に示します。エミュレータの時間を動かせば、温湿度が変化していく様子が確認できます。

```
Reading outdoor air temperature... 25.0 C
Reading outdoor relative humidity... 50.0 %
Reading global horizontal radiation... 0.0 W/m2
Reading drybulb temperature of zone at VRF2-4... 25.0 C
Reading relative humidity of zone at VRF2-4... 50.0 %

Reading outdoor air temperature... 25.0 C
Reading outdoor relative humidity... 50.0 %
...
```

3)　執務者情報の取得

執務者の情報を取得するためには OccupantCommunicator クラスを使います。プログラムを Code 4.3 に

示します。4 行がコンストラクタで、引数は通信につかう Device の ID です。

テナント別の人数を取得するためには、8 行に示すように「get_occupant_number」メソッドを使います。OccupantCommunicator クラスには南北のテナントを区別するための列挙型「Tenant」が定義されており、これを引数として与えます。8 行は北側テナントの人数を取得する例です。戻り値は、1 つ目が通信が成功したか否か、2 つ目が人数です。

ゾーン別に在室人数を取得することもでき、12 行の「get_zone_occupant_number」メソッドを使います。この場合には、引数としてゾーン番号も与えます。ゾーンの番号は Fig. 1.4 に示したとおりです。

また、ゾーン別の平均温冷感申告値と平均着衣量は 16 行と 20 行の「get_averaged_thermal_sensation」メソッドと「get_averaged_clothing_index」メソッドで取得できます。執務者が誰もいないときの戻り値は 0 です。

24 行は、特定の執務者が在室しているか否かを判別する「is_occupant_stay_in_office」メソッドの使用例です。執務者ごとの情報を取得する場合には、南北のテナントの別に加えて、そのテナントの何番目の執務者なのかという情報を与える必要があります。24 行では、南側テナントの 1 人目の執務者について情報を取得しています。それぞれの執務者の番号と座席のゾーンは Appendix 2 で確認できます。

同様に、それぞれの執務者の温冷感や着衣量は 28 行および 32 行に示す「get_thermal_sensation」メソッドと「get_clothing_index」メソッドで取得できます。

Code 4.3 Monitoring occupant state of the emulator (python)

sample3.py

```
1  import time
2  import OccupantCommunicator as occ
3
4  oCom = occ.OccupantCommunicator(15)
5
6  while True:
7      print('Reading occupant number in north tenant... ',end='')
8      val = oCom.get_occupant_number(occ.OccupantCommunicator.Tenant.North)
9      print(str(val[1]) if val[0] else ' Failed')
10
11     print('Reading occupant number in south tenant zone-1... ',end='')
12     val = oCom.get_zone_occupant_number(occ.OccupantCommunicator.Tenant.South,1)
13     print(str(val[1]) if val[0] else ' Failed')
14
15     print('Reading averaged thermal sensation (south tenant zone-1)... ',end='')
16     val = oCom.get_averaged_thermal_sensation(occ.OccupantCommunicator.Tenant.South,1)
17     print('{:.2f}'.format(val[1]) if val[0] else ' Failed')
18
19     print('Reading averaged clothing index (south tenant zone-1)... ',end='')
20     val = oCom.get_averaged_clothing_index(occ.OccupantCommunicator.Tenant.South,1)
21     print('{:.2f}'.format(val[1]) if val[0] else ' Failed')
22
23     print('Is occupant No.1 in south tenant stay in office? ... ',end='')
24     val = oCom.is_occupant_stay_in_office(occ.OccupantCommunicator.Tenant.South, 1)
25     print(str(val[1]) if val[0] else ' Failed')
26
27     print('Reading thermal sensation of occupant No.2 in south tenant... ',end='')
28     val = oCom.get_thermal_sensation(occ.OccupantCommunicator.Tenant.South, 2)
29     print(str(val[1]) if val[0] else ' Failed')
30
```

```
31      print('Reading clothing index of occupant No.3 in south tenant... ',end='')
32      val = oCom.get_clothing_index(occ.OccupantCommunicator.Tenant.South, 3)
33      print('{:.2f}'.format(val[1]) + ' Clo' if val[0] else ' Failed')
34
35      print('')
36      time.sleep(1)
```

Code 4.3 の実行結果を以下に示します。エミュレータの時間を動かせば、執務者数や温冷感などが変化する様子が確認できます。

```
Reading occupant number in south tenant... 0
Reading occupant number in north tenant... 0
Reading occupant number in south tenant zone-1... 0
Reading averaged thermal sensation (south tenant zone-1)... 0.0
Reading averaged clothing index (south tenant zone-1)... 0.0
Is occupant No.1 in south tenant stay in office? ... False
Reading thermal sensation of occupant No.2 in south tenant... 0
Reading clothing index of occupant No.3 in south tenant... 0.00 Clo

Reading occupant number in south tenant... 0
Reading occupant number in north tenant... 0
...
```

4)　換気システムの運転変更

換気システムを制御するためには VentilationSystemCommunicator クラスを使います。プログラムを Code 4.4 に示します。4 行がコンストラクタで、引数は通信につかう Device の ID です。

テナント単位で CO2 濃度を知ることができ、8 行と 12 行のメソッドで情報を取得します。他のクラスと同様に、返り値は 1 つ目が通信が成功したか否か、2 つ目が CO2 濃度の値です。

換気用全熱交換器を起動するには、16 行に示すように「start_ventilation」メソッドを使います。全熱交換器の位置は VRF の室内機と同じため、引数として VRF の室外機番号および室内機番号を与えます。16 行では VRF1-1 の室内機と同じゾーンに設置された全熱交換器を起動しています。同様に 20 行では全熱交換器を停止しています。

全熱交換器はファンの風量を強中弱で制御することができ、現在の設定値は 24 行に示すように「get_fan_speed」メソッドで取得できます。引数は VRF の室外機と室内機の番号です。戻り値は「FanSpeed」列挙型で、「High」「Middle」「Low」の 3 つの値を取ります。設定値を変えたい場合には 28 行の「change_fan_speed」を使い、引数として VRF の室外機と室内機の番号に加えて、「FanSpeed」を与えます。28 行では「Middle」に設定しています。

Code 4.4 Controlling ventilation system of the emulator (python)

sample4.py

```
1 import time
2 import VentilationSystemCommunicator as vsc
3
4 vCom = vsc.VentilationSystemCommunicator(16)
5
6 while True:
```

```
 7      print('Reading CO2 level of south tenant... ',end='')
 8      val = vCom.get_south_tenant_CO2_level()
 9      print(str(val[1]) if val[0] else ' Failed')
10
11      print('Reading CO2 level of north tenant... ',end='')
12      val = vCom.get_north_tenant_CO2_level()
13      print(str(val[1]) if val[0] else ' Failed')
14
15      print('Turning on HEX1-1... ',end='')
16      val = vCom.start_ventilation(1,1)
17      print('success' if val[0] else ' Failed')
18
19      print('Turning off HEX1-1... ',end='')
20      val = vCom.stop_ventilation(1,1)
21      print('success' if val[0] else ' Failed')
22
23      print('Reading fan speed of HEX1-1... ',end='')
24      val = vCom.get_fan_speed(1,1)
25      print(str(val[1]) if val[0] else ' Failed')
26
27      print('Changing fan speed of HEX1-1 to Middle...',end='')
28      rslt = vCom.change_fan_speed(1,1,vsc.VentilationSystemCommunicator.FanSpeed.Middle)
29      print('success' if rslt[0] else 'failed')
30
31      print('')
32      time.sleep(1)
```

Code 4.4 の実行結果を以下に示します。エミュレータの時間を動かせば、CO2 濃度が増減する様子が確認できます。

```
Reading CO2 level of south tenant... 400
Reading CO2 level of north tenant... 400
Turning on HEX1-1... success
Turning off HEX1-1... success
Reading fan speed of HEX1-1... FanSpeed.High
Changing fan speed of HEX1-1 to Middle...success

Reading CO2 level of south tenant... 400
Reading CO2 level of north tenant... 400
...
```

5)　VRF システムの運転変更

VRF システムを制御するためには VRFSystemCommunicator クラスを使います。プログラムを Code 4.5 に示します。4 行がコンストラクタで、引数は通信につかう Device の ID です。

室内機は還り空気の乾球温度と相対湿度を計測しており、8 行と 12 行に示すメソッドで情報を取得できます。引数は室外機と室内機の番号です。この例では VRF1-2 の還り空気状態を取得しています。

室内機を起動または停止するには 16 行および 20 行に示すように「turn_on」メソッドと「turn_off」メソッドを使います。室外機は、接続されている室内機が 1 台でも起動していれば起動し、全台が停止すれば停止します。

運転モードは 24 行に示す「change_mode」メソッドで変更します。引数は室外機と室内機番号の他、

「Mode」列挙型を与えます。運転モードは「Cooling」「Heating」「ThermoOff」の 3 つから選択できます。本エミュレータの VRF は冷房と暖房を切り替えるタイプのです。冷房と暖房の運転モードが混在した場合には、若い番号の室内機の運転モードが優先されます。

ファン風量を変える場合には 32 行の「change_fan_speed」メソッドを使います。引数として「FanSpeed」列挙型を使い、「High」「Middle」「Low」から選択します。

吹き出しの向きを変える場合には36行の「change_direction」メソッドを使います。引数として「Direction」列挙体を使い、22.5 度刻みで設定できます。選択肢は「Horizontal」「Degree_225」「Degree_450」「Degree_675」「Vertical」の 5 つです。

執務者による室内機のコントローラの操作を許可するには、40 行の「permit_local_control」メソッドを使います。また、禁止するときは 44 行の「prohibit_local_control」メソッドを使います。操作が許可された場合、執務者たちが自身の温冷感に従って温度設定値を変更するようになります。自分たちで操作できることで満足度が高くなる一方で、好き勝手な設定にするためにエネルギーが多く消費される危険性もあります。

Code 4.5 Controlling VRF system of the emulator (python)

sample5.py

```
1  import time
2  import VRFSystemCommunicator as vrc
3
4  vCom = vrc.VRFSystemCommunicator(12)
5
6  while True:
7      print('Reading return air temperature of VRF1-2...',end='')
8      rslt = vCom.get_return_air_temperature(1,2)
9      print(str(rslt[1]) + ' C' if rslt[0] else 'failed')
10
11     print('Reading return air relative humidity of VRF1-2...',end='')
12     rslt = vCom.get_return_air_relative_humidity(1,2)
13     print(str(rslt[1]) + ' %' if rslt[0] else 'failed')
14
15     print('Turning on VRF1-2...',end='')
16     rslt = vCom.turn_on(1,2)
17     print('success' if rslt[0] else 'failed')
18
19     print('Turning off VRF1-2...',end='')
20     rslt = vCom.turn_off(1,2)
21     print('success' if rslt[0] else 'failed')
22
23     print('Changing mode of VRF1-2 to cooling...',end='')
24     rslt = vCom.change_mode(1,2,vrc.VRFSystemCommunicator.Mode.Cooling)
25     print('success' if rslt[0] else 'failed')
26
27     print('Changing set point temperature of VRF1-2 to 26C...',end='')
28     rslt = vCom.change_setpoint_temperature(1,2,26)
29     print('success' if rslt[0] else 'failed')
30
31     print('Changing fan speed of VRF1-2 to high...',end='')
32     rslt = vCom.change_fan_speed(1,2,vrc.VRFSystemCommunicator.FanSpeed.High)
33     print('success' if rslt[0] else 'failed')
34
```

```
35      print('Changing direction of VRF1-2 to 45degree...',end='')
36      rslt = vCom.change_direction(1,2,vrc.VRFSystemCommunicator.Direction.Degree_450)
37      print('success' if rslt[0] else 'failed')
38
39      print('Permitting local control of VRF1-2...',end='')
40      rslt = vCom.permit_local_control(1,2)
41      print('success' if rslt[0] else 'failed')
42
43      print('Prohibiting local control of VRF1-2...',end='')
44      rslt = vCom.prohibit_local_control(1,2)
45      print('success' if rslt[0] else 'failed')
46
47      print('')
48      time.sleep(1)
```

Code 4.5 の実行結果を以下に示します。エミュレータの時間を動かせば、還り温湿度が変動する様子が確認できます。

```
Reading return air temperature of VRF1-2...24.0 C
Reading return air relative humidity of VRF1-2...50.0 %
Turning on VRF1-2...success
Turning off VRF1-2...success
Changing mode of VRF1-2 to cooling...success
Changing set point temperature of VRF1-2 to 26C...success
Changing fan speed of VRF1-2 to high...success
Changing direction of VRF1-2 to 45degree...success
Permitting local control of VRF1-2...success
Prohibiting local control of VRF1-2...success

Reading return air temperature of VRF1-2...24.0 C
Reading return air relative humidity of VRF1-2...50.0 %
...
```

6)　スケジュール制御

以上で説明したプログラムを基礎に、簡単なスケジューラを作成した例を Code 4.6 に示します。

6 行と 7 行で VRF および換気システムと通信するためのインスタンスを生成します。

現在の日時に応じて空調機器を制御するため、11 行で時刻の同期を有効にします。現在の日時は 1 つしか無いため、VRF と換気システムのインスタンスの両方を同期する必要はありません。

16 行はそれぞれの VRF が持っている室内機の数です。

19 行から 75 行のループで、0.5 秒ごとに空調を制御するか否かについて判定します。18 行や 74 行で示されるように、1 つ前のループの日時を「last_dt」に保存しておき、空調停止の時間帯から運転の時間帯に変わったときに起動、逆の場合に停止させます。空調を起動する時間帯か否かは曜日と時刻から判断し、77 行から 83 行に定義したメソッドで計算します。

21、22 行で現在の日時をコンソールに出力します。

冷房と暖房のモードや設定温度などは、25~28 行に示すように、夏か冬かの季節によって切り替えます。

空調を起動する場合には 31~58 行に示すように VRF と換気システムを起動し、さらに室内機の風量や風向きなども設定します。

停止する場合の処理は 61~72 行です。

Code 4.6 Simple VRF and ventilation system scheduler for the emulator (python)

sample6.py

```python
import time, datetime
import VRFSystemCommunicator as vrc
import VentilationSystemCommunicator as vsc

def main():
  vrCom = vrc.VRFCommunicator(12)
  vsCom = vsc.VentilationSystemCommunicator(16)

  # Enable current_date_time method
  print('Subscribe COV...')
  while not vrCom.subscribe_date_time_cov():
    time.sleep(0.1)
  print('success')

  # Number of indoor units in each VRF system
  i_unit_num = [5,4,5,4]

  last_dt = vrCom.current_date_time()
  while True:
    # Output current date and time
    dt = vrCom.current_date_time()
    print(dt.strftime('%Y/%m/%d %H:%M:%S'))

    # Change mode, air flow direction, and set point temperature depends on season
    is_s = 5 <= dt.month and dt.month <= 10
    mode = vrc.VRFSystemCommunicator.Mode.Cooling if is_s else vrc.VRFSystemCommunicator.Mode.Heating
    dir = vrc.VRFSystemCommunicator.Direction.Horizontal if is_s else vrc.VRFSystemCommunicator.Direction.Vertical
    sp = 26 if is_s else 22

    # When the HVAC changed to operating hours
    if(not(is_hvac_time(last_dt)) and is_hvac_time(dt)):
      for i in range(len(i_unit_num)):
        for j in range(i_unit_num[i]):
          v_name = 'VRF' + str(i + 1) + '-' + str(j+1)

          print('Turning on ' + v_name + '...',end='')
          rslt = vrCom.turn_on(i+1,j+1)
          print('success' if rslt[0] else 'failed: ' + rslt[1])

          print('Turning on ' + v_name + ' (Ventilation)...',end='')
          rslt = vsCom.start_ventilation(i+1,j+1)
          print('success' if rslt[0] else 'failed: ' + rslt[1])

          print('Changing mode of ' + v_name + ' to ' + str(mode) + '...',end='')
          rslt = vrCom.change_mode(i+1,j+1,mode)
          print('success' if rslt[0] else 'failed: ' + rslt[1])

          print('Changing set point temperature of ' + v_name + ' to ' + str(sp) + 'C...',end='')
          rslt = vrCom.change_setpoint_temperature(i+1,j+1,sp)
          print('success' if rslt[0] else 'failed: ' + rslt[1])

          print('Changing fanspeed of ' + v_name + ' to Middle...',end='')
```

```
53                rslt = vrCom.change_fan_speed(i+1,j+1,vrc.VRFSystemCommunicator.FanSpeed.Middle)
54                print('success' if rslt[0] else 'failed: ' + rslt[1])
55
56                print('Changing direction of ' + v_name + ' to ' + str(dir) + '...',end='')
57                rslt = vrCom.change_direction(i+1,j+1,dir)
58                print('success' if rslt[0] else 'failed: ' + rslt[1])
59
60        # When the HVAC changed to stop hours
61        if(is_hvac_time(last_dt) and not(is_hvac_time(dt))):
62            for i in range(len(i_unit_num)):
63                for j in range(i_unit_num[i]):
64                    v_name = 'VRF' + str(i + 1) + '-' + str(j+1)
65
66                    print('Turning off ' + v_name + '...',end='')
67                    rslt = vrCom.turn_off(i+1,j+1)
68                    print('success' if rslt else 'failed')
69
70                    print('Turning off ' + v_name + ' (Ventilation)...',end='')
71                    rslt = vsCom.stop_ventilation(i+1,j+1)
72                    print('success' if rslt else 'failed')
73
74        last_dt = dt # Save last date and time
75        time.sleep(0.5)
76
77  def is_hvac_time(dtime):
78      start_time = datetime.time(7, 0)
79      end_time = datetime.time(19, 0)
80      now = dtime.time()
81      is_business_hour = start_time <= now <= end_time
82      is_weekday = (dtime.weekday() != 5 and dtime.weekday() != 6)
83      return is_weekday and is_business_hour
84
85  if __name__ == "__main__":
86      main()
```

7)　CO2 濃度制御

CO2 濃度に応じて換気風量を調整するプログラムを Code 4.7 に示します。

時刻の同期や空調時間帯の判定などは Code 4.6 と同様です。

21 行から 40 行がファン風量の調整の処理です。空調時間帯は 1 秒間隔で繰り返し続けます。

23 行から 26 行でテナントごとの CO2 濃度を取得し、その濃度に応じて 29、30 行でファン風量を切り替えます。CO2 濃度からファン風量を決定するメソッドは 43~49 行で定義しています。37~40 行でそれぞれの全熱交換器の風量を変更します。

Code 4.7 Demand control ventilation with CO2 level (python)

sample7.py

```
1  import time, datetime
2  import VentilationSystemCommunicator as vsc
3
4  def main():
5      vsCom = vsc.VentilationSystemCommunicator(26)
6
7      # Enable current_date_time method
```

```
    print('Subscribe COV...')
    while not vsCom.subscribe_date_time_cov():
        time.sleep(0.1)
    print('success')

    # Number of indoor units in each VRF system
    i_unit_num = [5,4,5,4]

    while True:
        # Output current date and time
        dt = vsCom.current_date_time()
        print(dt.strftime('%Y/%m/%d %H:%M:%S'))

        if(is_hvac_time(dt)):
            # Get CO2 level
            val = vsCom.get_south_tenant_CO2_level()
            south_co2 = val[1] if val[0] else 1000
            val = vsCom.get_north_tenant_CO2_level()
            north_co2 = val[1] if val[0] else 1000

            # Switch fan speed
            south_fs = get_fan_speed(south_co2)
            north_fs = get_fan_speed(north_co2)

            # Output status
            print('South tenant: ' + str(south_fs) + ' (' + str(south_co2) + ')')
            print('North tenant: ' + str(north_fs) + ' (' + str(north_co2) + ')')

            # Change fan speed
            for i in range(len(i_unit_num)):
                fs = south_fs if i == 0 or i==1 else north_fs
                for j in range(i_unit_num[i]):
                    val = vsCom.change_fan_speed(i+1,j+1,fs)
        time.sleep(1.0)

def get_fan_speed(co2_level):
    if co2_level < 600:
        return vsc.VentilationSystemCommunicator.FanSpeed.Low
    elif co2_level < 800:
        return vsc.VentilationSystemCommunicator.FanSpeed.Middle
    else:
        return vsc.VentilationSystemCommunicator.FanSpeed.High

def is_hvac_time(dtime):
    start_time = datetime.time(7, 0)
    end_time = datetime.time(19, 0)
    now = dtime.time()
    is_business_hour = start_time <= now <= end_time
    is_weekday = (dtime.weekday() != 5 and dtime.weekday() != 6)
    return is_weekday and is_business_hour

if __name__ == "__main__":
    main()
```

なお、このプログラムは全熱交換器の風量しか制御しないため、発停を操作する別のプログラムも動か

す必要があります。既に開発した Code 4.6 と同時に起動すれば良いでしょう。BACnet Device は複数の Device に対して別々に通信できるため、Fig. 4.3 に示すように制御機能を分散することができます。ただし、Device の ID（Code 4.6 の 7 行と Code 4.7 の 5 行）を重複させないようにする必要があります。

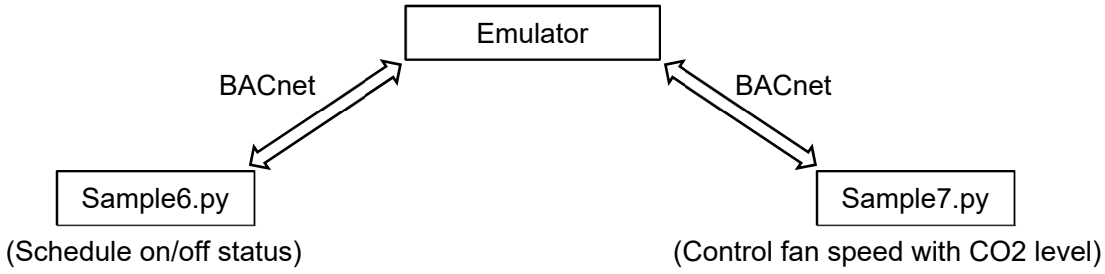


Fig. 4.3 Control the emulator by multiple BACnet Devices

Code 4.6 のみで制御した場合と、Code 4.7 を有効にした場合とで、南側テナントの CO2 濃度がどのように変化するのか、1 週間の推移を Fig. 4.4 に示します。CO2 制御があると換気量が絞られ、CO2 濃度がやや高い値で推移します。1 週間の一次エネルギー消費量も 8.73 GJ から 7.71 GJ へ、1 割強、削減されます。

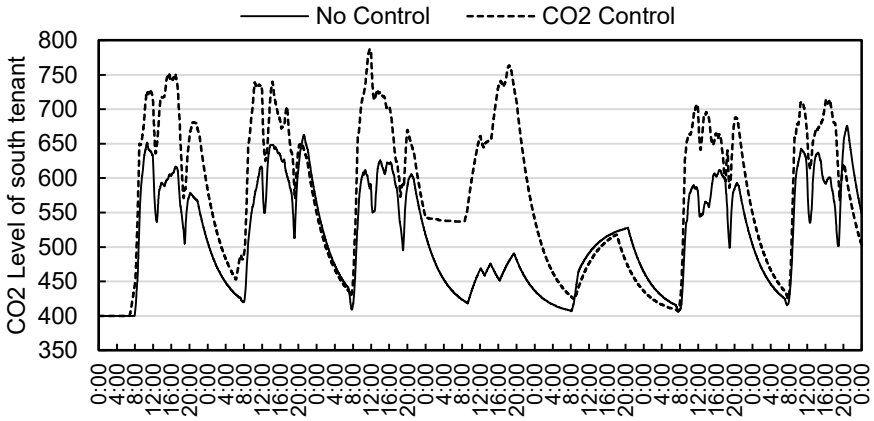


Fig. 4.4 CO2 level of south tenant with and without CO2 control

4.3 C#によるプログラム作成

本節では前節で作成した Python によるプログラムと同じ機能を持ったプログラムを C#で作成します。

Libraries ディレクトリの中にある「dotnet.zip」を解凍して、Fig. 4.5 に示す Visual Studio のソリューションを用意します。

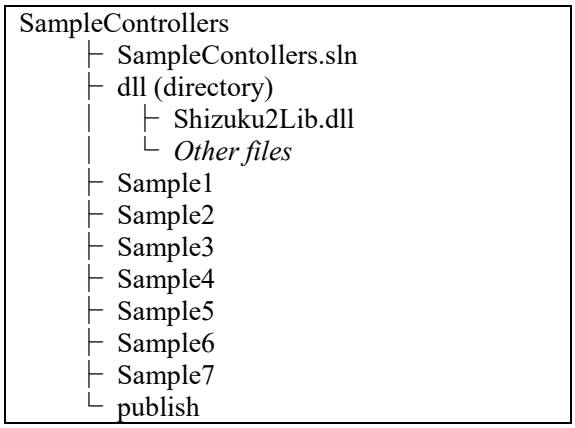


Fig. 4.5 Sample controller projects for Visual Studio

.NET 用の BACnet 通信ライブラリとしては「BACSharp」があります。「dll」ディレクトリの中にある「Shizuku2Lib.dll」は BACSharp を使ってエミュレータと通信するためのライブラリです。この dll を読み込むことで、C#または Basic で簡単にエミュレータと通信することができます。

用意されているクラスは Fig. 4.2 に示した Python のライブラリと全く同じで、基礎となる「PresentValueReadWriter」クラスと、ここから派生した 4 つの具体的な通信用クラスです。

以下、前節と同じ機能について具体的なプログラムを示していきますが、それぞれのクラスに定義されたメソッド名称はほぼ Python のライブラリと同じため、重複する説明は省きます。

各クラスのドキュメントは以下の Web サイトで確認できます。

http://www.wccbo.org/lib/dotnet

なお、以下のサンプルプログラムは、プロジェクトとして Sample1~Sample7 に入っており、実行ファイルは publish ディレクトリにまとめられています。

1) 時刻の同期

Python の例と同様に、9 行で PresentValueReadWriter クラスのインスタンスを生成します。引数は Device の ID です。C#の場合には、10 行に示すように BACnet 通信の開始を指示する StartService メソッドを呼ぶ必要があります。この処理はこれ以降のサンプルでも同様です。

13 行で COV に登録して時刻の同期を始めます。C#では 19 行に示すように「CurrentDateTime」プロパティで現在時刻を参照できるようになります。

Code 4.8 How to synchronize the time with the emulator (C#)

	Sample1/Program.cs
1	using Shizuku2.BACnet;

```
 2
 3 namespace Sample1
 4 {
 5     internal class Program
 6     {
 7         static void Main(string[] args)
 8         {
 9             PresentValueReadWriter pvrw = new PresentValueReadWriter(10);
10             pvrw.StartService();
11
12             Console.Write("Subscribe COV...");
13             while (!pvrw.SubscribeDateTimeCOV())
14                 Thread.Sleep(100);
15             Console.WriteLine("success");
16
17             while (true)
18             {
19                 DateTime dt = pvrw.CurrentDateTime;
20                 Console.WriteLine(dt.ToString("yyyy/MM/dd HH:mm:ss"));
21                 Thread.Sleep(1000);
22             }
23         }
24     }
25 }
```

2)　室内外の環境の取得

インスタンスの生成は Python と同様です。

C#の場合には通信の成功の真偽がメソッドの引数に参照渡しされます。例えば 17 行では「succeeded」に通信が成功したか否かの結果が代入されます。

Code 4.9 Monitoring indoor and outdoor environment of the emulator (C#)

Sample2/Program.cs

```
 1 using Shizuku2.BACnet;
 2
 3 namespace Sample2
 4 {
 5     internal class Program
 6     {
 7         static void Main(string[] args)
 8         {
 9             EnvironmentCommunicator eCom = new EnvironmentCommunicator(14);
10             eCom.StartService();
11
12             while (true)
13             {
14                 bool succeeded;
15
16                 Console.Write("Reading outdoor air temperature...");
17                 double dbt = eCom.GetDrybulbTemperature(out succeeded);
18                 Console.WriteLine(succeeded ? dbt.ToString("F1") : "failed");
19
20                 Console.Write("Reading outdoor air relative humidity...");
21                 double hmd = eCom.GetRelativeHumidity(out succeeded);
```

```
            Console.WriteLine(succeeded ? hmd.ToString("F1") : "failed");

            Console.Write("Reading global horizontal radiation...");
            double rad = eCom.GetGlobalHorizontalRadiation(out succeeded);
            Console.WriteLine(succeeded ? rad.ToString("F1") : "failed");

            Console.Write("Reading drybulb temperature of zone at VRF2-4...");
            double dbtZn = eCom.GetZoneDrybulbTemperature(2, 4, out succeeded);
            Console.WriteLine(succeeded ? dbtZn.ToString("F1") : "failed");

            Console.Write("Reading relative humidity of zone at VRF2-4...");
            double hmdZn = eCom.GetZoneRelativeHumidity(2, 4, out succeeded);
            Console.WriteLine(succeeded ? hmdZn.ToString("F1") : "failed");

            Console.WriteLine();
            Thread.Sleep(1000);
        }
    }
  }
}
```

3) 執務者情報の取得

Python 用のライブラリと同様です。

Code 4.10 Monitoring occupant state of the emulator (C#)

Sample3/Program.cs

```
using Shizuku2.BACnet;

namespace Sample3
{
  internal class Program
  {
    static void Main(string[] args)
    {
      OccupantCommunicator oCom = new OccupantCommunicator(15);
      oCom.StartService();

      while (true)
      {
        bool succeeded;

        Console.Write("Reading occupant number in north tenant.....");
        int oNum = oCom.GetOccupantNumber(OccupantCommunicator.Tenant.North, out succeeded);
        Console.WriteLine(succeeded ? oNum.ToString() : "failed");

        Console.Write("Reading occupant number in south tenant zone-1...");
        int oNumZ = oCom.GetOccupantNumber(OccupantCommunicator.Tenant.North, 1, out succeeded);
        Console.WriteLine(succeeded ? oNumZ.ToString() : "failed");

        Console.Write("Reading averaged thermal sensation (south tenant zone-1)...");
        float aTS = oCom.GetAveragedThermalSensation(OccupantCommunicator.Tenant.North, 1, out succeeded);
        Console.WriteLine(succeeded ? aTS.ToString("F1") : "failed");

        Console.Write("Reading averaged clothing index (south tenant zone-1)...");
```

```
            float aCI = oCom.GetAveragedClothingIndex(OccupantCommunicator.Tenant.North, 1, out succeeded);
            Console.WriteLine(succeeded ? aCI.ToString("F1") : "failed");

            Console.Write("Is occupant No.1 in south tenant stay in office? ...");
            bool ocS = oCom.IsOccupantStayInOffice(OccupantCommunicator.Tenant.North, 1, out succeeded);
            Console.WriteLine(succeeded ? ocS.ToString() : "failed");

            Console.Write("Reading thermal sensation of occupant No.2 in south tenant...");
            OccupantCommunicator.ThermalSensation ts =
               oCom.GetThermalSensation(OccupantCommunicator.Tenant.South, 2, out succeeded);
            Console.WriteLine(succeeded ? ts.ToString() : "failed");

            Console.Write("Reading clothing index of occupant No.3 in south tenant...");
            float ci = oCom.GetClothingIndex(OccupantCommunicator.Tenant.North, 3, out succeeded);
            Console.WriteLine(succeeded ? ci.ToString("F2") : "failed");

            Console.WriteLine();
            Thread.Sleep(1000);
        }
    }
  }
}
```

4)　換気システムの運転変更

Python 用のライブラリと同様です。

Code 4.11 Controlling ventilation system of the emulator (C#)

Sample4/Program.cs

```
using Shizuku2.BACnet;

namespace Sample4
{
  internal class Program
  {
    static void Main(string[] args)
    {
      VentilationSystemCommunicator vCom = new VentilationSystemCommunicator(16);
      vCom.StartService();

      while (true)
      {
        bool succeeded;

        Console.Write("Reading CO2 level of south tenant...");
        double coS = vCom.GetSouthTenantCO2Level(out succeeded);
        Console.WriteLine(succeeded ? coS.ToString() : "failed");

        Console.Write("Reading CO2 level of north tenant...");
        double coN = vCom.GetNorthTenantCO2Level(out succeeded);
        Console.WriteLine(succeeded ? coN.ToString() : "failed");

        Console.Write("Turning on HEX1-1...");
        vCom.StartVentilation(1, 1, out succeeded);
        Console.WriteLine(succeeded ? "success" : "failed");
```

```
27
28              Console.Write("Turning off HEX1-1...");
29              vCom.StopVentilation(1, 1, out succeeded);
30              Console.WriteLine(succeeded ? "success" : "failed");
31
32              Console.Write("Reading fan speed of HEX1-1...");
33              VentilationSystemCommunicator.FanSpeed fs = vCom.GetFanSpeed(1, 1, out succeeded);
34              Console.WriteLine(succeeded ? fs.ToString() : "failed");
35
36              Console.Write("Changing fan speed of HEX1-1 to Middle...");
37              vCom.ChangeFanSpeed(1, 1, VentilationSystemCommunicator.FanSpeed.Middle, out succeeded);
38              Console.WriteLine(succeeded ? "success" : "failed");
39
40              Console.WriteLine();
41              Thread.Sleep(1000);
42          }
43      }
44  }
45 }
```

5) VRF システムの運転変更

Python 用のライブラリと同様です。

Code 4.12 Controlling VRF system of the emulator (C#)

sample5.py

```
1  using Shizuku2.BACnet;
2
3  namespace Sample5
4  {
5      internal class Program
6      {
7          static void Main(string[] args)
8          {
9              VRFSystemCommunicator vCom = new VRFSystemCommunicator(12);
10             vCom.StartService();
11
12             while (true)
13             {
14                 bool succeeded;
15
16                 Console.Write("Reading return air temperature of VRF1-2...");
17                 double dbt = vCom.GetReturnAirTemperature(1, 2, out succeeded);
18                 Console.WriteLine(succeeded ? dbt.ToString("F1") : "failed");
19
20                 Console.Write("Reading return air relative humidity of VRF1-2...");
21                 double hmd = vCom.GetReturnAirRelativeHumidity(1, 2, out succeeded);
22                 Console.WriteLine(succeeded ? hmd.ToString("F1") : "failed");
23
24                 Console.Write("Turning on VRF1-2...");
25                 vCom.TurnOn(1, 2, out succeeded);
26                 Console.WriteLine(succeeded ? "success" : "failed");
27
28                 Console.Write("Turning off VRF1-2...");
29                 vCom.TurnOff(1, 2, out succeeded);
```

```
30              Console.WriteLine(succeeded ? "success" : "failed");
31
32              Console.Write("Changing mode of VRF1-2 to cooling...");
33              vCom.ChangeMode(1, 2, VRFSystemCommunicator.Mode.Cooling, out succeeded);
34              Console.WriteLine(succeeded ? "success" : "failed");
35
36              Console.Write("Changing set point temperature of VRF1-2 to 26C...");
37              vCom.ChangeSetpointTemperature(1, 2, 26, out succeeded);
38              Console.WriteLine(succeeded ? "success" : "failed");
39
40              Console.Write("Changing fan speed of VRF1-2 to high...");
41              vCom.ChangeFanSpeed(1, 2, VRFSystemCommunicator.FanSpeed.High, out succeeded);
42              Console.WriteLine(succeeded ? "success" : "failed");
43
44              Console.Write("Changing direction of VRF1-2 to 45degree...");
45              vCom.ChangeDirection(1, 2, VRFSystemCommunicator.Direction.Degree_450, out succeeded);
46              Console.WriteLine(succeeded ? "success" : "failed");
47
48              Console.Write("Permitting local control of VRF1-2...");
49              vCom.PermitLocalControl(1,2,out succeeded);
50              Console.WriteLine(succeeded ? "success" : "failed");
51
52              Console.Write("Prohibiting local control of VRF1-2...");
53              vCom.ProhibitLocalControl(1,2,out succeeded);
54              Console.WriteLine(succeeded ? "success" : "failed");
55
56              Console.WriteLine();
57              Thread.Sleep(1000);
58          }
59      }
60  }
61}
```

6)　スケジュール制御

Python 用のライブラリと同様です。

Code 4.13 Simple VRF and ventilation system scheduler for the emulator (C#)

Sample6/Program.cs

```
1 using Shizuku2.BACnet;
2
3 namespace Sample6
4 {
5     internal class Program
6     {
7         static void Main(string[] args)
8         {
9             VRFSystemCommunicator vrCom = new VRFSystemCommunicator(12);
10            VentilationSystemCommunicator vsCom = new VentilationSystemCommunicator(16);
11            vrCom.StartService();
12            vsCom.StartService();
13
14            // Enable CurrentDateTime property
15            Console.Write("Subscribe COV...");
16            while (!vrCom.SubscribeDateTimeCOV())
```

```
17              Thread.Sleep(100);
18          Console.WriteLine("success");
19
20          // Number of indoor units in each VRF system
21          int[] iUnitNum = new int[] { 5, 4, 5, 4 };
22
23          DateTime lastDt = vrCom.CurrentDateTime;
24          while (true)
25          {
26              DateTime dt = vrCom.CurrentDateTime;
27              Console.WriteLine(dt.ToString("yyyy/MM/dd HH:mm:ss"));
28
29              // Change mode, air flow direction, and set point temperature depends on season
30              bool isSum = 5 <= dt.Month && dt.Month <= 10;
31              VRFSystemCommunicator.Mode mode = VRFSystemCommunicator.Mode.Heating;
32              VRFSystemCommunicator.Direction dir = VRFSystemCommunicator.Direction.Vertical;
33              float sp = 22;
34              if (isSum)
35              {
36                  mode = VRFSystemCommunicator.Mode.Cooling;
37                  dir = VRFSystemCommunicator.Direction.Horizontal;
38                  sp = 26;
39              }
40
41              // When the HVAC changed to operating hours
42              if (!isHVACTime(lastDt) && isHVACTime(dt))
43              {
44                  for (int i = 0; i < iUnitNum.Length; i++)
45                  {
46                      for (int j = 0; j < iUnitNum[i]; j++)
47                      {
48                          bool succeeded;
49                          uint oIdx = (uint)(i + 1);
50                          uint iIdx = (uint)(j + 1);
51                          string vName = "VRF" + oIdx + "-" + iIdx;
52
53                          Console.Write("Turning on " + vName + "...");
54                          vrCom.TurnOn(oIdx, iIdx, out succeeded);
55                          Console.WriteLine(succeeded ? "success" : "failed");
56
57                          Console.Write("Turning on " + vName + "(Ventilation)...");
58                          vsCom.StartVentilation(oIdx, iIdx, out succeeded);
59                          Console.WriteLine(succeeded ? "success" : "failed");
60
61                          Console.Write("Changing mode of " + vName + " to " + mode + "...");
62                          vrCom.ChangeMode(oIdx, iIdx, mode, out succeeded);
63                          Console.WriteLine(succeeded ? "success" : "failed");
64
65                          Console.Write("Changing set point temperature of " + vName + " to " + sp + "C...");
66                          vrCom.ChangeSetpointTemperature(oIdx, iIdx, sp, out succeeded);
67                          Console.WriteLine(succeeded ? "success" : "failed");
68
69                          Console.Write("Changing fan speed of " + vName + " to Middle...");
70                          vrCom.ChangeFanSpeed(oIdx, iIdx, VRFSystemCommunicator.FanSpeed.Middle, out succeeded);
71                          Console.WriteLine(succeeded ? "success" : "failed");
72
73                          Console.Write("Changing air flow direction of " + vName + " to " + dir + "...");
74                          vrCom.ChangeDirection(oIdx, iIdx, dir, out succeeded);
```

```
                    Console.WriteLine(succeeded ? "success" : "failed");
                }
            }
        }
        // When the HVAC changed to stop hours
        else if (isHVACTime(lastDt) && !isHVACTime(dt))
        {
            for (int i = 0; i < iUnitNum.Length; i++)
            {
                for (int j = 0; j < iUnitNum[i]; j++)
                {
                    bool succeeded;
                    uint oIdx = (uint)(i + 1);
                    uint iIdx = (uint)(j + 1);
                    string vName = "VRF" + oIdx + "-" + iIdx;

                    Console.Write("Turning off " + vName + "...");
                    vrCom.TurnOff(oIdx, iIdx, out succeeded);
                    Console.WriteLine(succeeded ? "success" : "failed");

                    Console.Write("Turning off " + vName + "(Ventilation)...");
                    vsCom.StopVentilation(oIdx, iIdx, out succeeded);
                    Console.WriteLine(succeeded ? "success" : "failed");
                }
            }
        }

        lastDt = dt;
        Thread.Sleep(500);
    }
  }

  static bool isHVACTime(DateTime dt)
  {
    bool isBusinessHour = 7 <= dt.Hour && dt.Hour <= 19;
    bool isWeekday = dt.DayOfWeek != DayOfWeek.Saturday && dt.DayOfWeek != DayOfWeek.Sunday;
    return isWeekday && isBusinessHour;
  }
 }
}
```

7) CO2 濃度制御

Python 用のライブラリと同様です。

Code 4.14 Demand control ventilation with CO2 level (C#)

Sample7/Program.cs

```
using Shizuku2.BACnet;

namespace Sample7
{
  internal class Program
  {
    static void Main(string[] args)
    {
```

```
9         VentilationSystemCommunicator vsCom = new VentilationSystemCommunicator(26);
10        vsCom.StartService();
11
12        // Enable CurrentDateTime property
13        Console.Write("Subscribe COV...");
14        while (!vsCom.SubscribeDateTimeCOV())
15           Thread.Sleep(100);
16        Console.WriteLine("success");
17
18        // Number of indoor units in each VRF system
19        int[] iUnitNum = new int[] { 5, 4, 5, 4 };
20
21        while (true)
22        {
23           DateTime dt = vsCom.CurrentDateTime;
24           Console.WriteLine(dt.ToString("yyyy/MM/dd HH:mm:ss"));
25
26           // When the HVAC changed to operating hours
27           if (isHVACTime(dt))
28           {
29              for (int i = 0; i < iUnitNum.Length; i++)
30              {
31                 bool succeeded;
32                 uint southCO2 = vsCom.GetSouthTenantCO2Level(out succeeded);
33                 uint northCO2 = vsCom.GetNorthTenantCO2Level(out succeeded);
34
35                 VentilationSystemCommunicator.FanSpeed southFS = getFanSpeed(southCO2);
36                 VentilationSystemCommunicator.FanSpeed northFS = getFanSpeed(northCO2);
37
38                 Console.WriteLine("South tenant: " + southFS.ToString() + "(" + southCO2.ToString() + ")");
39                 Console.WriteLine("North tenant: " + northFS.ToString() + "(" + northCO2.ToString() + ")");
40
41                 for (int j = 0; j < iUnitNum[i]; j++)
42                 {
43                    VentilationSystemCommunicator.FanSpeed fs = i == 0 ? southFS : northFS;
44                    vsCom.ChangeFanSpeed((uint)(i + 1), (uint)(j + 1), fs, out _);
45                 }
46              }
47           }
48
49           Thread.Sleep(1000);
50        }
51     }
52
53     static VentilationSystemCommunicator.FanSpeed getFanSpeed(uint co2Level)
54     {
55        if (co2Level < 600) return VentilationSystemCommunicator.FanSpeed.Low;
56        else if (co2Level < 800) return VentilationSystemCommunicator.FanSpeed.Middle;
57        else return VentilationSystemCommunicator.FanSpeed.High;
58     }
59
60     static bool isHVACTime(DateTime dt)
61     {
62        bool isBusinessHour = 7 <= dt.Hour && dt.Hour <= 19;
63        bool isWeekday = dt.DayOfWeek != DayOfWeek.Saturday && dt.DayOfWeek != DayOfWeek.Sunday;
64        return isWeekday && isBusinessHour;
65     }
66  }
```

```
67 }
```

第5章　運用最適化の注意点

建物と VRF と執務者に関して、エネルギー消費と快適性に影響を与える点を列挙します。これらはいずれもエミュレータの内部で物理式や統計情報を使って明示的に表現されているため、VRF の運転を最適化する上で、その影響や相互のトレードオフを注意すべきです。

5.1　主に建物に関わる注意点

1) 外皮の影響により、一般にペリメータゾーンとインテリアゾーンでは熱負荷の傾向が異なる。特に冬季にはペリメータゾーンが暖房、インテリアゾーンが冷房となる場合もあり、空調による供給熱が混合して損失となる可能性がある。
2) 太陽の位置が変わるため、方位と時刻に応じて熱環境は変わる。東側は朝に日射の影響が大きく、北側は終日、影響が小さい。
3) 冬季には朝が暖房、昼が冷房というように、1 日の中で冷房と暖房の需要が入れ替わる可能性がある。これは特に外皮の影響の小さいインテリアゾーンで発生しやすい。
4) 建物の熱容量により、空調を開始してから室温が安定するまでには時間がかかる。この時間は建物の内外の温度差が大きい方が長いため、一般に夏季よりは冬季の方が大きい。
5) 建物の熱容量により、空調を停止しても室内の温度はすぐには外気と同等にはならない。
6) ペリメータゾーンには外気の影響を受けた窓や外壁があるため、インテリアゾーンとは放射温熱環境が異なる。従って、ペリメータゾーンはインテリアゾーンに比べて、空気の温湿度が同一であっても、一般に、冷房時には暑く、暖房時には寒く感じやすい。
7) 室内の空気は水平方向には混ざりやすいため、あるゾーンの室内機を停止させたとしても、隣接するゾーンの空気との混合によって温湿度はある程度は安定化する。
8) 空気は温度によって密度が異なるため、上側が暖かく、下側が冷たい上下温度分布が生まれる。水平方向に比べて垂直方向に空気は混ざりにくいため、ファンによって強制的にかき混ぜない限りは、上下温度分布は解消されにくい。
9) 全熱交換器は室内空気と外気との間で熱を交換させるシステムだが、特に冷房時には、外気温度によっては熱交換させないことで熱負荷が小さくなることもある。

5.2　主に VRF システムに関わる注意点

1) 冷房運転時に室温設定値を下げると熱負荷が大きくなり、エネルギー消費が増える。
2) 冷房運転時に蒸発温度を上げると同じ熱負荷であってもエネルギー消費が減る。一方で VRF の最大能力は小さくなり、除湿量も減る。
3) 暖房運転時に室温設定値を上げると熱負荷が大きくなり、エネルギー消費が増える。
4) 暖房運転時に凝縮温度を下げると同じ熱負荷であってもエネルギー消費が減る。一方で、VRF の最大能力は小さくなる。また、室内機の吹出温度が低くなりドラフトを感じる可能性が高まる。
5) VRF のエネルギー効率は負荷率によって変化し、一般に低負荷な方が低くなる。
6) 冷房時に室内機から吹き出した気流はまっすぐには進まず、床面に向かって曲がって落ちていく。その傾きは吹き出し温度が低いほど大きい。
7) 暖房時に室内機から吹き出した気流はまっすぐには進まず、天井に向かって曲がって上っていく。その傾きは吹き出し温度は高いほど大きい。
8) 室内機の吹き出し風速が大きいほど、気流は遠くまで到達する。従って、暖房時に風速を下げすぎる

と下部空間まで気流が到達せず、上下温度分布が大きくなる危険性が高まる。

9) 室内機の吹き出し風速が大きいほど、室内機の能力は大きくなる。ただし、冷房時には潜熱（除湿）の処理割合が減る。

10) 室内機の吹き出し角度を垂直に近づけると下部空間へ到達する気流の割合が大きくなり、上下温度分布が小さくなる。一方で、執務者にあたる気流の速度が大きくなるため、ドラフトの危険性は高まる。

5.3 主に執務者に関わる注意点

1) 人は主に乾球温度、相対湿度、平均放射温度、相対気流速度、着衣量、代謝量、の 6 つの要素に影響を受けて温冷感を感じる。

2) 人がどのような熱環境を好むのかについては、ある程度の個人差がある。

3) 室内空間の垂直方向に温度の分布が大きいとき、人は不満を感じることがある。

4) 室内機からの吹き出し気流が直接に肌に当たると、人は肌寒さによって不満を感じることがある。ただし、空間を暖かく感じている場合には、このような不満は発生しない。

5) 換気量が少なく、空気が過剰に汚れると人は不満を感じる。（本エミュレータでは CO2 濃度が 1,000 ppm を超えると人は不満を感じるようにプログラムされているが、現実の人間は CO2 濃度に対しては、それほど敏感ではない）

6) 人は、前日の執務室の温熱環境と当日の朝の外気条件を参考にして当日の着衣量を決める。また、出社後であっても上着を着る、腕まくりをする、と言った方法で、ある程度は着衣量を調整できる。

7) 人は、空調機の操作が許されていて自分で熱環境を調整できるとき、満足を感じやすくなる。

8) 人は、まず個人の着衣で温冷感を調整することを試み、それでも不満が解消されないときに空調機の温度を操作しようとする。

【参考文献】

1) ASHRAE (American Society of Heating, Refrigerating and Air-Conditioning Engineers) (2020): Standard 135-2020, BACnet - A Data Communication Protocol for Building Automation and Control Networks

Appendix 1

BACnet Devices and objects

1) Objects in the “DateTimeController” device

Inst. No.	Type	Name	Description	Initial value
1	DATETIME_VALUE	Current date and time	Current date and time on the simulation. This value might been accelerated.	1999/7/21 0:00
2	ANALOG_OUTPUT	Acceleration rate	This object is used to set the acceleration rate to run the emulator.	0
3	DATETIME_VALUE	Base real date and time	Real world date and time starting to accelerate.	2023/9/25 18:42
4	DATETIME_VALUE	Base date and time in the simulation	Date and time on the simulation when the acceleration started	1999/7/21 0:00

2) Objects in the “VRFController” device

インスタンス番号 = 1000×室外機番号 + 100×室内機番号 + Member number

ただし、システム全体に関わる情報に関しては室内機番号=0 とする。

Member number:

OnOff_Setting = 1, OnOff_Status = 2, OperationMode_Setting = 3, OperationMode_Status = 4, Setpoint_Setting = 5, Setpoint_Status = 6, MeasuredRoomTemperature = 7, MeasuredRelativeHumidity = 8, FanSpeed_Setting = 9, FanSpeed_Status = 10, AirflowDirection_Setting = 11, AirflowDirection_Status = 12, RemoteControllerPermittion_Setpoint_Setting = 13, RemoteControllerPermittion_Setpoint_Status = 14, ForcedRefrigerantTemperature_Setting = 15, ForcedRefrigerantTemperature_Status = 16, EvaporatingTemperatureSetpoint_Setting = 17, EvaporatingTemperatureSetpoint_Status = 18, CondensingTemperatureSetpoint_Setting = 19, CondensingTemperatureSetpoint_Status = 20, Electricity = 21, HeatLoad = 22

Inst. No.	Type	Name	Description	Initial value
1015	BINARY_VALUE	RefrigerantTempCtrlSetting_VRF1	This object is used to change the forced evaporating/condensing control of VRF system.	0
1016	BINARY_INPUT	RefrigerantTempCtrlStatus_VRF1	This object is used to monitor the forced evaporating/condensing control of VRF system.	0
1017	ANALOG_VALUE	EvpTempSetting_VRF1	This object is used to set the evaporating temperature of VRF system.	10
1018	ANALOG_INPUT	EvpTempStatus_VRF1	This object is used to monitor the evaporating temperature of VRF system.	10
1019	ANALOG_VALUE	CndTempSetting_VRF1	This object is used to set the condensing temperature of VRF system.	45
1020	ANALOG_INPUT	CndTempStatus_VRF1	This object is used to monitor the condensing temperature of VRF system.	45
1021	ANALOG_INPUT	Electricity_VRF1	This object is used to monitor the outdoor unit's electric consumption (fans and compressors).	0
1022	ANALOG_INPUT	HeatLoad_VRF1	This object is used to monitor the heat load of VRF system.	0
1101	BINARY_OUTPUT	OnOffCommand_VRF1-1	This object is used to start (On)/stop (Off) the indoor unit.	0
1102	BINARY_INPUT	OnOffStatus_VRF1-1	This object is used to monitor the indoor unit's On/Off status.	0
1103	MULTI_STATE_OUTPUT	ModeCommand_VRF1-1	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1104	MULTI_STATE_INPUT	ModeStatus_VRF1-1	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1105	ANALOG_VALUE	TempSPSetting_VRF1-1	This object is used to set the indoor unit's setpoint.	24
1106	ANALOG_INPUT	TempSPStatus_VRF1-1	This object is used to monitor the indoor unit's setpoint.	24
1107	ANALOG_INPUT	RoomTemp_VRF1-1	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
1108	ANALOG_INPUT	RoomRHmid_VRF1-1	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
1109	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF1-1	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1110	MULTI_STATE_INPUT	AirFlowRateStatus_VRF1-1	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1111	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF1-1	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
1112	MULTI_STATE_INPUT	AirDirectionStatus_VRF1-1	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5

1113	BINARY_VALUE	RemoteControlStart_VRF1-1	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
1114	BINARY_INPUT	RemoteControlStart_VRF1-1	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
1121	ANALOG_INPUT	Electricity_VRF1-1	This object is used to monitor the indoor unit's electric consumption.	0
1122	ANALOG_INPUT	HeatLoad_VRF1-1	This object is used to monitor the heat load of indoor unit.	0
1201	BINARY_OUTPUT	OnOffCommand_VRF1-2	This object is used to start (On)/stop (Off) the indoor unit.	0
1202	BINARY_INPUT	OnOffStatus_VRF1-2	This object is used to monitor the indoor unit's On/Off status.	0
1203	MULTI_STATE_OUTPUT	ModeCommand_VRF1-2	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1204	MULTI_STATE_INPUT	ModeStatus_VRF1-2	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1205	ANALOG_VALUE	TempSPSetting_VRF1-2	This object is used to set the indoor unit's setpoint.	24
1206	ANALOG_INPUT	TempSPStatus_VRF1-2	This object is used to monitor the indoor unit's setpoint.	24
1207	ANALOG_INPUT	RoomTemp_VRF1-2	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
1208	ANALOG_INPUT	RoomRHmid_VRF1-2	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
1209	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF1-2	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1210	MULTI_STATE_INPUT	AirFlowRateStatus_VRF1-2	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1211	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF1-2	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
1212	MULTI_STATE_INPUT	AirDirectionStatus_VRF1-2	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
1213	BINARY_VALUE	RemoteControlStart_VRF1-2	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
1214	BINARY_INPUT	RemoteControlStart_VRF1-2	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
1221	ANALOG_INPUT	Electricity_VRF1-2	This object is used to monitor the indoor unit's electric consumption.	0
1222	ANALOG_INPUT	HeatLoad_VRF1-2	This object is used to monitor the heat load of indoor unit.	0
1301	BINARY_OUTPUT	OnOffCommand_VRF1-3	This object is used to start (On)/stop (Off) the indoor unit.	0
1302	BINARY_INPUT	OnOffStatus_VRF1-3	This object is used to monitor the indoor unit's On/Off status.	0
1303	MULTI_STATE_OUTPUT	ModeCommand_VRF1-3	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1304	MULTI_STATE_INPUT	ModeStatus_VRF1-3	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1305	ANALOG_VALUE	TempSPSetting_VRF1-3	This object is used to set the indoor unit's setpoint.	24
1306	ANALOG_INPUT	TempSPStatus_VRF1-3	This object is used to monitor the indoor unit's setpoint.	24
1307	ANALOG_INPUT	RoomTemp_VRF1-3	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
1308	ANALOG_INPUT	RoomRHmid_VRF1-3	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
1309	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF1-3	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1310	MULTI_STATE_INPUT	AirFlowRateStatus_VRF1-3	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1311	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF1-3	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
1312	MULTI_STATE_INPUT	AirDirectionStatus_VRF1-3	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
1313	BINARY_VALUE	RemoteControlStart_VRF1-3	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
1314	BINARY_INPUT	RemoteControlStart_VRF1-3	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
1321	ANALOG_INPUT	Electricity_VRF1-3	This object is used to monitor the indoor unit's electric consumption.	0
1322	ANALOG_INPUT	HeatLoad_VRF1-3	This object is used to monitor the heat load of indoor unit.	0
1401	BINARY_OUTPUT	OnOffCommand_VRF1-4	This object is used to start (On)/stop (Off) the indoor unit.	0
1402	BINARY_INPUT	OnOffStatus_VRF1-4	This object is used to monitor the indoor unit's On/Off status.	0
1403	MULTI_STATE_OUTPUT	ModeCommand_VRF1-4	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1404	MULTI_STATE_INPUT	ModeStatus_VRF1-4	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1405	ANALOG_VALUE	TempSPSetting_VRF1-4	This object is used to set the indoor unit's setpoint.	24
1406	ANALOG_INPUT	TempSPStatus_VRF1-4	This object is used to monitor the indoor unit's setpoint.	24
1407	ANALOG_INPUT	RoomTemp_VRF1-4	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
1408	ANALOG_INPUT	RoomRHmid_VRF1-4	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
1409	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF1-4	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1410	MULTI_STATE_INPUT	AirFlowRateStatus_VRF1-4	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1411	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF1-4	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
1412	MULTI_STATE_INPUT	AirDirectionStatus_VRF1-4	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5

1413	BINARY_VALUE	RemoteControlStart_VRF1-4	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
1414	BINARY_INPUT	RemoteControlStart_VRF1-4	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
1421	ANALOG_INPUT	Electricity_VRF1-4	This object is used to monitor the indoor unit's electric consumption.	0
1422	ANALOG_INPUT	HeatLoad_VRF1-4	This object is used to monitor the heat load of indoor unit.	0
1501	BINARY_OUTPUT	OnOffCommand_VRF1-5	This object is used to start (On)/stop (Off) the indoor unit.	0
1502	BINARY_INPUT	OnOffStatus_VRF1-5	This object is used to monitor the indoor unit's On/Off status.	0
1503	MULTI_STATE_OUTPUT	ModeCommand_VRF1-5	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1504	MULTI_STATE_INPUT	ModeStatus_VRF1-5	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
1505	ANALOG_VALUE	TempSPSetting_VRF1-5	This object is used to set the indoor unit's setpoint.	24
1506	ANALOG_INPUT	TempSPStatus_VRF1-5	This object is used to monitor the indoor unit's setpoint.	24
1507	ANALOG_INPUT	RoomTemp_VRF1-5	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
1508	ANALOG_INPUT	RoomRHmid_VRF1-5	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
1509	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF1-5	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1510	MULTI_STATE_INPUT	AirFlowRateStatus_VRF1-5	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
1511	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF1-5	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
1512	MULTI_STATE_INPUT	AirDirectionStatus_VRF1-5	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
1513	BINARY_VALUE	RemoteControlStart_VRF1-5	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
1514	BINARY_INPUT	RemoteControlStart_VRF1-5	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
1521	ANALOG_INPUT	Electricity_VRF1-5	This object is used to monitor the indoor unit's electric consumption.	0
1522	ANALOG_INPUT	HeatLoad_VRF1-5	This object is used to monitor the heat load of indoor unit.	0
2015	BINARY_VALUE	RefrigerantTempCtrlSetting_VRF2	This object is used to change the forced evaporating/condensing control of VRF system.	0
2016	BINARY_INPUT	RefrigerantTempCtrlStatus_VRF2	This object is used to monitor the forced evaporating/condensing control of VRF system.	0
2017	ANALOG_VALUE	EvpTempSetting_VRF2	This object is used to set the evaporating temperature of VRF system.	10
2018	ANALOG_INPUT	EvpTempStatus_VRF2	This object is used to monitor the evaporating temperature of VRF system.	10
2019	ANALOG_VALUE	CndTempSetting_VRF2	This object is used to set the condensing temperature of VRF system.	45
2020	ANALOG_INPUT	CndTempStatus_VRF2	This object is used to monitor the condensing temperature of VRF system.	45
2021	ANALOG_INPUT	Electricity_VRF2	This object is used to monitor the outdoor unit's electric consumption (fans and compressors).	0
2022	ANALOG_INPUT	HeatLoad_VRF2	This object is used to monitor the heat load of VRF system.	0
2101	BINARY_OUTPUT	OnOffCommand_VRF2-1	This object is used to start (On)/stop (Off) the indoor unit.	0
2102	BINARY_INPUT	OnOffStatus_VRF2-1	This object is used to monitor the indoor unit's On/Off status.	0
2103	MULTI_STATE_OUTPUT	ModeCommand_VRF2-1	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
2104	MULTI_STATE_INPUT	ModeStatus_VRF2-1	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
2105	ANALOG_VALUE	TempSPSetting_VRF2-1	This object is used to set the indoor unit's setpoint.	24
2106	ANALOG_INPUT	TempSPStatus_VRF2-1	This object is used to monitor the indoor unit's setpoint.	24
2107	ANALOG_INPUT	RoomTemp_VRF2-1	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
2108	ANALOG_INPUT	RoomRHmid_VRF2-1	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
2109	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF2-1	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
2110	MULTI_STATE_INPUT	AirFlowRateStatus_VRF2-1	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
2111	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF2-1	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
2112	MULTI_STATE_INPUT	AirDirectionStatus_VRF2-1	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
2113	BINARY_VALUE	RemoteControlStart_VRF2-1	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
2114	BINARY_INPUT	RemoteControlStart_VRF2-1	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
2121	ANALOG_INPUT	Electricity_VRF2-1	This object is used to monitor the indoor unit's electric consumption.	0
2122	ANALOG_INPUT	HeatLoad_VRF2-1	This object is used to monitor the heat load of indoor unit.	0
2201	BINARY_OUTPUT	OnOffCommand_VRF2-2	This object is used to start (On)/stop (Off) the indoor unit.	0
2202	BINARY_INPUT	OnOffStatus_VRF2-2	This object is used to monitor the indoor unit's On/Off status.	0
2203	MULTI_STATE_OUTPUT	ModeCommand_VRF2-2	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
2204	MULTI_STATE_INPUT	ModeStatus_VRF2-2	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3

2205	ANALOG_VALUE	TempSPSetting_VRF2-2	This object is used to set the indoor unit's setpoint.	24
2206	ANALOG_INPUT	TempSPStatus_VRF2-2	This object is used to monitor the indoor unit's setpoint.	24
2207	ANALOG_INPUT	RoomTemp_VRF2-2	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
2208	ANALOG_INPUT	RoomRHmid_VRF2-2	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
2209	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF2-2	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
2210	MULTI_STATE_INPUT	AirFlowRateStatus_VRF2-2	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
2211	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF2-2	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
2212	MULTI_STATE_INPUT	AirDirectionStatus_VRF2-2	This object is used to monitor the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
2213	BINARY_VALUE	RemoteControlStart_VRF2-2	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
2214	BINARY_INPUT	RemoteControlStart_VRF2-2	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
2221	ANALOG_INPUT	Electricity_VRF2-2	This object is used to monitor the indoor unit's electric consumption.	0
2222	ANALOG_INPUT	HeatLoad_VRF2-2	This object is used to monitor the heat load of indoor unit.	0
2301	BINARY_OUTPUT	OnOffCommand_VRF2-3	This object is used to start (On)/stop (Off) the indoor unit.	0
2302	BINARY_INPUT	OnOffStatus_VRF2-3	This object is used to monitor the indoor unit's On/Off status.	0
2303	MULTI_STATE_OUTPUT	ModeCommand_VRF2-3	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
2304	MULTI_STATE_INPUT	ModeStatus_VRF2-3	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
2305	ANALOG_VALUE	TempSPSetting_VRF2-3	This object is used to set the indoor unit's setpoint.	24
2306	ANALOG_INPUT	TempSPStatus_VRF2-3	This object is used to monitor the indoor unit's setpoint.	24
2307	ANALOG_INPUT	RoomTemp_VRF2-3	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
2308	ANALOG_INPUT	RoomRHmid_VRF2-3	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
2309	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF2-3	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
2310	MULTI_STATE_INPUT	AirFlowRateStatus_VRF2-3	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
2311	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF2-3	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
2312	MULTI_STATE_INPUT	AirDirectionStatus_VRF2-3	This object is used to monitor the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
2313	BINARY_VALUE	RemoteControlStart_VRF2-3	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
2314	BINARY_INPUT	RemoteControlStart_VRF2-3	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
2321	ANALOG_INPUT	Electricity_VRF2-3	This object is used to monitor the indoor unit's electric consumption.	0
2322	ANALOG_INPUT	HeatLoad_VRF2-3	This object is used to monitor the heat load of indoor unit.	0
2401	BINARY_OUTPUT	OnOffCommand_VRF2-4	This object is used to start (On)/stop (Off) the indoor unit.	0
2402	BINARY_INPUT	OnOffStatus_VRF2-4	This object is used to monitor the indoor unit's On/Off status.	0
2403	MULTI_STATE_OUTPUT	ModeCommand_VRF2-4	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
2404	MULTI_STATE_INPUT	ModeStatus_VRF2-4	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
2405	ANALOG_VALUE	TempSPSetting_VRF2-4	This object is used to set the indoor unit's setpoint.	24
2406	ANALOG_INPUT	TempSPStatus_VRF2-4	This object is used to monitor the indoor unit's setpoint.	24
2407	ANALOG_INPUT	RoomTemp_VRF2-4	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
2408	ANALOG_INPUT	RoomRHmid_VRF2-4	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
2409	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF2-4	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
2410	MULTI_STATE_INPUT	AirFlowRateStatus_VRF2-4	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
2411	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF2-4	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
2412	MULTI_STATE_INPUT	AirDirectionStatus_VRF2-4	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
2413	BINARY_VALUE	RemoteControlStart_VRF2-4	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
2414	BINARY_INPUT	RemoteControlStart_VRF2-4	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
2421	ANALOG_INPUT	Electricity_VRF2-4	This object is used to monitor the indoor unit's electric consumption.	0
2422	ANALOG_INPUT	HeatLoad_VRF2-4	This object is used to monitor the heat load of indoor unit.	0
3015	BINARY_VALUE	RefrigerantTempCtrlSetting_VRF3	This object is used to change the forced evaporating/condensing control of VRF system.	0
3016	BINARY_INPUT	RefrigerantTempCtrlStatus_VRF3	This object is used to monitor the forced evaporating/condensing control of VRF system.	0
3017	ANALOG_VALUE	EvpTempSetting_VRF3	This object is used to set the evaporating temperature of VRF system.	10
3018	ANALOG_INPUT	EvpTempStatus_VRF3	This object is used to monitor the evaporating temperature of VRF system.	10

3019	ANALOG_VALUE	CndTempSetting_VRF3	This object is used to set the condensing temperature of VRF system.	45
3020	ANALOG_INPUT	CndTempStatus_VRF3	This object is used to monitor the condensing temperature of VRF system.	45
3021	ANALOG_INPUT	Electricity_VRF3	This object is used to monitor the outdoor unit's electric consumption (fans and compressors).	0
3022	ANALOG_INPUT	HeatLoad_VRF3	This object is used to monitor the heat load of VRF system.	0
3101	BINARY_OUTPUT	OnOffCommand_VRF3-1	This object is used to start (On)/stop (Off) the indoor unit.	0
3102	BINARY_INPUT	OnOffStatus_VRF3-1	This object is used to monitor the indoor unit's On/Off status.	0
3103	MULTI_STATE_OUTPUT	ModeCommand_VRF3-1	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3104	MULTI_STATE_INPUT	ModeStatus_VRF3-1	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3105	ANALOG_VALUE	TempSPSetting_VRF3-1	This object is used to set the indoor unit's setpoint.	24
3106	ANALOG_INPUT	TempSPStatus_VRF3-1	This object is used to monitor the indoor unit's setpoint.	24
3107	ANALOG_INPUT	RoomTemp_VRF3-1	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
3108	ANALOG_INPUT	RoomRHmid_VRF3-1	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
3109	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF3-1	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3110	MULTI_STATE_INPUT	AirFlowRateStatus_VRF3-1	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3111	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF3-1	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
3112	MULTI_STATE_INPUT	AirDirectionStatus_VRF3-1	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
3113	BINARY_VALUE	RemoteControlStart_VRF3-1	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
3114	BINARY_INPUT	RemoteControlStart_VRF3-1	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
3121	ANALOG_INPUT	Electricity_VRF3-1	This object is used to monitor the indoor unit's electric consumption.	0
3122	ANALOG_INPUT	HeatLoad_VRF3-1	This object is used to monitor the heat load of indoor unit.	0
3201	BINARY_OUTPUT	OnOffCommand_VRF3-2	This object is used to start (On)/stop (Off) the indoor unit.	0
3202	BINARY_INPUT	OnOffStatus_VRF3-2	This object is used to monitor the indoor unit's On/Off status.	0
3203	MULTI_STATE_OUTPUT	ModeCommand_VRF3-2	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3204	MULTI_STATE_INPUT	ModeStatus_VRF3-2	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3205	ANALOG_VALUE	TempSPSetting_VRF3-2	This object is used to set the indoor unit's setpoint.	24
3206	ANALOG_INPUT	TempSPStatus_VRF3-2	This object is used to monitor the indoor unit's setpoint.	24
3207	ANALOG_INPUT	RoomTemp_VRF3-2	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
3208	ANALOG_INPUT	RoomRHmid_VRF3-2	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
3209	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF3-2	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3210	MULTI_STATE_INPUT	AirFlowRateStatus_VRF3-2	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3211	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF3-2	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
3212	MULTI_STATE_INPUT	AirDirectionStatus_VRF3-2	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
3213	BINARY_VALUE	RemoteControlStart_VRF3-2	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
3214	BINARY_INPUT	RemoteControlStart_VRF3-2	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
3221	ANALOG_INPUT	Electricity_VRF3-2	This object is used to monitor the indoor unit's electric consumption.	0
3222	ANALOG_INPUT	HeatLoad_VRF3-2	This object is used to monitor the heat load of indoor unit.	0
3301	BINARY_OUTPUT	OnOffCommand_VRF3-3	This object is used to start (On)/stop (Off) the indoor unit.	0
3302	BINARY_INPUT	OnOffStatus_VRF3-3	This object is used to monitor the indoor unit's On/Off status.	0
3303	MULTI_STATE_OUTPUT	ModeCommand_VRF3-3	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3304	MULTI_STATE_INPUT	ModeStatus_VRF3-3	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3305	ANALOG_VALUE	TempSPSetting_VRF3-3	This object is used to set the indoor unit's setpoint.	24
3306	ANALOG_INPUT	TempSPStatus_VRF3-3	This object is used to monitor the indoor unit's setpoint.	24
3307	ANALOG_INPUT	RoomTemp_VRF3-3	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
3308	ANALOG_INPUT	RoomRHmid_VRF3-3	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
3309	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF3-3	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3310	MULTI_STATE_INPUT	AirFlowRateStatus_VRF3-3	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3311	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF3-3	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
3312	MULTI_STATE_INPUT	AirDirectionStatus_VRF3-3	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5

3313	BINARY_VALUE	RemoteControlStart_VRF3-3	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
3314	BINARY_INPUT	RemoteControlStart_VRF3-3	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
3321	ANALOG_INPUT	Electricity_VRF3-3	This object is used to monitor the indoor unit's electric consumption.	0
3322	ANALOG_INPUT	HeatLoad_VRF3-3	This object is used to monitor the heat load of indoor unit.	0
3401	BINARY_OUTPUT	OnOffCommand_VRF3-4	This object is used to start (On)/stop (Off) the indoor unit.	0
3402	BINARY_INPUT	OnOffStatus_VRF3-4	This object is used to monitor the indoor unit's On/Off status.	0
3403	MULTI_STATE_OUTPUT	ModeCommand_VRF3-4	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3404	MULTI_STATE_INPUT	ModeStatus_VRF3-4	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3405	ANALOG_VALUE	TempSPSetting_VRF3-4	This object is used to set the indoor unit's setpoint.	24
3406	ANALOG_INPUT	TempSPStatus_VRF3-4	This object is used to monitor the indoor unit's setpoint.	24
3407	ANALOG_INPUT	RoomTemp_VRF3-4	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
3408	ANALOG_INPUT	RoomRHmid_VRF3-4	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
3409	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF3-4	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3410	MULTI_STATE_INPUT	AirFlowRateStatus_VRF3-4	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3411	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF3-4	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
3412	MULTI_STATE_INPUT	AirDirectionStatus_VRF3-4	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
3413	BINARY_VALUE	RemoteControlStart_VRF3-4	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
3414	BINARY_INPUT	RemoteControlStart_VRF3-4	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
3421	ANALOG_INPUT	Electricity_VRF3-4	This object is used to monitor the indoor unit's electric consumption.	0
3422	ANALOG_INPUT	HeatLoad_VRF3-4	This object is used to monitor the heat load of indoor unit.	0
3501	BINARY_OUTPUT	OnOffCommand_VRF3-5	This object is used to start (On)/stop (Off) the indoor unit.	0
3502	BINARY_INPUT	OnOffStatus_VRF3-5	This object is used to monitor the indoor unit's On/Off status.	0
3503	MULTI_STATE_OUTPUT	ModeCommand_VRF3-5	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3504	MULTI_STATE_INPUT	ModeStatus_VRF3-5	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
3505	ANALOG_VALUE	TempSPSetting_VRF3-5	This object is used to set the indoor unit's setpoint.	24
3506	ANALOG_INPUT	TempSPStatus_VRF3-5	This object is used to monitor the indoor unit's setpoint.	24
3507	ANALOG_INPUT	RoomTemp_VRF3-5	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
3508	ANALOG_INPUT	RoomRHmid_VRF3-5	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
3509	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF3-5	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3510	MULTI_STATE_INPUT	AirFlowRateStatus_VRF3-5	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
3511	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF3-5	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
3512	MULTI_STATE_INPUT	AirDirectionStatus_VRF3-5	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
3513	BINARY_VALUE	RemoteControlStart_VRF3-5	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
3514	BINARY_INPUT	RemoteControlStart_VRF3-5	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
3521	ANALOG_INPUT	Electricity_VRF3-5	This object is used to monitor the indoor unit's electric consumption.	0
3522	ANALOG_INPUT	HeatLoad_VRF3-5	This object is used to monitor the heat load of indoor unit.	0
4015	BINARY_VALUE	RefrigerantTempCtrlSetting_VRF4	This object is used to change the forced evaporating/condensing control of VRF system.	0
4016	BINARY_INPUT	RefrigerantTempCtrlStatus_VRF4	This object is used to monitor the forced evaporating/condensing control of VRF system.	0
4017	ANALOG_VALUE	EvpTempSetting_VRF4	This object is used to set the evaporating temperature of VRF system.	10
4018	ANALOG_INPUT	EvpTempStatus_VRF4	This object is used to monitor the evaporating temperature of VRF system.	10
4019	ANALOG_VALUE	CndTempSetting_VRF4	This object is used to set the condensing temperature of VRF system.	45
4020	ANALOG_INPUT	CndTempStatus_VRF4	This object is used to monitor the condensing temperature of VRF system.	45
4021	ANALOG_INPUT	Electricity_VRF4	This object is used to monitor the outdoor unit's electric consumption (fans and compressors).	0
4022	ANALOG_INPUT	HeatLoad_VRF4	This object is used to monitor the heat load of VRF system.	0
4101	BINARY_OUTPUT	OnOffCommand_VRF4-1	This object is used to start (On)/stop (Off) the indoor unit.	0
4102	BINARY_INPUT	OnOffStatus_VRF4-1	This object is used to monitor the indoor unit's On/Off status.	0
4103	MULTI_STATE_OUTPUT	ModeCommand_VRF4-1	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
4104	MULTI_STATE_INPUT	ModeStatus_VRF4-1	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3

4105	ANALOG_VALUE	TempSPSetting_VRF4-1	This object is used to set the indoor unit's setpoint.	24
4106	ANALOG_INPUT	TempSPStatus_VRF4-1	This object is used to monitor the indoor unit's setpoint.	24
4107	ANALOG_INPUT	RoomTemp_VRF4-1	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
4108	ANALOG_INPUT	RoomRHmid_VRF4-1	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
4109	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF4-1	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
4110	MULTI_STATE_INPUT	AirFlowRateStatus_VRF4-1	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
4111	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF4-1	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
4112	MULTI_STATE_INPUT	AirDirectionStatus_VRF4-1	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
4113	BINARY_VALUE	RemoteControlStart_VRF4-1	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
4114	BINARY_INPUT	RemoteControlStart_VRF4-1	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
4121	ANALOG_INPUT	Electricity_VRF4-1	This object is used to monitor the indoor unit's electric consumption.	0
4122	ANALOG_INPUT	HeatLoad_VRF4-1	This object is used to monitor the heat load of indoor unit.	0
4201	BINARY_OUTPUT	OnOffCommand_VRF4-2	This object is used to start (On)/stop (Off) the indoor unit.	0
4202	BINARY_INPUT	OnOffStatus_VRF4-2	This object is used to monitor the indoor unit's On/Off status.	0
4203	MULTI_STATE_OUTPUT	ModeCommand_VRF4-2	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
4204	MULTI_STATE_INPUT	ModeStatus_VRF4-2	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
4205	ANALOG_VALUE	TempSPSetting_VRF4-2	This object is used to set the indoor unit's setpoint.	24
4206	ANALOG_INPUT	TempSPStatus_VRF4-2	This object is used to monitor the indoor unit's setpoint.	24
4207	ANALOG_INPUT	RoomTemp_VRF4-2	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
4208	ANALOG_INPUT	RoomRHmid_VRF4-2	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
4209	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF4-2	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
4210	MULTI_STATE_INPUT	AirFlowRateStatus_VRF4-2	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
4211	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF4-2	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
4212	MULTI_STATE_INPUT	AirDirectionStatus_VRF4-2	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
4213	BINARY_VALUE	RemoteControlStart_VRF4-2	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
4214	BINARY_INPUT	RemoteControlStart_VRF4-2	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
4221	ANALOG_INPUT	Electricity_VRF4-2	This object is used to monitor the indoor unit's electric consumption.	0
4222	ANALOG_INPUT	HeatLoad_VRF4-2	This object is used to monitor the heat load of indoor unit.	0
4301	BINARY_OUTPUT	OnOffCommand_VRF4-3	This object is used to start (On)/stop (Off) the indoor unit.	0
4302	BINARY_INPUT	OnOffStatus_VRF4-3	This object is used to monitor the indoor unit's On/Off status.	0
4303	MULTI_STATE_OUTPUT	ModeCommand_VRF4-3	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
4304	MULTI_STATE_INPUT	ModeStatus_VRF4-3	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
4305	ANALOG_VALUE	TempSPSetting_VRF4-3	This object is used to set the indoor unit's setpoint.	24
4306	ANALOG_INPUT	TempSPStatus_VRF4-3	This object is used to monitor the indoor unit's setpoint.	24
4307	ANALOG_INPUT	RoomTemp_VRF4-3	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
4308	ANALOG_INPUT	RoomRHmid_VRF4-3	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
4309	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF4-3	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
4310	MULTI_STATE_INPUT	AirFlowRateStatus_VRF4-3	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
4311	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF4-3	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
4312	MULTI_STATE_INPUT	AirDirectionStatus_VRF4-3	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
4313	BINARY_VALUE	RemoteControlStart_VRF4-3	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
4314	BINARY_INPUT	RemoteControlStart_VRF4-3	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
4321	ANALOG_INPUT	Electricity_VRF4-3	This object is used to monitor the indoor unit's electric consumption.	0
4322	ANALOG_INPUT	HeatLoad_VRF4-3	This object is used to monitor the heat load of indoor unit.	0
4401	BINARY_OUTPUT	OnOffCommand_VRF4-4	This object is used to start (On)/stop (Off) the indoor unit.	0
4402	BINARY_INPUT	OnOffStatus_VRF4-4	This object is used to monitor the indoor unit's On/Off status.	0
4403	MULTI_STATE_OUTPUT	ModeCommand_VRF4-4	This object is used to set an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3
4404	MULTI_STATE_INPUT	ModeStatus_VRF4-4	This object is used to monitor an indoor unit's operation mode. 1: cool; 2: heat; 3: fan	3

4405	ANALOG_VALUE	TempSPSetting_VRF4-4	This object is used to set the indoor unit's setpoint.	24
4406	ANALOG_INPUT	TempSPStatus_VRF4-4	This object is used to monitor the indoor unit's setpoint.	24
4407	ANALOG_INPUT	RoomTemp_VRF4-4	This object is used to monitor the room dry-bulb temperature detected by the indoor unit return air sensor.	24
4408	ANALOG_INPUT	RoomRHmid_VRF4-4	This object is used to monitor the room relative humidity detected by the indoor unit return air sensor.	50
4409	MULTI_STATE_OUTPUT	AirFlowRateCommand_VRF4-4	This object is used to set an indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
4410	MULTI_STATE_INPUT	AirFlowRateStatus_VRF4-4	This object is used to monitor the indoor unit's fan speed. 1: Low; 2: Middle; 3: High	2
4411	MULTI_STATE_OUTPUT	AirDirectionCommand_VRF4-4	This object is used to change the indoor unit's airflow direction. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
4412	MULTI_STATE_INPUT	AirDirectionStatus_VRF4-4	This object is used to monitor the indoor unit’s airflow direction.. 1: Horizontal; 2: 22.5deg; 3: 45deg; 4: 67.5deg; 5: Vertical	5
4413	BINARY_VALUE	RemoteControlStart_VRF4-4	This object is used to permit or prohibit the On/Off operation from the remote controller.	0
4414	BINARY_INPUT	RemoteControlStart_VRF4-4	This object is used to monitor status of permit or prohibit the On/Off operation from the remote controller.	0
4421	ANALOG_INPUT	Electricity_VRF4-4	This object is used to monitor the indoor unit's electric consumption.	0
4422	ANALOG_INPUT	HeatLoad_VRF4-4	This object is used to monitor the heat load of indoor unit.	0

3) Objects in the “EnvironmentMonitor” device

乾球温度のインスタンス番号 ＝1000×室外機番号 ＋100×室内機番号 ＋1

相対湿度のインスタンス番号 ＝1000×室外機番号 ＋100×室内機番号 ＋2

Inst. No.	Type	Name	Description	Initial value
1	ANALOG_INPUT	Outdoor_DBT	Outdoor dry-bulb temperature.	25
2	ANALOG_INPUT	Outdoor_RHMD	Outdoor relative humidity.	50
3	ANALOG_INPUT	G_Radiation	Global horizontal radiation.	0
4	ANALOG_INPUT	N_adiation	Nocturnal radiation.	0
1101	ANALOG_INPUT	DBT_VRF1-1	Dry-bulb temperature of zone at VRF1-1.	25
1102	ANALOG_INPUT	RHMD_VRF1-1	Relative humidity of zone at VRF1-1.	50
1201	ANALOG_INPUT	DBT_VRF1-2	Dry-bulb temperature of zone at VRF1-2.	25
1202	ANALOG_INPUT	RHMD_VRF1-2	Relative humidity of zone at VRF1-2.	50
1301	ANALOG_INPUT	DBT_VRF1-3	Dry-bulb temperature of zone at VRF1-3.	25
1302	ANALOG_INPUT	RHMD_VRF1-3	Relative humidity of zone at VRF1-3.	50
1401	ANALOG_INPUT	DBT_VRF1-4	Dry-bulb temperature of zone at VRF1-4.	25
1402	ANALOG_INPUT	RHMD_VRF1-4	Relative humidity of zone at VRF1-4.	50
1501	ANALOG_INPUT	DBT_VRF1-5	Dry-bulb temperature of zone at VRF1-5.	25
1502	ANALOG_INPUT	RHMD_VRF1-5	Relative humidity of zone at VRF1-5.	50
2101	ANALOG_INPUT	DBT_VRF2-1	Dry-bulb temperature of zone at VRF2-1.	25
2102	ANALOG_INPUT	RHMD_VRF2-1	Relative humidity of zone at VRF2-1.	50
2201	ANALOG_INPUT	DBT_VRF2-2	Dry-bulb temperature of zone at VRF2-2.	25
2202	ANALOG_INPUT	RHMD_VRF2-2	Relative humidity of zone at VRF2-2.	50
2301	ANALOG_INPUT	DBT_VRF2-3	Dry-bulb temperature of zone at VRF2-3.	25
2302	ANALOG_INPUT	RHMD_VRF2-3	Relative humidity of zone at VRF2-3.	50
2401	ANALOG_INPUT	DBT_VRF2-4	Dry-bulb temperature of zone at VRF2-4.	25
2402	ANALOG_INPUT	RHMD_VRF2-4	Relative humidity of zone at VRF2-4.	50
3101	ANALOG_INPUT	DBT_VRF3-1	Dry-bulb temperature of zone at VRF3-1.	25
3102	ANALOG_INPUT	RHMD_VRF3-1	Relative humidity of zone at VRF3-1.	50
3201	ANALOG_INPUT	DBT_VRF3-2	Dry-bulb temperature of zone at VRF3-2.	25

3202	ANALOG_INPUT	RHMD_VRF3-2	Relative humidity of zone at VRF3-2.	50
3301	ANALOG_INPUT	DBT_VRF3-3	Dry-bulb temperature of zone at VRF3-3.	25
3302	ANALOG_INPUT	RHMD_VRF3-3	Relative humidity of zone at VRF3-3.	50
3401	ANALOG_INPUT	DBT_VRF3-4	Dry-bulb temperature of zone at VRF3-4.	25
3402	ANALOG_INPUT	RHMD_VRF3-4	Relative humidity of zone at VRF3-4.	50
3501	ANALOG_INPUT	DBT_VRF3-5	Dry-bulb temperature of zone at VRF3-5.	25
3502	ANALOG_INPUT	RHMD_VRF3-5	Relative humidity of zone at VRF3-5.	50
4101	ANALOG_INPUT	DBT_VRF4-1	Dry-bulb temperature of zone at VRF4-1.	25
4102	ANALOG_INPUT	RHMD_VRF4-1	Relative humidity of zone at VRF4-1.	50
4201	ANALOG_INPUT	DBT_VRF4-2	Dry-bulb temperature of zone at VRF4-2.	25
4202	ANALOG_INPUT	RHMD_VRF4-2	Relative humidity of zone at VRF4-2.	50
4301	ANALOG_INPUT	DBT_VRF4-3	Dry-bulb temperature of zone at VRF4-3.	25
4302	ANALOG_INPUT	RHMD_VRF4-3	Relative humidity of zone at VRF4-3.	50
4401	ANALOG_INPUT	DBT_VRF4-4	Dry-bulb temperature of zone at VRF4-4.	25
4402	ANALOG_INPUT	RHMD_VRF4-4	Relative humidity of zone at VRF4-4.	50

4) Objects in the “OccupantMonitor” device

ゾーンの執務者数 ＝10000×テナント番号 ＋1000×ゾーン番号 ＋1

ゾーンの平均温冷感 ＝10000×テナント番号 ＋1000×ゾーン番号 ＋3

ゾーンの平均着衣量 ＝10000×テナント番号 ＋1000×ゾーン番号 ＋4

執務者の在不在のインスタンス番号 ＝10000×テナント番号 ＋10×執務者番号 ＋2

執務者の温冷感のインスタンス番号 ＝10000×テナント番号 ＋10×執務者番号 ＋3

執務者の着衣量のインスタンス番号 ＝10000×テナント番号 ＋10×執務者番号 ＋4

ただし、以下の例は執務者に関わる乱数シード（rseed_oprm）を 1 とした場合の値。乱数シードを変えると登場する執務者も変わる。

Inst. No.	Type	Name	Description	Initial value
10001	ANALOG_INPUT	Occupant number	Number of occupants stay in office (tenant-1).	0
11001	ANALOG_INPUT	Occupant number_ZN1_TNT1	Number of occupants stay in zone-1 of tenant-1	0
11003	ANALOG_INPUT	Ave_T_Sensation_ZN1_TNT1	Averaged thermal sensation of zone-1 of tenant-1	0
11004	ANALOG_INPUT	Ave_Clo_ZN1_TNT1	Averaged clothing index of zone-1 of tenant-1	0
12001	ANALOG_INPUT	Occupant number_ZN2_TNT1	Number of occupants stay in zone-2 of tenant-1	0
12003	ANALOG_INPUT	Ave_T_Sensation_ZN2_TNT1	Averaged thermal sensation of zone-2 of tenant-1	0
12004	ANALOG_INPUT	Ave_Clo_ZN2_TNT1	Averaged clothing index of zone-2 of tenant-1	0
13001	ANALOG_INPUT	Occupant number_ZN3_TNT1	Number of occupants stay in zone-3 of tenant-1	0
13003	ANALOG_INPUT	Ave_T_Sensation_ZN3_TNT1	Averaged thermal sensation of zone-3 of tenant-1	0
13004	ANALOG_INPUT	Ave_Clo_ZN3_TNT1	Averaged clothing index of zone-3 of tenant-1	0
14001	ANALOG_INPUT	Occupant number_ZN4_TNT1	Number of occupants stay in zone-4 of tenant-1	0
14003	ANALOG_INPUT	Ave_T_Sensation_ZN4_TNT1	Averaged thermal sensation of zone-4 of tenant-1	0
14004	ANALOG_INPUT	Ave_Clo_ZN4_TNT1	Averaged clothing index of zone-4 of tenant-1	0
15001	ANALOG_INPUT	Occupant number_ZN5_TNT1	Number of occupants stay in zone-5 of tenant-1	0

15003	ANALOG_INPUT	Ave_T_Sensation_ZN5_TNT1	Averaged thermal sensation of zone-5 of tenant-1	0
15004	ANALOG_INPUT	Ave_Clo_ZN5_TNT1	Averaged clothing index of zone-5 of tenant-1	0
16001	ANALOG_INPUT	Occupant number_ZN6_TNT1	Number of occupants stay in zone-6 of tenant-1	0
16003	ANALOG_INPUT	Ave_T_Sensation_ZN6_TNT1	Averaged thermal sensation of zone-6 of tenant-1	0
16004	ANALOG_INPUT	Ave_Clo_ZN6_TNT1	Averaged clothing index of zone-6 of tenant-1	0
17001	ANALOG_INPUT	Occupant number_ZN7_TNT1	Number of occupants stay in zone-7 of tenant-1	0
17003	ANALOG_INPUT	Ave_T_Sensation_ZN7_TNT1	Averaged thermal sensation of zone-7 of tenant-1	0
17004	ANALOG_INPUT	Ave_Clo_ZN7_TNT1	Averaged clothing index of zone-7 of tenant-1	0
18001	ANALOG_INPUT	Occupant number_ZN8_TNT1	Number of occupants stay in zone-8 of tenant-1	0
18003	ANALOG_INPUT	Ave_T_Sensation_ZN8_TNT1	Averaged thermal sensation of zone-8 of tenant-1	0
18004	ANALOG_INPUT	Ave_Clo_ZN8_TNT1	Averaged clothing index of zone-8 of tenant-1	0
19001	ANALOG_INPUT	Occupant number_ZN9_TNT1	Number of occupants stay in zone-9 of tenant-1	0
19003	ANALOG_INPUT	Ave_T_Sensation_ZN9_TNT1	Averaged thermal sensation of zone-9 of tenant-1	0
19004	ANALOG_INPUT	Ave_Clo_ZN9_TNT1	Averaged clothing index of zone-9 of tenant-1	0
10012	BINARY_INPUT	Availability_OC_1	Availability of occupant-1 of tenant-1 (Dana Hattersley)	0
10013	ANALOG_INPUT	T_Sensation_OC_1	Thermal sensation of occupant-1 of tenant-1 (Dana Hattersley)	0
10014	ANALOG_INPUT	Clo_OC_1	Clothing index of occupant-1 of tenant-1 (Dana Hattersley)	0
10022	BINARY_INPUT	Availability_OC_2	Availability of occupant-2 of tenant-1 (Humphrey Lock)	0
10023	ANALOG_INPUT	T_Sensation_OC_2	Thermal sensation of occupant-2 of tenant-1 (Humphrey Lock)	0
10024	ANALOG_INPUT	Clo_OC_2	Clothing index of occupant-2 of tenant-1 (Humphrey Lock)	0
10032	BINARY_INPUT	Availability_OC_3	Availability of occupant-3 of tenant-1 (Cassie Harris)	0
10033	ANALOG_INPUT	T_Sensation_OC_3	Thermal sensation of occupant-3 of tenant-1 (Cassie Harris)	0
10034	ANALOG_INPUT	Clo_OC_3	Clothing index of occupant-3 of tenant-1 (Cassie Harris)	0
10042	BINARY_INPUT	Availability_OC_4	Availability of occupant-4 of tenant-1 (Cecil Topping)	0
10043	ANALOG_INPUT	T_Sensation_OC_4	Thermal sensation of occupant-4 of tenant-1 (Cecil Topping)	0
10044	ANALOG_INPUT	Clo_OC_4	Clothing index of occupant-4 of tenant-1 (Cecil Topping)	0
10052	BINARY_INPUT	Availability_OC_5	Availability of occupant-5 of tenant-1 (Laila Black)	0
10053	ANALOG_INPUT	T_Sensation_OC_5	Thermal sensation of occupant-5 of tenant-1 (Laila Black)	0
10054	ANALOG_INPUT	Clo_OC_5	Clothing index of occupant-5 of tenant-1 (Laila Black)	0
10062	BINARY_INPUT	Availability_OC_6	Availability of occupant-6 of tenant-1 (Clive Toolson)	0
10063	ANALOG_INPUT	T_Sensation_OC_6	Thermal sensation of occupant-6 of tenant-1 (Clive Toolson)	0
10064	ANALOG_INPUT	Clo_OC_6	Clothing index of occupant-6 of tenant-1 (Clive Toolson)	0
10072	BINARY_INPUT	Availability_OC_7	Availability of occupant-7 of tenant-1 (Monique Cartwright)	0
10073	ANALOG_INPUT	T_Sensation_OC_7	Thermal sensation of occupant-7 of tenant-1 (Monique Cartwright)	0
10074	ANALOG_INPUT	Clo_OC_7	Clothing index of occupant-7 of tenant-1 (Monique Cartwright)	0
10082	BINARY_INPUT	Availability_OC_8	Availability of occupant-8 of tenant-1 (Josiah Conder)	0
10083	ANALOG_INPUT	T_Sensation_OC_8	Thermal sensation of occupant-8 of tenant-1 (Josiah Conder)	0
10084	ANALOG_INPUT	Clo_OC_8	Clothing index of occupant-8 of tenant-1 (Josiah Conder)	0
10092	BINARY_INPUT	Availability_OC_9	Availability of occupant-9 of tenant-1 (Phil Barker)	0
10093	ANALOG_INPUT	T_Sensation_OC_9	Thermal sensation of occupant-9 of tenant-1 (Phil Barker)	0
10094	ANALOG_INPUT	Clo_OC_9	Clothing index of occupant-9 of tenant-1 (Phil Barker)	0
10102	BINARY_INPUT	Availability_OC_10	Availability of occupant-10 of tenant-1 (Meredith Baldridge)	0
10103	ANALOG_INPUT	T_Sensation_OC_10	Thermal sensation of occupant-10 of tenant-1 (Meredith Baldridge)	0
10104	ANALOG_INPUT	Clo_OC_10	Clothing index of occupant-10 of tenant-1 (Meredith Baldridge)	0
10112	BINARY_INPUT	Availability_OC_11	Availability of occupant-11 of tenant-1 (Angelica Roundell)	0
10113	ANALOG_INPUT	T_Sensation_OC_11	Thermal sensation of occupant-11 of tenant-1 (Angelica Roundell)	0
10114	ANALOG_INPUT	Clo_OC_11	Clothing index of occupant-11 of tenant-1 (Angelica Roundell)	0
10122	BINARY_INPUT	Availability_OC_12	Availability of occupant-12 of tenant-1 (Hermann Rietschel)	0

10123	ANALOG_INPUT	T_Sensation_OC_12	Thermal sensation of occupant-12 of tenant-1 (Hermann Rietschel)	0
10124	ANALOG_INPUT	Clo_OC_12	Clothing index of occupant-12 of tenant-1 (Hermann Rietschel)	0
10132	BINARY_INPUT	Availability_OC_13	Availability of occupant-13 of tenant-1 (Allyn Galbraith)	0
10133	ANALOG_INPUT	T_Sensation_OC_13	Thermal sensation of occupant-13 of tenant-1 (Allyn Galbraith)	0
10134	ANALOG_INPUT	Clo_OC_13	Clothing index of occupant-13 of tenant-1 (Allyn Galbraith)	0
10142	BINARY_INPUT	Availability_OC_14	Availability of occupant-14 of tenant-1 (Wallace Sabine)	0
10143	ANALOG_INPUT	T_Sensation_OC_14	Thermal sensation of occupant-14 of tenant-1 (Wallace Sabine)	0
10144	ANALOG_INPUT	Clo_OC_14	Clothing index of occupant-14 of tenant-1 (Wallace Sabine)	0
10152	BINARY_INPUT	Availability_OC_15	Availability of occupant-15 of tenant-1 (David Midwinter)	0
10153	ANALOG_INPUT	T_Sensation_OC_15	Thermal sensation of occupant-15 of tenant-1 (David Midwinter)	0
10154	ANALOG_INPUT	Clo_OC_15	Clothing index of occupant-15 of tenant-1 (David Midwinter)	0
10162	BINARY_INPUT	Availability_OC_16	Availability of occupant-16 of tenant-1 (Rowland Rouse)	0
10163	ANALOG_INPUT	T_Sensation_OC_16	Thermal sensation of occupant-16 of tenant-1 (Rowland Rouse)	0
10164	ANALOG_INPUT	Clo_OC_16	Clothing index of occupant-16 of tenant-1 (Rowland Rouse)	0
10172	BINARY_INPUT	Availability_OC_17	Availability of occupant-17 of tenant-1 (Yuichiro Iio)	0
10173	ANALOG_INPUT	T_Sensation_OC_17	Thermal sensation of occupant-17 of tenant-1 (Yuichiro Iio)	0
10174	ANALOG_INPUT	Clo_OC_17	Clothing index of occupant-17 of tenant-1 (Yuichiro Iio)	0
10182	BINARY_INPUT	Availability_OC_18	Availability of occupant-18 of tenant-1 (Zachariah Venables-Vernon-Harcourt)	0
10183	ANALOG_INPUT	T_Sensation_OC_18	Thermal sensation of occupant-18 of tenant-1 (Zachariah Venables-Vernon-Harcourt)	0
10184	ANALOG_INPUT	Clo_OC_18	Clothing index of occupant-18 of tenant-1 (Zachariah Venables-Vernon-Harcourt)	0
10192	BINARY_INPUT	Availability_OC_19	Availability of occupant-19 of tenant-1 (Allyn Lympany)	0
10193	ANALOG_INPUT	T_Sensation_OC_19	Thermal sensation of occupant-19 of tenant-1 (Allyn Lympany)	0
10194	ANALOG_INPUT	Clo_OC_19	Clothing index of occupant-19 of tenant-1 (Allyn Lympany)	0
10202	BINARY_INPUT	Availability_OC_20	Availability of occupant-20 of tenant-1 (Daiki Kobayashi)	0
10203	ANALOG_INPUT	T_Sensation_OC_20	Thermal sensation of occupant-20 of tenant-1 (Daiki Kobayashi)	0
10204	ANALOG_INPUT	Clo_OC_20	Clothing index of occupant-20 of tenant-1 (Daiki Kobayashi)	0
10212	BINARY_INPUT	Availability_OC_21	Availability of occupant-21 of tenant-1 (Yvonne Murrills)	0
10213	ANALOG_INPUT	T_Sensation_OC_21	Thermal sensation of occupant-21 of tenant-1 (Yvonne Murrills)	0
10214	ANALOG_INPUT	Clo_OC_21	Clothing index of occupant-21 of tenant-1 (Yvonne Murrills)	0
10222	BINARY_INPUT	Availability_OC_22	Availability of occupant-22 of tenant-1 (Vince Cok)	0
10223	ANALOG_INPUT	T_Sensation_OC_22	Thermal sensation of occupant-22 of tenant-1 (Vince Cok)	0
10224	ANALOG_INPUT	Clo_OC_22	Clothing index of occupant-22 of tenant-1 (Vince Cok)	0
10232	BINARY_INPUT	Availability_OC_23	Availability of occupant-23 of tenant-1 (Niccolo Giannetti)	0
10233	ANALOG_INPUT	T_Sensation_OC_23	Thermal sensation of occupant-23 of tenant-1 (Niccolo Giannetti)	0
10234	ANALOG_INPUT	Clo_OC_23	Clothing index of occupant-23 of tenant-1 (Niccolo Giannetti)	0
10242	BINARY_INPUT	Availability_OC_24	Availability of occupant-24 of tenant-1 (Elizabeth Roundell)	0
10243	ANALOG_INPUT	T_Sensation_OC_24	Thermal sensation of occupant-24 of tenant-1 (Elizabeth Roundell)	0
10244	ANALOG_INPUT	Clo_OC_24	Clothing index of occupant-24 of tenant-1 (Elizabeth Roundell)	0
10252	BINARY_INPUT	Availability_OC_25	Availability of occupant-25 of tenant-1 (Nicola Turnbull)	0
10253	ANALOG_INPUT	T_Sensation_OC_25	Thermal sensation of occupant-25 of tenant-1 (Nicola Turnbull)	0
10254	ANALOG_INPUT	Clo_OC_25	Clothing index of occupant-25 of tenant-1 (Nicola Turnbull)	0
10262	BINARY_INPUT	Availability_OC_26	Availability of occupant-26 of tenant-1 (Masahi Momota)	0
10263	ANALOG_INPUT	T_Sensation_OC_26	Thermal sensation of occupant-26 of tenant-1 (Masahi Momota)	0
10264	ANALOG_INPUT	Clo_OC_26	Clothing index of occupant-26 of tenant-1 (Masahi Momota)	0
10272	BINARY_INPUT	Availability_OC_27	Availability of occupant-27 of tenant-1 (Jade Mollison)	0
10273	ANALOG_INPUT	T_Sensation_OC_27	Thermal sensation of occupant-27 of tenant-1 (Jade Mollison)	0
10274	ANALOG_INPUT	Clo_OC_27	Clothing index of occupant-27 of tenant-1 (Jade Mollison)	0
10282	BINARY_INPUT	Availability_OC_28	Availability of occupant-28 of tenant-1 (Linus Hanley)	0

10283	ANALOG_INPUT	T_Sensation_OC_28	Thermal sensation of occupant-28 of tenant-1 (Linus Hanley)	0
10284	ANALOG_INPUT	Clo_OC_28	Clothing index of occupant-28 of tenant-1 (Linus Hanley)	0
10292	BINARY_INPUT	Availability_OC_29	Availability of occupant-29 of tenant-1 (Valentine Elliston)	0
10293	ANALOG_INPUT	T_Sensation_OC_29	Thermal sensation of occupant-29 of tenant-1 (Valentine Elliston)	0
10294	ANALOG_INPUT	Clo_OC_29	Clothing index of occupant-29 of tenant-1 (Valentine Elliston)	0
10302	BINARY_INPUT	Availability_OC_30	Availability of occupant-30 of tenant-1 (Roman Steele)	0
10303	ANALOG_INPUT	T_Sensation_OC_30	Thermal sensation of occupant-30 of tenant-1 (Roman Steele)	0
10304	ANALOG_INPUT	Clo_OC_30	Clothing index of occupant-30 of tenant-1 (Roman Steele)	0
10312	BINARY_INPUT	Availability_OC_31	Availability of occupant-31 of tenant-1 (Savannah Biggs)	0
10313	ANALOG_INPUT	T_Sensation_OC_31	Thermal sensation of occupant-31 of tenant-1 (Savannah Biggs)	0
10314	ANALOG_INPUT	Clo_OC_31	Clothing index of occupant-31 of tenant-1 (Savannah Biggs)	0
10322	BINARY_INPUT	Availability_OC_32	Availability of occupant-32 of tenant-1 (Howard Astley)	0
10323	ANALOG_INPUT	T_Sensation_OC_32	Thermal sensation of occupant-32 of tenant-1 (Howard Astley)	0
10324	ANALOG_INPUT	Clo_OC_32	Clothing index of occupant-32 of tenant-1 (Howard Astley)	0
10332	BINARY_INPUT	Availability_OC_33	Availability of occupant-33 of tenant-1 (Masato Miyata)	0
10333	ANALOG_INPUT	T_Sensation_OC_33	Thermal sensation of occupant-33 of tenant-1 (Masato Miyata)	0
10334	ANALOG_INPUT	Clo_OC_33	Clothing index of occupant-33 of tenant-1 (Masato Miyata)	0
10342	BINARY_INPUT	Availability_OC_34	Availability of occupant-34 of tenant-1 (Aileen Winder)	0
10343	ANALOG_INPUT	T_Sensation_OC_34	Thermal sensation of occupant-34 of tenant-1 (Aileen Winder)	0
10344	ANALOG_INPUT	Clo_OC_34	Clothing index of occupant-34 of tenant-1 (Aileen Winder)	0
10352	BINARY_INPUT	Availability_OC_35	Availability of occupant-35 of tenant-1 (Landon Ackroyd)	0
10353	ANALOG_INPUT	T_Sensation_OC_35	Thermal sensation of occupant-35 of tenant-1 (Landon Ackroyd)	0
10354	ANALOG_INPUT	Clo_OC_35	Clothing index of occupant-35 of tenant-1 (Landon Ackroyd)	0
10362	BINARY_INPUT	Availability_OC_36	Availability of occupant-36 of tenant-1 (Leo Quantrill)	0
10363	ANALOG_INPUT	T_Sensation_OC_36	Thermal sensation of occupant-36 of tenant-1 (Leo Quantrill)	0
10364	ANALOG_INPUT	Clo_OC_36	Clothing index of occupant-36 of tenant-1 (Leo Quantrill)	0
10372	BINARY_INPUT	Availability_OC_37	Availability of occupant-37 of tenant-1 (Eisuke Togashi)	0
10373	ANALOG_INPUT	T_Sensation_OC_37	Thermal sensation of occupant-37 of tenant-1 (Eisuke Togashi)	0
10374	ANALOG_INPUT	Clo_OC_37	Clothing index of occupant-37 of tenant-1 (Eisuke Togashi)	0
10382	BINARY_INPUT	Availability_OC_38	Availability of occupant-38 of tenant-1 (Wilhelmina Chalmers)	0
10383	ANALOG_INPUT	T_Sensation_OC_38	Thermal sensation of occupant-38 of tenant-1 (Wilhelmina Chalmers)	0
10384	ANALOG_INPUT	Clo_OC_38	Clothing index of occupant-38 of tenant-1 (Wilhelmina Chalmers)	0
20001	ANALOG_INPUT	Occupant number	Number of occupants stay in office (tenant-2).	0
21001	ANALOG_INPUT	Occupant number_ZN1_TNT2	Number of occupants stay in zone-1 of tenant-2	0
21003	ANALOG_INPUT	Ave_T_Sensation_ZN1_TNT2	Averaged thermal sensation of zone-1 of tenant-2	0
21004	ANALOG_INPUT	Ave_Clo_ZN1_TNT2	Averaged clothing index of zone-1 of tenant-2	0
22001	ANALOG_INPUT	Occupant number_ZN2_TNT2	Number of occupants stay in zone-2 of tenant-2	0
22003	ANALOG_INPUT	Ave_T_Sensation_ZN2_TNT2	Averaged thermal sensation of zone-2 of tenant-2	0
22004	ANALOG_INPUT	Ave_Clo_ZN2_TNT2	Averaged clothing index of zone-2 of tenant-2	0
23001	ANALOG_INPUT	Occupant number_ZN3_TNT2	Number of occupants stay in zone-3 of tenant-2	0
23003	ANALOG_INPUT	Ave_T_Sensation_ZN3_TNT2	Averaged thermal sensation of zone-3 of tenant-2	0
23004	ANALOG_INPUT	Ave_Clo_ZN3_TNT2	Averaged clothing index of zone-3 of tenant-2	0
24001	ANALOG_INPUT	Occupant number_ZN4_TNT2	Number of occupants stay in zone-4 of tenant-2	0
24003	ANALOG_INPUT	Ave_T_Sensation_ZN4_TNT2	Averaged thermal sensation of zone-4 of tenant-2	0
24004	ANALOG_INPUT	Ave_Clo_ZN4_TNT2	Averaged clothing index of zone-4 of tenant-2	0
25001	ANALOG_INPUT	Occupant number_ZN5_TNT2	Number of occupants stay in zone-5 of tenant-2	0
25003	ANALOG_INPUT	Ave_T_Sensation_ZN5_TNT2	Averaged thermal sensation of zone-5 of tenant-2	0
25004	ANALOG_INPUT	Ave_Clo_ZN5_TNT2	Averaged clothing index of zone-5 of tenant-2	0

26001	ANALOG_INPUT	Occupant number_ZN6_TNT2	Number of occupants stay in zone-6 of tenant-2	0
26003	ANALOG_INPUT	Ave_T_Sensation_ZN6_TNT2	Averaged thermal sensation of zone-6 of tenant-2	0
26004	ANALOG_INPUT	Ave_Clo_ZN6_TNT2	Averaged clothing index of zone-6 of tenant-2	0
27001	ANALOG_INPUT	Occupant number_ZN7_TNT2	Number of occupants stay in zone-7 of tenant-2	0
27003	ANALOG_INPUT	Ave_T_Sensation_ZN7_TNT2	Averaged thermal sensation of zone-7 of tenant-2	0
27004	ANALOG_INPUT	Ave_Clo_ZN7_TNT2	Averaged clothing index of zone-7 of tenant-2	0
28001	ANALOG_INPUT	Occupant number_ZN8_TNT2	Number of occupants stay in zone-8 of tenant-2	0
28003	ANALOG_INPUT	Ave_T_Sensation_ZN8_TNT2	Averaged thermal sensation of zone-8 of tenant-2	0
28004	ANALOG_INPUT	Ave_Clo_ZN8_TNT2	Averaged clothing index of zone-8 of tenant-2	0
29001	ANALOG_INPUT	Occupant number_ZN9_TNT2	Number of occupants stay in zone-9 of tenant-2	0
29003	ANALOG_INPUT	Ave_T_Sensation_ZN9_TNT2	Averaged thermal sensation of zone-9 of tenant-2	0
29004	ANALOG_INPUT	Ave_Clo_ZN9_TNT2	Averaged clothing index of zone-9 of tenant-2	0
20012	BINARY_INPUT	Availability_OC_1	Availability of occupant-1 of tenant-2 (Kim Collingwood)	0
20013	ANALOG_INPUT	T_Sensation_OC_1	Thermal sensation of occupant-1 of tenant-2 (Kim Collingwood)	0
20014	ANALOG_INPUT	Clo_OC_1	Clothing index of occupant-1 of tenant-2 (Kim Collingwood)	0
20022	BINARY_INPUT	Availability_OC_2	Availability of occupant-2 of tenant-2 (Takahiro Ueno)	0
20023	ANALOG_INPUT	T_Sensation_OC_2	Thermal sensation of occupant-2 of tenant-2 (Takahiro Ueno)	0
20024	ANALOG_INPUT	Clo_OC_2	Clothing index of occupant-2 of tenant-2 (Takahiro Ueno)	0
20032	BINARY_INPUT	Availability_OC_3	Availability of occupant-3 of tenant-2 (Kimberly Holder)	0
20033	ANALOG_INPUT	T_Sensation_OC_3	Thermal sensation of occupant-3 of tenant-2 (Kimberly Holder)	0
20034	ANALOG_INPUT	Clo_OC_3	Clothing index of occupant-3 of tenant-2 (Kimberly Holder)	0
20042	BINARY_INPUT	Availability_OC_4	Availability of occupant-4 of tenant-2 (Sophie Coffin)	0
20043	ANALOG_INPUT	T_Sensation_OC_4	Thermal sensation of occupant-4 of tenant-2 (Sophie Coffin)	0
20044	ANALOG_INPUT	Clo_OC_4	Clothing index of occupant-4 of tenant-2 (Sophie Coffin)	0
20052	BINARY_INPUT	Availability_OC_5	Availability of occupant-5 of tenant-2 (Rolla Carpenter)	0
20053	ANALOG_INPUT	T_Sensation_OC_5	Thermal sensation of occupant-5 of tenant-2 (Rolla Carpenter)	0
20054	ANALOG_INPUT	Clo_OC_5	Clothing index of occupant-5 of tenant-2 (Rolla Carpenter)	0
20062	BINARY_INPUT	Availability_OC_6	Availability of occupant-6 of tenant-2 (Pauline Gooding)	0
20063	ANALOG_INPUT	T_Sensation_OC_6	Thermal sensation of occupant-6 of tenant-2 (Pauline Gooding)	0
20064	ANALOG_INPUT	Clo_OC_6	Clothing index of occupant-6 of tenant-2 (Pauline Gooding)	0
20072	BINARY_INPUT	Availability_OC_7	Availability of occupant-7 of tenant-2 (Sei Nagashima)	0
20073	ANALOG_INPUT	T_Sensation_OC_7	Thermal sensation of occupant-7 of tenant-2 (Sei Nagashima)	0
20074	ANALOG_INPUT	Clo_OC_7	Clothing index of occupant-7 of tenant-2 (Sei Nagashima)	0
20082	BINARY_INPUT	Availability_OC_8	Availability of occupant-8 of tenant-2 (Louisa Street)	0
20083	ANALOG_INPUT	T_Sensation_OC_8	Thermal sensation of occupant-8 of tenant-2 (Louisa Street)	0
20084	ANALOG_INPUT	Clo_OC_8	Clothing index of occupant-8 of tenant-2 (Louisa Street)	0
20092	BINARY_INPUT	Availability_OC_9	Availability of occupant-9 of tenant-2 (Lindsay Buckler)	0
20093	ANALOG_INPUT	T_Sensation_OC_9	Thermal sensation of occupant-9 of tenant-2 (Lindsay Buckler)	0
20094	ANALOG_INPUT	Clo_OC_9	Clothing index of occupant-9 of tenant-2 (Lindsay Buckler)	0
20102	BINARY_INPUT	Availability_OC_10	Availability of occupant-10 of tenant-2 (Katsuyuki Edahiro)	0
20103	ANALOG_INPUT	T_Sensation_OC_10	Thermal sensation of occupant-10 of tenant-2 (Katsuyuki Edahiro)	0
20104	ANALOG_INPUT	Clo_OC_10	Clothing index of occupant-10 of tenant-2 (Katsuyuki Edahiro)	0
20112	BINARY_INPUT	Availability_OC_11	Availability of occupant-11 of tenant-2 (Carey Blanchfield)	0
20113	ANALOG_INPUT	T_Sensation_OC_11	Thermal sensation of occupant-11 of tenant-2 (Carey Blanchfield)	0
20114	ANALOG_INPUT	Clo_OC_11	Clothing index of occupant-11 of tenant-2 (Carey Blanchfield)	0
20122	BINARY_INPUT	Availability_OC_12	Availability of occupant-12 of tenant-2 (Cordelia Woodson)	0
20123	ANALOG_INPUT	T_Sensation_OC_12	Thermal sensation of occupant-12 of tenant-2 (Cordelia Woodson)	0
20124	ANALOG_INPUT	Clo_OC_12	Clothing index of occupant-12 of tenant-2 (Cordelia Woodson)	0

20132	BINARY_INPUT	Availability_OC_13	Availability of occupant-13 of tenant-2 (Theodore Place)	0
20133	ANALOG_INPUT	T_Sensation_OC_13	Thermal sensation of occupant-13 of tenant-2 (Theodore Place)	0
20134	ANALOG_INPUT	Clo_OC_13	Clothing index of occupant-13 of tenant-2 (Theodore Place)	0
20142	BINARY_INPUT	Availability_OC_14	Availability of occupant-14 of tenant-2 (Tomoya Katayama)	0
20143	ANALOG_INPUT	T_Sensation_OC_14	Thermal sensation of occupant-14 of tenant-2 (Tomoya Katayama)	0
20144	ANALOG_INPUT	Clo_OC_14	Clothing index of occupant-14 of tenant-2 (Tomoya Katayama)	0
20152	BINARY_INPUT	Availability_OC_15	Availability of occupant-15 of tenant-2 (Michaela Nutter)	0
20153	ANALOG_INPUT	T_Sensation_OC_15	Thermal sensation of occupant-15 of tenant-2 (Michaela Nutter)	0
20154	ANALOG_INPUT	Clo_OC_15	Clothing index of occupant-15 of tenant-2 (Michaela Nutter)	0
20162	BINARY_INPUT	Availability_OC_16	Availability of occupant-16 of tenant-2 (Hajime Ogata)	0
20163	ANALOG_INPUT	T_Sensation_OC_16	Thermal sensation of occupant-16 of tenant-2 (Hajime Ogata)	0
20164	ANALOG_INPUT	Clo_OC_16	Clothing index of occupant-16 of tenant-2 (Hajime Ogata)	0
20172	BINARY_INPUT	Availability_OC_17	Availability of occupant-17 of tenant-2 (Lewis Swaine)	0
20173	ANALOG_INPUT	T_Sensation_OC_17	Thermal sensation of occupant-17 of tenant-2 (Lewis Swaine)	0
20174	ANALOG_INPUT	Clo_OC_17	Clothing index of occupant-17 of tenant-2 (Lewis Swaine)	0
20182	BINARY_INPUT	Availability_OC_18	Availability of occupant-18 of tenant-2 (Valentine Wellington)	0
20183	ANALOG_INPUT	T_Sensation_OC_18	Thermal sensation of occupant-18 of tenant-2 (Valentine Wellington)	0
20184	ANALOG_INPUT	Clo_OC_18	Clothing index of occupant-18 of tenant-2 (Valentine Wellington)	0
20192	BINARY_INPUT	Availability_OC_19	Availability of occupant-19 of tenant-2 (Stephanie Hines)	0
20193	ANALOG_INPUT	T_Sensation_OC_19	Thermal sensation of occupant-19 of tenant-2 (Stephanie Hines)	0
20194	ANALOG_INPUT	Clo_OC_19	Clothing index of occupant-19 of tenant-2 (Stephanie Hines)	0
20202	BINARY_INPUT	Availability_OC_20	Availability of occupant-20 of tenant-2 (Leonard Hill)	0
20203	ANALOG_INPUT	T_Sensation_OC_20	Thermal sensation of occupant-20 of tenant-2 (Leonard Hill)	0
20204	ANALOG_INPUT	Clo_OC_20	Clothing index of occupant-20 of tenant-2 (Leonard Hill)	0
20212	BINARY_INPUT	Availability_OC_21	Availability of occupant-21 of tenant-2 (Hisao Ayame)	0
20213	ANALOG_INPUT	T_Sensation_OC_21	Thermal sensation of occupant-21 of tenant-2 (Hisao Ayame)	0
20214	ANALOG_INPUT	Clo_OC_21	Clothing index of occupant-21 of tenant-2 (Hisao Ayame)	0
20222	BINARY_INPUT	Availability_OC_22	Availability of occupant-22 of tenant-2 (Masanari Ukai)	0
20223	ANALOG_INPUT	T_Sensation_OC_22	Thermal sensation of occupant-22 of tenant-2 (Masanari Ukai)	0
20224	ANALOG_INPUT	Clo_OC_22	Clothing index of occupant-22 of tenant-2 (Masanari Ukai)	0
20232	BINARY_INPUT	Availability_OC_23	Availability of occupant-23 of tenant-2 (Pamela Stackhouse)	0
20233	ANALOG_INPUT	T_Sensation_OC_23	Thermal sensation of occupant-23 of tenant-2 (Pamela Stackhouse)	0
20234	ANALOG_INPUT	Clo_OC_23	Clothing index of occupant-23 of tenant-2 (Pamela Stackhouse)	0
20242	BINARY_INPUT	Availability_OC_24	Availability of occupant-24 of tenant-2 (William Trollope)	0
20243	ANALOG_INPUT	T_Sensation_OC_24	Thermal sensation of occupant-24 of tenant-2 (William Trollope)	0
20244	ANALOG_INPUT	Clo_OC_24	Clothing index of occupant-24 of tenant-2 (William Trollope)	0
20252	BINARY_INPUT	Availability_OC_25	Availability of occupant-25 of tenant-2 (Jasmine Flowers)	0
20253	ANALOG_INPUT	T_Sensation_OC_25	Thermal sensation of occupant-25 of tenant-2 (Jasmine Flowers)	0
20254	ANALOG_INPUT	Clo_OC_25	Clothing index of occupant-25 of tenant-2 (Jasmine Flowers)	0
20262	BINARY_INPUT	Availability_OC_26	Availability of occupant-26 of tenant-2 (Constantin Yaglou)	0
20263	ANALOG_INPUT	T_Sensation_OC_26	Thermal sensation of occupant-26 of tenant-2 (Constantin Yaglou)	0
20264	ANALOG_INPUT	Clo_OC_26	Clothing index of occupant-26 of tenant-2 (Constantin Yaglou)	0
20272	BINARY_INPUT	Availability_OC_27	Availability of occupant-27 of tenant-2 (Edwin Gwatkin)	0
20273	ANALOG_INPUT	T_Sensation_OC_27	Thermal sensation of occupant-27 of tenant-2 (Edwin Gwatkin)	0
20274	ANALOG_INPUT	Clo_OC_27	Clothing index of occupant-27 of tenant-2 (Edwin Gwatkin)	0
20282	BINARY_INPUT	Availability_OC_28	Availability of occupant-28 of tenant-2 (Jeff Northcutt)	0
20283	ANALOG_INPUT	T_Sensation_OC_28	Thermal sensation of occupant-28 of tenant-2 (Jeff Northcutt)	0
20284	ANALOG_INPUT	Clo_OC_28	Clothing index of occupant-28 of tenant-2 (Jeff Northcutt)	0

20292	BINARY_INPUT	Availability_OC_29	Availability of occupant-29 of tenant-2 (Pat Hightower)	0
20293	ANALOG_INPUT	T_Sensation_OC_29	Thermal sensation of occupant-29 of tenant-2 (Pat Hightower)	0
20294	ANALOG_INPUT	Clo_OC_29	Clothing index of occupant-29 of tenant-2 (Pat Hightower)	0
20302	BINARY_INPUT	Availability_OC_30	Availability of occupant-30 of tenant-2 (Brendon Byrd)	0
20303	ANALOG_INPUT	T_Sensation_OC_30	Thermal sensation of occupant-30 of tenant-2 (Brendon Byrd)	0
20304	ANALOG_INPUT	Clo_OC_30	Clothing index of occupant-30 of tenant-2 (Brendon Byrd)	0
20312	BINARY_INPUT	Availability_OC_31	Availability of occupant-31 of tenant-2 (Abel Cleverly)	0
20313	ANALOG_INPUT	T_Sensation_OC_31	Thermal sensation of occupant-31 of tenant-2 (Abel Cleverly)	0
20314	ANALOG_INPUT	Clo_OC_31	Clothing index of occupant-31 of tenant-2 (Abel Cleverly)	0
20322	BINARY_INPUT	Availability_OC_32	Availability of occupant-32 of tenant-2 (Daniel Calladine)	0
20323	ANALOG_INPUT	T_Sensation_OC_32	Thermal sensation of occupant-32 of tenant-2 (Daniel Calladine)	0
20324	ANALOG_INPUT	Clo_OC_32	Clothing index of occupant-32 of tenant-2 (Daniel Calladine)	0
20332	BINARY_INPUT	Availability_OC_33	Availability of occupant-33 of tenant-2 (Makoto Satoh)	0
20333	ANALOG_INPUT	T_Sensation_OC_33	Thermal sensation of occupant-33 of tenant-2 (Makoto Satoh)	0
20334	ANALOG_INPUT	Clo_OC_33	Clothing index of occupant-33 of tenant-2 (Makoto Satoh)	0
20342	BINARY_INPUT	Availability_OC_34	Availability of occupant-34 of tenant-2 (Walter Heston)	0
20343	ANALOG_INPUT	T_Sensation_OC_34	Thermal sensation of occupant-34 of tenant-2 (Walter Heston)	0
20344	ANALOG_INPUT	Clo_OC_34	Clothing index of occupant-34 of tenant-2 (Walter Heston)	0
20352	BINARY_INPUT	Availability_OC_35	Availability of occupant-35 of tenant-2 (Robin Hurst)	0
20353	ANALOG_INPUT	T_Sensation_OC_35	Thermal sensation of occupant-35 of tenant-2 (Robin Hurst)	0
20354	ANALOG_INPUT	Clo_OC_35	Clothing index of occupant-35 of tenant-2 (Robin Hurst)	0
20362	BINARY_INPUT	Availability_OC_36	Availability of occupant-36 of tenant-2 (Rick Dobbs)	0
20363	ANALOG_INPUT	T_Sensation_OC_36	Thermal sensation of occupant-36 of tenant-2 (Rick Dobbs)	0
20364	ANALOG_INPUT	Clo_OC_36	Clothing index of occupant-36 of tenant-2 (Rick Dobbs)	0
20372	BINARY_INPUT	Availability_OC_37	Availability of occupant-37 of tenant-2 (Oswald Coffin)	0
20373	ANALOG_INPUT	T_Sensation_OC_37	Thermal sensation of occupant-37 of tenant-2 (Oswald Coffin)	0
20374	ANALOG_INPUT	Clo_OC_37	Clothing index of occupant-37 of tenant-2 (Oswald Coffin)	0
20382	BINARY_INPUT	Availability_OC_38	Availability of occupant-38 of tenant-2 (Godfrey Doust)	0
20383	ANALOG_INPUT	T_Sensation_OC_38	Thermal sensation of occupant-38 of tenant-2 (Godfrey Doust)	0
20384	ANALOG_INPUT	Clo_OC_38	Clothing index of occupant-38 of tenant-2 (Godfrey Doust)	0
20392	BINARY_INPUT	Availability_OC_39	Availability of occupant-39 of tenant-2 (Hiroyuki Hatada)	0
20393	ANALOG_INPUT	T_Sensation_OC_39	Thermal sensation of occupant-39 of tenant-2 (Hiroyuki Hatada)	0
20394	ANALOG_INPUT	Clo_OC_39	Clothing index of occupant-39 of tenant-2 (Hiroyuki Hatada)	0
20402	BINARY_INPUT	Availability_OC_40	Availability of occupant-40 of tenant-2 (Lindsey Ottley)	0
20403	ANALOG_INPUT	T_Sensation_OC_40	Thermal sensation of occupant-40 of tenant-2 (Lindsey Ottley)	0
20404	ANALOG_INPUT	Clo_OC_40	Clothing index of occupant-40 of tenant-2 (Lindsey Ottley)	0
20412	BINARY_INPUT	Availability_OC_41	Availability of occupant-41 of tenant-2 (Malcolm Watt)	0
20413	ANALOG_INPUT	T_Sensation_OC_41	Thermal sensation of occupant-41 of tenant-2 (Malcolm Watt)	0
20414	ANALOG_INPUT	Clo_OC_41	Clothing index of occupant-41 of tenant-2 (Malcolm Watt)	0
20422	BINARY_INPUT	Availability_OC_42	Availability of occupant-42 of tenant-2 (Elton Vickers)	0
20423	ANALOG_INPUT	T_Sensation_OC_42	Thermal sensation of occupant-42 of tenant-2 (Elton Vickers)	0
20424	ANALOG_INPUT	Clo_OC_42	Clothing index of occupant-42 of tenant-2 (Elton Vickers)	0
20432	BINARY_INPUT	Availability_OC_43	Availability of occupant-43 of tenant-2 (Rodney Benge)	0
20433	ANALOG_INPUT	T_Sensation_OC_43	Thermal sensation of occupant-43 of tenant-2 (Rodney Benge)	0
20434	ANALOG_INPUT	Clo_OC_43	Clothing index of occupant-43 of tenant-2 (Rodney Benge)	0
20442	BINARY_INPUT	Availability_OC_44	Availability of occupant-44 of tenant-2 (Stanley Neilson)	0
20443	ANALOG_INPUT	T_Sensation_OC_44	Thermal sensation of occupant-44 of tenant-2 (Stanley Neilson)	0
20444	ANALOG_INPUT	Clo_OC_44	Clothing index of occupant-44 of tenant-2 (Stanley Neilson)	0

20452	BINARY_INPUT	Availability_OC_45	Availability of occupant-45 of tenant-2 (Willis Carrier)	0
20453	ANALOG_INPUT	T_Sensation_OC_45	Thermal sensation of occupant-45 of tenant-2 (Willis Carrier)	0
20454	ANALOG_INPUT	Clo_OC_45	Clothing index of occupant-45 of tenant-2 (Willis Carrier)	0
20462	BINARY_INPUT	Availability_OC_46	Availability of occupant-46 of tenant-2 (Emma Botting)	0
20463	ANALOG_INPUT	T_Sensation_OC_46	Thermal sensation of occupant-46 of tenant-2 (Emma Botting)	0
20464	ANALOG_INPUT	Clo_OC_46	Clothing index of occupant-46 of tenant-2 (Emma Botting)	0
20472	BINARY_INPUT	Availability_OC_47	Availability of occupant-47 of tenant-2 (Wanda Madgwick)	0
20473	ANALOG_INPUT	T_Sensation_OC_47	Thermal sensation of occupant-47 of tenant-2 (Wanda Madgwick)	0
20474	ANALOG_INPUT	Clo_OC_47	Clothing index of occupant-47 of tenant-2 (Wanda Madgwick)	0
20482	BINARY_INPUT	Availability_OC_48	Availability of occupant-48 of tenant-2 (Quincy Windsor-Clive)	0
20483	ANALOG_INPUT	T_Sensation_OC_48	Thermal sensation of occupant-48 of tenant-2 (Quincy Windsor-Clive)	0
20484	ANALOG_INPUT	Clo_OC_48	Clothing index of occupant-48 of tenant-2 (Quincy Windsor-Clive)	0

5) Objects in the “VentilationController” device

On/Off 状態のインスタンス番号 ＝1000×室外機番号 ＋100×室内機番号 ＋3

バイパス制御の有効無効のインスタンス番号 ＝1000×室外機番号 ＋100×室内機番号 ＋4

ファン風量のインスタンス番号 ＝1000×室外機番号 ＋100×室内機番号 ＋5

Inst. No.	Type	Name	Description	Initial value
1	ANALOG_INPUT	CO2 level of south tenant	CO2 level of south tenant.	400
2	ANALOG_INPUT	CO2 level of north tenant	CO2 level of north tenant.	400
1103	BINARY_OUTPUT	On/Off setting/state (HEX1-1)	This object is used to control or monitor On/Off state of HEX1-1	0
1104	BINARY_OUTPUT	Bypass control setting/state (HEX1-1)	This object is used to control or monitor bypass control state of HEX1-1	0
1105	MULTI_STATE_OUTPUT	Fan speed (HEX1-1)	This object is used to control or monitor fan speed of HEX1-1. 1:Low; 2:Middle; 3:High	3
1203	BINARY_OUTPUT	On/Off setting/state (HEX1-2)	This object is used to control or monitor On/Off state of HEX1-2	0
1204	BINARY_OUTPUT	Bypass control setting/state (HEX1-2)	This object is used to control or monitor bypass control state of HEX1-2	0
1205	MULTI_STATE_OUTPUT	Fan speed (HEX1-2)	This object is used to control or monitor fan speed of HEX1-2. 1:Low; 2:Middle; 3:High	3
1303	BINARY_OUTPUT	On/Off setting/state (HEX1-3)	This object is used to control or monitor On/Off state of HEX1-3	0
1304	BINARY_OUTPUT	Bypass control setting/state (HEX1-3)	This object is used to control or monitor bypass control state of HEX1-3	0
1305	MULTI_STATE_OUTPUT	Fan speed (HEX1-3)	This object is used to control or monitor fan speed of HEX1-3. 1:Low; 2:Middle; 3:High	3
1403	BINARY_OUTPUT	On/Off setting/state (HEX1-4)	This object is used to control or monitor On/Off state of HEX1-4	0
1404	BINARY_OUTPUT	Bypass control setting/state (HEX1-4)	This object is used to control or monitor bypass control state of HEX1-4	0
1405	MULTI_STATE_OUTPUT	Fan speed (HEX1-4)	This object is used to control or monitor fan speed of HEX1-4. 1:Low; 2:Middle; 3:High	3
1503	BINARY_OUTPUT	On/Off setting/state (HEX1-5)	This object is used to control or monitor On/Off state of HEX1-5	0
1504	BINARY_OUTPUT	Bypass control setting/state (HEX1-5)	This object is used to control or monitor bypass control state of HEX1-5	0
1505	MULTI_STATE_OUTPUT	Fan speed (HEX1-5)	This object is used to control or monitor fan speed of HEX1-5. 1:Low; 2:Middle; 3:High	3
2103	BINARY_OUTPUT	On/Off setting/state (HEX2-1)	This object is used to control or monitor On/Off state of HEX2-1	0
2104	BINARY_OUTPUT	Bypass control setting/state (HEX2-1)	This object is used to control or monitor bypass control state of HEX2-1	0
2105	MULTI_STATE_OUTPUT	Fan speed (HEX2-1)	This object is used to control or monitor fan speed of HEX2-1. 1:Low; 2:Middle; 3:High	3
2203	BINARY_OUTPUT	On/Off setting/state (HEX2-2)	This object is used to control or monitor On/Off state of HEX2-2	0
2204	BINARY_OUTPUT	Bypass control setting/state (HEX2-2)	This object is used to control or monitor bypass control state of HEX2-2	0
2205	MULTI_STATE_OUTPUT	Fan speed (HEX2-2)	This object is used to control or monitor fan speed of HEX2-2. 1:Low; 2:Middle; 3:High	3
2303	BINARY_OUTPUT	On/Off setting/state (HEX2-3)	This object is used to control or monitor On/Off state of HEX2-3	0

2304	BINARY_OUTPUT	Bypass control setting/state (HEX2-3)	This object is used to control or monitor bypass control state of HEX2-3	0
2305	MULTI_STATE_OUTPUT	Fan speed (HEX2-3)	This object is used to control or monitor fan speed of HEX2-3. 1:Low; 2:Middle; 3:High	3
2403	BINARY_OUTPUT	On/Off setting/state (HEX2-4)	This object is used to control or monitor On/Off state of HEX2-4	0
2404	BINARY_OUTPUT	Bypass control setting/state (HEX2-4)	This object is used to control or monitor bypass control state of HEX2-4	0
2405	MULTI_STATE_OUTPUT	Fan speed (HEX2-4)	This object is used to control or monitor fan speed of HEX2-4. 1:Low; 2:Middle; 3:High	3
3103	BINARY_OUTPUT	On/Off setting/state (HEX3-1)	This object is used to control or monitor On/Off state of HEX3-1	0
3104	BINARY_OUTPUT	Bypass control setting/state (HEX3-1)	This object is used to control or monitor bypass control state of HEX3-1	0
3105	MULTI_STATE_OUTPUT	Fan speed (HEX3-1)	This object is used to control or monitor fan speed of HEX3-1. 1:Low; 2:Middle; 3:High	3
3203	BINARY_OUTPUT	On/Off setting/state (HEX3-2)	This object is used to control or monitor On/Off state of HEX3-2	0
3204	BINARY_OUTPUT	Bypass control setting/state (HEX3-2)	This object is used to control or monitor bypass control state of HEX3-2	0
3205	MULTI_STATE_OUTPUT	Fan speed (HEX3-2)	This object is used to control or monitor fan speed of HEX3-2. 1:Low; 2:Middle; 3:High	3
3303	BINARY_OUTPUT	On/Off setting/state (HEX3-3)	This object is used to control or monitor On/Off state of HEX3-3	0
3304	BINARY_OUTPUT	Bypass control setting/state (HEX3-3)	This object is used to control or monitor bypass control state of HEX3-3	0
3305	MULTI_STATE_OUTPUT	Fan speed (HEX3-3)	This object is used to control or monitor fan speed of HEX3-3. 1:Low; 2:Middle; 3:High	3
3403	BINARY_OUTPUT	On/Off setting/state (HEX3-4)	This object is used to control or monitor On/Off state of HEX3-4	0
3404	BINARY_OUTPUT	Bypass control setting/state (HEX3-4)	This object is used to control or monitor bypass control state of HEX3-4	0
3405	MULTI_STATE_OUTPUT	Fan speed (HEX3-4)	This object is used to control or monitor fan speed of HEX3-4. 1:Low; 2:Middle; 3:High	3
3503	BINARY_OUTPUT	On/Off setting/state (HEX3-5)	This object is used to control or monitor On/Off state of HEX3-5	0
3504	BINARY_OUTPUT	Bypass control setting/state (HEX3-5)	This object is used to control or monitor bypass control state of HEX3-5	0
3505	MULTI_STATE_OUTPUT	Fan speed (HEX3-5)	This object is used to control or monitor fan speed of HEX3-5. 1:Low; 2:Middle; 3:High	3
4103	BINARY_OUTPUT	On/Off setting/state (HEX4-1)	This object is used to control or monitor On/Off state of HEX4-1	0
4104	BINARY_OUTPUT	Bypass control setting/state (HEX4-1)	This object is used to control or monitor bypass control state of HEX4-1	0
4105	MULTI_STATE_OUTPUT	Fan speed (HEX4-1)	This object is used to control or monitor fan speed of HEX4-1. 1:Low; 2:Middle; 3:High	3
4203	BINARY_OUTPUT	On/Off setting/state (HEX4-2)	This object is used to control or monitor On/Off state of HEX4-2	0
4204	BINARY_OUTPUT	Bypass control setting/state (HEX4-2)	This object is used to control or monitor bypass control state of HEX4-2	0
4205	MULTI_STATE_OUTPUT	Fan speed (HEX4-2)	This object is used to control or monitor fan speed of HEX4-2. 1:Low; 2:Middle; 3:High	3
4303	BINARY_OUTPUT	On/Off setting/state (HEX4-3)	This object is used to control or monitor On/Off state of HEX4-3	0
4304	BINARY_OUTPUT	Bypass control setting/state (HEX4-3)	This object is used to control or monitor bypass control state of HEX4-3	0
4305	MULTI_STATE_OUTPUT	Fan speed (HEX4-3)	This object is used to control or monitor fan speed of HEX4-3. 1:Low; 2:Middle; 3:High	3
4403	BINARY_OUTPUT	On/Off setting/state (HEX4-4)	This object is used to control or monitor On/Off state of HEX4-4	0
4404	BINARY_OUTPUT	Bypass control setting/state (HEX4-4)	This object is used to control or monitor bypass control state of HEX4-4	0
4405	MULTI_STATE_OUTPUT	Fan speed (HEX4-4)	This object is used to control or monitor fan speed of HEX4-4. 1:Low; 2:Middle; 3:High	3

6) Objects in the “DummyDevice”

Inst. No.	Type	Name	Description	Initial value
1	ANALOG_VALUE	Analog value (int)	Dummy object to test communication of analog value (int).	1
2	ANALOG_OUTPUT	Analog output (int)	Dummy object to test communication of analog output (int).	2
3	ANALOG_INPUT	Analog input (int)	Dummy object to test communication of analog input (int).	3
4	ANALOG_VALUE	Analog value (float)	Dummy object to test communication of analog value (real).	4
5	ANALOG_OUTPUT	Analog output (float)	Dummy object to test communication of analog output (real).	5
6	ANALOG_INPUT	Analog input (float)	Dummy object to test communication of analog input (real).	6
7	BINARY_VALUE	Binary value	Dummy object to test communication of binary value.	0
8	BINARY_OUTPUT	Binary output	Dummy object to test communication of binary output.	0
9	BINARY_INPUT	Binary input	Dummy object to test communication of binary input.	0
10	MULTI_STATE_VALUE	Multistate value	Dummy object to test communication of multistate value.	1
11	MULTI_STATE_OUTPUT	Multistate output	Dummy object to test communication of multistate output.	2
12	MULTI_STATE_INPUT	Multistate input	Dummy object to test communication of multistate input.	3
13	DATETIME_VALUE	BACnet date time	Dummy object to test communication of bacnet date time.	1980/6/14 0:00

Appendix 2

Occupants

No	Tenant	Zone	First name	Last name	Age	Height	Weight	M/F
1	South	S1	Dana	Hattersley	45	160.9	69.4	F
2	South	S1	Humphrey	Lock	45	180.3	55.8	M
3	South	S1	Cassie	Harris	65	156.2	53.3	F
4	South	S2	Cecil	Topping	35	168.3	65.0	M
5	South	S2	Laila	Black	65	155.8	51.2	F
6	South	S2	Clive	Toolson	65	173.6	59.1	M
7	South	S3	Monique	Cartwright	25	159.3	50.2	F
8	South	S3	Josiah	Conder	55	170.0	72.0	M
9	South	S3	Phil	Barker	65	163.4	63.8	M
10	South	S4	Meredith	Baldridge	25	169.9	79.0	M
11	South	S4	Angelica	Roundell	35	164.0	51.3	F
12	South	S4	Hermann	Rietschel	35	172.3	66.2	M
13	South	S4	Allyn	Galbraith	45	172.0	66.2	M
14	South	S5	Wallace	Sabine	35	174.5	58.0	M
15	South	S5	David	Midwinter	45	165.2	77.2	M
16	South	S5	Rowland	Rouse	35	175.4	71.9	M
17	South	S5	Yuichiro	Iio	45	168.6	81.0	M
18	South	S6	Zachariah	Vernon	25	181.7	69.6	M
19	South	S6	Allyn	Lympany	65	149.0	58.2	F
20	South	S7	Daiki	Kobayashi	25	175.2	56.6	M
21	South	S7	Yvonne	Murrills	65	153.6	62.2	F
22	South	S7	Vince	Cok	55	162.8	66.6	M
23	South	S7	Niccolo	Giannetti	25	165.9	58.0	M
24	South	S7	Elizabeth	Roundell	65	153.0	62.4	F
25	South	S7	Nicola	Turnbull	45	160.3	61.6	F
26	South	S8	Masahi	Momota	55	156.5	67.3	M
27	South	S8	Jade	Mollison	65	153.7	64.0	F
28	South	S8	Linus	Hanley	45	160.9	77.5	M
29	South	S8	Valentine	Elliston	45	172.7	70.5	M
30	South	S8	Roman	Steele	45	173.4	68.8	M
31	South	S8	Savannah	Biggs	55	149.4	55.5	F
32	South	S8	Howard	Astley	25	173.5	72.0	M
33	South	S8	Masato	Miyata	35	179.6	64.4	M
34	South	S9	Aileen	Winder	45	153.8	60.6	F
35	South	S9	Landon	Ackroyd	25	173.2	49.9	M
36	South	S9	Leo	Quantrill	65	175.9	57.7	M
37	South	S9	Eisuke	Togashi	55	171.4	78.7	M
38	South	S9	Wilhelmina	Chalmers	35	160.5	57.0	F
39	North	N1	Kim	Collingwood	35	162.9	68.1	M
40	North	N1	Takahiro	Ueno	45	170.6	75.5	M
41	North	N1	Kimberly	Holder	25	157.5	44.1	F
42	North	N1	Sophie	Coffin	45	157.5	56.9	F
43	North	N1	Rolla	Carpenter	55	162.8	74.0	M
44	North	N2	Pauline	Gooding	35	164.5	48.0	F
45	North	N2	Sei	Nagashima	35	178.0	64.9	M
46	North	N2	Louisa	Street	45	156.7	40.3	F
47	North	N2	Lindsay	Buckler	25	157.1	46.9	F
48	North	N2	Katsuyuki	Edahiro	55	180.0	69.4	M
49	North	N3	Carey	Blanchfield	55	175.5	68.9	M
50	North	N3	Cordelia	Woodson	25	169.1	54.0	F
51	North	N3	Theodore	Place	35	172.7	70.7	M
52	North	N4	Tomoya	Katayama	45	171.3	67.1	M
53	North	N4	Michaela	Nutter	45	167.7	54.3	F
54	North	N4	Hajime	Ogata	65	158.1	69.7	M
55	North	N4	Lewis	Swaine	35	172.9	61.8	M
56	North	N4	Valentine	Wellington	45	170.7	73.6	M
57	North	N4	Stephanie	Hines	35	162.8	55.4	F
58	North	N4	Leonard	Hill	35	176.4	54.8	M
59	North	N5	Hisao	Ayame	35	180.8	67.0	M
60	North	N5	Masanari	Ukai	45	171.1	74.1	M
61	North	N5	Pamela	Stackhouse	45	164.6	53.4	F
62	North	N5	William	Trollope	35	179.1	35.1	M
63	North	N5	Jasmine	Flowers	65	160.4	44.1	F
64	North	N5	Constantin	Yaglou	35	164.2	73.1	M
65	North	N5	Edwin	Gwatkin	45	168.6	50.7	M
66	North	N6	Jeff	Northcutt	55	169.3	69.6	M
67	North	N6	Pat	Hightower	35	178.4	44.3	M
68	North	N6	Brendon	Byrd	25	170.5	71.0	M
69	North	N6	Abel	Cleverly	55	175.9	69.1	M
70	North	N6	Daniel	Calladine	35	167.9	66.2	M
71	North	N7	Makoto	Satoh	25	163.4	72.3	M
72	North	N7	Walter	Heston	35	176.1	80.4	M
73	North	N7	Robin	Hurst	25	177.7	60.6	M
74	North	N7	Rick	Dobbs	55	163.6	64.8	M
75	North	N8	Oswald	Coffin	45	168.7	59.0	M
76	North	N8	Godfrey	Doust	45	157.8	78.7	M
77	North	N8	Hiroyuki	Hatada	45	177.4	67.6	M
78	North	N8	Lindsey	Ottley	35	152.2	48.8	F
79	North	N8	Malcolm	Watt	35	167.6	74.0	M
70	North	N8	Elton	Vickers	45	179.8	64.3	M
81	North	N8	Rodney	Benge	35	169.4	69.9	M
82	North	N9	Stanley	Neilson	45	166.0	55.4	M
83	North	N9	Willis	Carrier	45	162.5	78.3	M
84	North	N9	Emma	Botting	45	165.5	51.0	F
85	North	N9	Wanda	Madgwick	35	150.4	48.6	F
86	North	N9	Quincy	Windsor-Clive	35	171.2	72.2	M

† Height, weight, and gender are just set for the fun of giving reality and do not affect the calculation results.