

**BA3-CL1 プログラミング ツール
(形式:BA3CL1KW)**

**BA3-CL3 用 Firmware Library
取扱説明書**

NM-7283-D 改 2

はじめに

本マニュアルは、Building Automation Controller(以下、「コントローラ」)に搭載のソフトロジック用専用ファンクション及びファンクションブロックについて説明します。

ご使用になる前に本書をよくお読み頂き、正しくお使い下さい。

なお、本マニュアルは、IEC61131-3 仕様を理解している方を前提に作成しています。用語については、それぞれのマニュアルを参照して下さい。

マニュアルについて

本マニュアルに記載されている記号、および共通注意事項は以下のとおりです。

■記号説明

注 意 : 操作時の注意事項を記載しています。

MEMO : 操作時のポイント事項を記載しています。

注 意

- ・ 本書の内容に関しては、改良のために予告なしに仕様等変更することがありますのでご了承ください。

目 次

1. ライブライ詳細.....	5
1.1. ライブライ一覧	5
1.2. MsysFwLibBA3CL3 詳細.....	6
1.3. ファンクションブロック エラーコード一覧.....	8
ANA_SCHEDULER [FB]	9
CALORIE [FB].....	11
CMP / CMP_F [FB].....	12
CYT [FB].....	13
DLY [FB].....	14
ENTHALPY [FB].....	15
FILTER [FB].....	16
LMT [FB].....	17
LOAD_RESET [FB]	18
LOOP_SINGLE [FB].....	20
MOMENTARY_OUTPUT [FB].....	23
PULSE_COUNTER [FB]	24
WEIGHTED_AVERAGE [FB]	25
IS_REAL_NAN [FN].....	26
SET_REAL_NAN [FN].....	27
POINT_HISTORY [FB].....	28
RTC_NOW [FB].....	29
ANA_INPUT [FB]	30
ANA_INPUT_UI32 [FB].....	31
ANA_OUTPUT [FB]	32
ANA_OUTPUT_UI32 [FB].....	33
DIG_INPUT [FB].....	34
DIG_OUTPUT [FB].....	35
LON_NV_INFO [FB].....	36
LON_NVI_general [FB]	37
LON_SCPT_maxRcvTime [FB].....	39
LON_SNVI_address [FB]	40
LON_SNVI_count [FB].....	41
LON_SNVI_count_inc [FB]	42
LON_SNVI_count_inc_f [FB]	43
LON_SNVI_flow [FB]	44
LON_SNVI_hvac_emerg [FB]	45
LON_SNVI_hvac_mode [FB]	46
LON_SNVI_hvac_status [FB]	47
LON_SNVI_lev_cont [FB]	48
LON_SNVI_lev_cont_f [FB]	49
LON_SNVI_lev_percent [FB]	50
LON_SNVI_occupancy [FB]	51
LON_SNVI_ppm [FB]	52

LON_SNVI_press_p [FB]	53
LON_SNVI_state [FB]	54
LON_SNVI_str_asc [FB]	55
LON_SNVI_switch [FB]	56
LON_SNVI_temp_p [FB]	57
LON_SNVI_time_sec [FB]	58
LON_SNVI_tod_event [FB]	59
LON_SNVI_volt_f [FB]	60
LON_NVO_general [FB]	61
LON_SNVO_address [FB]	63
LON_SNVO_count [FB]	64
LON_SNVO_count_inc [FB]	65
LON_SNVO_count_inc_f [FB]	66
LON_SNVO_flow [FB]	67
LON_SNVO_hvac_emerg [FB]	68
LON_SNVO_hvac_mode [FB]	69
LON_SNVO_hvac_status [FB]	70
LON_SNVO_lev_cont [FB]	71
LON_SNVO_lev_cont_f [FB]	72
LON_SNVO_lev_percent [FB]	73
LON_SNVO_occupancy [FB]	74
LON_SNVO_ppm [FB]	75
LON_SNVO_press_p [FB]	76
LON_SNVO_state [FB]	77
LON_SNVO_str_asc [FB]	78
LON_SNVO_switch [FB]	79
LON_SNVO_temp_p [FB]	80
LON_SNVO_time_sec [FB]	81
LON_SNVO_tod_event [FB]	82
LON_SNVO_volt_f [FB]	83
1.4. 型別ファンクションブロックを用意している SNVT	84
1.5. 型別ファンクションブロックが用意されていない Network Variable Type の扱い	86

1. ライブラリ詳細

1.1. ライブラリ一覧

ライブラリ名	機能
MsysFwLibBA3CL3	BA3-CL3 Functions

1.2. MsysFwLibBA3CL3 詳細

TYPE は FB (Function Block), FN (Function) を表しています。

POU	TYPE	機能	サポート Library *1
演算			
ANA_SCHEDULER	FB	折れ線リニアライザ	
CALORIE	FB	熱量演算	
CMP	FB	ヒステリシス付き比較	
CMP_F	FB	ヒステリシス付き比較	
CYT	FB	サイクルタイマ	
DLY	FB	デュアルディレイタイマ	
ENTHALPY	FB	エンタルピ演算	
FILTER	FB	一次遅れフィルタ	
LMT	FB	変化量制限	
LOAD_RESET	FB	給気温度最適化制御	
LOOP_SINGLE	FB	PID 演算	
MOMENTARY_OUTPUT	FB	モメンタリ出力	
PULSE_COUNTER	FB	パルスカウント	
WEIGHTED_AVERAGE	FB	加重平均	
浮動小数点演算関連			
IS_REAL_NAN	FN	NaN 値判定	
SET_REAL_NAN	FN	NaN 値設定	
システム関連			
POINT_HISTORY	FB	変数値の履歴書き込み	
RTC_NOW	FB	現在日付時刻の取得	
入出力関連			
ANA_INPUT	FB	H/W 入力変換 (Analog)	
ANA_INPUT_UI32	FB	H/W 32bit 入力 (Analog)	
ANA_OUTPUT	FB	H/W 出力変換 (Analog)	
ANA_OUTPUT_UI32	FB	H/W 32bit 出力 (Analog)	
DIG_INPUT	FB	H/W 入力変換 (Digital)	
DIG_OUTPUT	FB	H/W 出力変換 (Digital)	
LonWorks 関連			
LON_NV_INFO	FB	LON nv 情報の取得	
LonWorks 関連(値取得)			
LON_NVI_general	FB	汎用データアクセス	
LON_SCPT_maxRcvTime	FB	SCPT_maxRcvTime(48)	
LON_SNVI_address	FB	SNVT_address(114)	
LON_SNVI_count	FB	SNVT_count(8)	
LON_SNVI_count_inc	FB	SNVT_count_inc(9)	
LON_SNVI_count_inc_f	FB	SNVT_count_inc_f(52)	
LON_SNVI_flow	FB	SNVT_flow(15)	
LON_SNVI_hvac_emerg	FB	SNVT_hvac_emerg(103)	
LON_SNVI_hvac_mode	FB	SNVT_hvac_mode(108)	
LON_SNVI_hvac_status	FB	SNVT_hvac_status(112)	
LON_SNVI_lev_cont	FB	SNVT_lev_cont(21)	
LON_SNVI_lev_cont_f	FB	SNVT_lev_cont_f(55)	
LON_SNVI_lev_percent	FB	SNVT_lev_percent(81)	
LON_SNVI_occupancy	FB	SNVT_occupancy(109)	
LON_SNVI_ppm	FB	SNVT_ppm(29)	
LON_SNVI_press_p	FB	SNVT_press_p(113)	
LON_SNVI_state	FB	SNVT_state(83)	

1. ライブラリ詳細

LON_SNVI_str_asc	FB	SNVT_str_asc(36)	
LON_SNVI_switch	FB	SNVT_switch(95)	
LON_SNVI_temp_p	FB	SNVT_temp_p(105)	
LON_SNVI_time_sec	FB	SNVT_time_sec(107)	
LON_SNVI_tod_event	FB	SNVT_tod_event(128)	
LON_SNVI_volt_f	FB	SNVT_volt_f(66)	
LonWorks 関連(値設定)			
LON_NVO_general	FB	汎用データアクセス	
LON_SNVO_address	FB	SNVT_address(114)	
LON_SNVO_count	FB	SNVT_count(8)	
LON_SNVO_count_inc	FB	SNVT_count_inc(9)	
LON_SNVO_count_inc_f	FB	SNVT_count_inc_f(52)	
LON_SNVO_flow	FB	SNVT_flow(15)	
LON_SNVO_hvac_emerg	FB	SNVT_hvac_emerg(103)	
LON_SNVO_hvac_mode	FB	SNVT_hvac_mode(108)	
LON_SNVO_hvac_status	FB	SNVT_hvac_status(112)	
LON_SNVO_lev_cont	FB	SNVT_lev_cont(21)	
LON_SNVO_lev_cont_f	FB	SNVT_lev_cont_f(55)	
LON_SNVO_lev_percent	FB	SNVT_lev_percent(81)	
LON_SNVO_occupancy	FB	SNVT_occupancy(109)	
LON_SNVO_ppm	FB	SNVT_ppm(29)	
LON_SNVO_press_p	FB	SNVT_press_p(113)	
LON_SNVO_state	FB	SNVT_state(83)	
LON_SNVO_str_asc	FB	SNVT_str_asc(36)	
LON_SNVO_switch	FB	SNVT_switch(95)	
LON_SNVO_temp_p	FB	SNVT_temp_p(105)	
LON_SNVO_time_sec	FB	SNVT_time_sec(107)	
LON_SNVO_tod_event	FB	SNVT_tod_event(128)	
LON_SNVO_volt_f	FB	SNVT_volt_f(66)	

*1) サポート Library 欄は、そのファンクションあるいはファンクションブロックのサポートを開始したライブラリのバージョンを記述しています。この欄が空の場合は v1.0.0 からサポートしていることを示します。

補 足

- LonWorks 関連ファンクションブロックについて
LON_SCPT_～ファンクションブロックは、CP の現在値の取得に使用します。
LON_SNVI_～ファンクションブロックは、nvi, nci 変数の現在値の取得に使用します。
LON_SNVO_～ファンクションブロックは、nvo 変数への値設定に使用します。

1.3. ファンクションブロック エラーコード一覧

番号	記号	内容
0	NO_ERROR	正常(エラーなし)
11	ERR_MATH_DivByZero	演算で0割が発生
		[ANA_***, DIG_*** の FB]
101	ERR_PARA_CARDSLOT	指定の R3 I/O カードスロット番号は範囲外
102	ERR_PARA_CARDADDR	指定の R3 I/O アドレスは範囲外
103	ERR_PARA_CARDTypeMismatch	指定の R3 I/O カードの入出力タイプと FB が不一致
104	ERR_PARA_CARDPOINT	指定の R3 I/O アドレスは実ポイン数の範囲外
105	ERR_PARA_CARDStateIsNotValid	R3 I/O の該当カードにカード入出力情報が未確定
106	ERR_PARA_CARDStateHasError	R3 I/O の該当カードにデータエラーあるいはハードエラーが発生
107	ERR_PARA_CARDChHwError	R3 I/O の該当チャネルにハードウェアエラーは発生
108	ERR_PARA_CARDChInpError	R3 I/O の該当チャネルに入力データエラーが発生
109	ERR_PARA_CARDChInpNotEnabled	R3 I/O の該当チャネルの入力データステータスが無効 [LonWorks 関連 FB]
110	ERR_PARA_NV_Type	変数 TYPE が範囲外
111	ERR_PARA_NV_Index	指定のインデックスは範囲外
112	ERR_PARA_NV_ReqDataLen	指定のデータ長は範囲外
113	ERR_PARA_NV_TypeIndexMismatch	FB が規定している Network variable type と指定されている変数の現在の Type が不一致
114	ERR_PARA_NV_viPos_Range	viPos + viDataType のデータ長 <= 31 を超えている [General]
190	ERR_PARA_ARG	入力パラメータが範囲外
191	ERR_PARA_IsNaN	必須入力パラメータが設定されていないか、或いは NaN 値が設定されている [General (演算中)]
201	ERR_PARA_Raw_Range	上下限設定が範囲外
202	ERR_PARA_Scale_Range	スケール設定が範囲外
203	ERR_PARA_vi_Range	vi 入力値が範囲外
204	ERR_PARA_MinMax_Range	上下限が範囲外あるいは上下値が逆転している [POINT_HISTORY]
301	ERR_PARA_viUNIT	viUint の設定が範囲外 [LOOP_SINGLE]
302	ERR_PARA_PID_vipB_IsZero	vipB が0
303	ERR_PARA_PID_vo_Empty	演算エラーのため vo に結果は格納されていない
304	ERR_PARA_PID_vo_Range	演算結果は規定の範囲を超えている

ANA_SCHEDULER [FB]**折れ線リニアライザ**

入力 vi を折れ線補正し出力します。

ANA_SCHEDULER

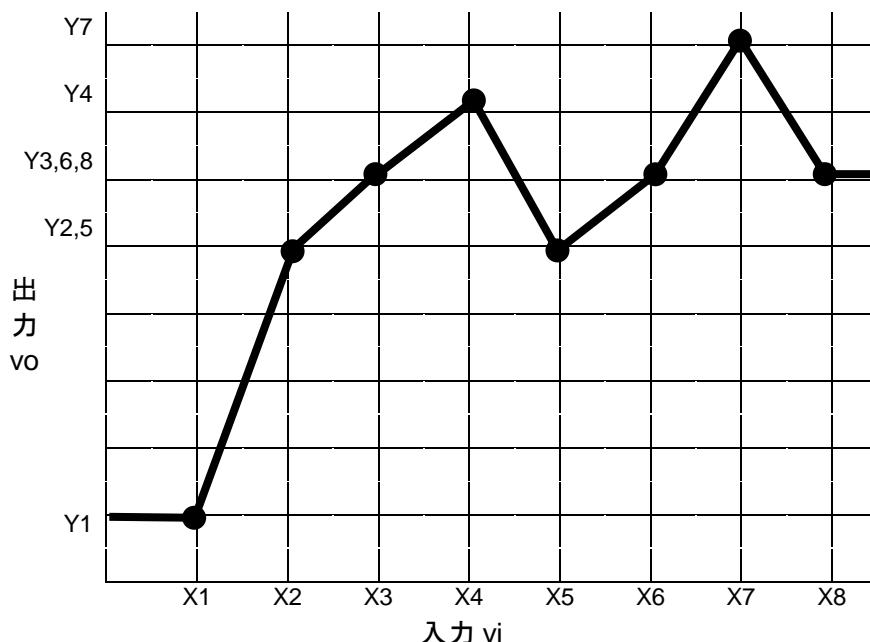
viEN	voENO
vi	voERRNO
viX1 .. 8	vo
viY1 .. 8	:

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		(FALSE) vo : FALSE (TRUE) vo : 演算結果 Default (TRUE)
vi	入力	REAL	NaN, Floating point	Default (NaN)
viX1 .. 8	X 座標	REAL	NaN, Floating point	Default (NaN)
viY1 .. 8	Y 座標	REAL	NaN, Floating point	Default (NaN)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	出力	REAL	NaN, Floating point	

(解説)

- viXn <= vi < viXn+1 の場合

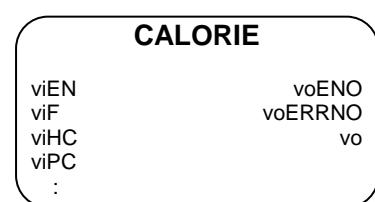
$$vo = \frac{viY_{n+1} - viY_n}{viX_{n+1} - viX_n} (vi - viX_n) + viY_n$$

(注意)

- 1) Y 座標は昇順である必要はないが X 座標は昇順であること。
X 座標の昇順が崩れた場合はその直前までが有効とします。
例えば、 $X1 \leq X2 \leq X3 > X4$ の場合は $X3$ までが有効となります。
- 2) 入力 vi が設定座標の範囲外の場合は両端の値に固定されます。
 $vi < X1$ の場合は $vo = Y1$
 $vi > Xn$ の場合は $vo = Yn$ 但し n は最終有効値
- 3) 同じ X 座標に複数の定義がある場合は最終の値を有効とします。
 $(X1, Y1)=(m,n1), (X2, Y2)=(m, n2)$ で $vi=m$ であれば $vo=n2$ となります。
- 4) X 座標あるいは Y 座標値が NaN の場合はその定義は無視されます。
- 5) 有効な座標定義がない場合は $vo=NaN$ となります。

CALORIE [FB]**熱量演算**

往水温度・還水温度・流量または給気温度・還気温度・風量
より熱量を演算し vo に出力します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		(FALSE) vo : FALSE (TRUE) vo : 演算結果 Default (TRUE)
viF	流量(風量)入力	REAL	NaN, Floating point	Default (NaN)
viHC	冷暖切換入力	BOOL		FALSE: 暖房 TRUE: 冷房 Default (FALSE)
viPC	冷房時カロリー演算定数	REAL	NaN, Floating point	Default (NaN)
viPH	暖房時カロリー演算定数	REAL	NaN, Floating point	Default (NaN)
viTR	還水(還気)温度入力	REAL	NaN, Floating point	Default (NaN)
viTS	往水(給気)温度入力	REAL	NaN, Floating point	Default (NaN)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	結果(熱量)	REAL	NaN, Floating point	

(解説)

冷暖切換入力 viHC の値により下記の演算を行います。

viHC=FALSE (暖房)

$$\text{結果} = (\text{viTS} - \text{viTR}) * \text{viF} * \text{viPH}$$

viHC=TRUE (冷房)

$$\text{結果} = (\text{viTR} - \text{viTS}) * \text{viF} * \text{viPC}$$

(注意)

1) 演算に必要な入力に NaN 値が含まれる場合は結果を NaN 値で返します。

CMP / CMP_F [FB]**比較**

入力を比較した結果を返します。

CMP / CMP_F

viEN	voENO
vi1	voERRNO
vi2	vo
viHys	

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: vo=FALSE TRUE: vo=演算結果 Default (TRUE)
vi1	入力値(vi1)	REAL	NaN, Floating Point	Default (0.0)
vi2	比較設定値(vi2)	REAL	〃	Default (0.0)
viHys	ヒステリシス	REAL	NaN, Floating Point ≥ 0	Default (0.0) 自動下限制限

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	比較結果	BOOL		

(解説)

各関数は下記の演算結果を返します。

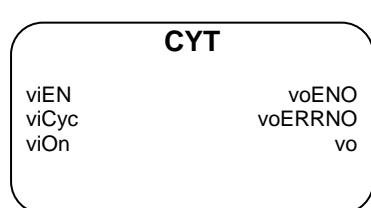
CMP 関数 $vi1 \geq vi2$ の時点から $vi1 < (vi2 - viHys)$ までの間 vo=TRUE

CMP_F 関数 $vi1 \leq vi2$ の時点から $vi1 > (vi2 + viHys)$ までの間 vo=TRUE



CYT [FB]**サイクリックタイマ**

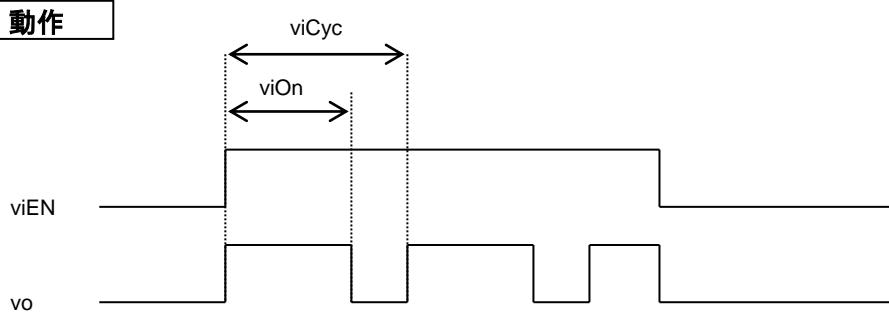
EN の周期毎に viOn 時間だけ出力 vo を ON (TRUE) にします。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		(FALSE) vo : FALSE (TRUE) vo : 演算結果 Default (TRUE)
viCyc	サイクル時間	INT	0 ~ 9999	[sec] Default (0) 自動上下限制限
viOn	ON 時間	INT	0 ~ 9999	[sec] Default (0) 自動上下限制限

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	結果	BOOL		

(解説)**動作**

DLY [FB]**デュアルディレイタイマ****(解説)**

入力 vi に対して設定した ON ディレイまたは OFF ディレイした結果を vo に出力します。

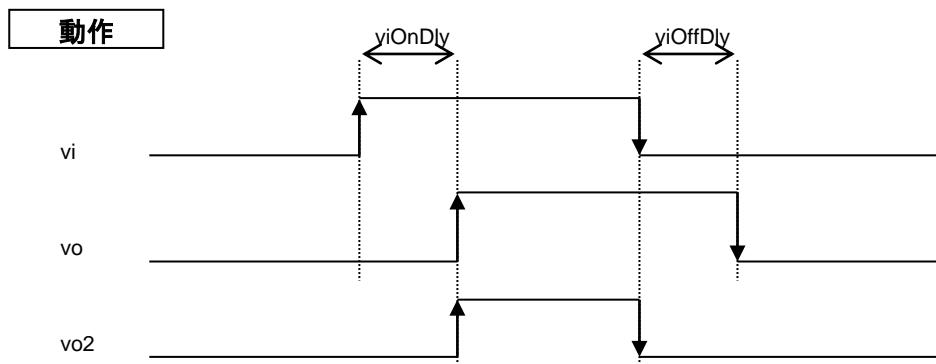
DLY	
viEN	voENO
viAct	voERRNO
vi	vo
viOnDly	vo2
viOffDly	

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		(FALSE) vo,vo2 : FALSE (TRUE) vo,vo2 : 演算結果 Default (TRUE)
viAct	ディレイ動作選択	BOOL		Default (FALSE) (TRUE) ディレイ動作有効 (FALSE) ディレイ動作無効 (vo = vi)
vi	入力値	BOOL		
viOnDly	ON ディレイ時間 0: ON ディレイ無効	INT	0 ~ 9999	[sec] Default (0) 自動上下限制限
viOffDly	OFF ディレイ時間 0: OFF ディレイ無効	INT	"/"	[sec] Default (0) 自動上下限制限

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	結果	BOOL		
vo2	結果2	BOOL		

**(注意)**

- 初回(viEN=FALSE→TRUE または viEN=TRUE かつ viAct=FALSE→TRUE)で既に vi=TRUE であった場合は FALSE→TRUE に変化したものとみなします。
- viOnDly 中に vi が TRUE→FALSE に変化したら vo=FALSE とします。また、viOffDly 中に vi が FALSE→TRUE に変化した場合は vo=TRUE とします。
- viEN が FALSE である場合は vo,vo2 は FALSE となります。

ENTHALPY [FB]**エンタルピ演算**

乾球温度／相対湿度、又は乾球温度／露点温度からエンタルピを算出します。

ENTHALPY

viEN	voENO
viDB	voERRNO
viRH	vo
viDPT	

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL	FALSE, TRUE	FALSE: vo=0.0 TRUE: vo=演算結果 Default (TRUE)
viDB	乾球温度	REAL	Nan, -20.00 ~ +99.00	[°C] Default (NaN) 自動上下限制限
viRH	相対湿度	REAL	Nan, 0 ~ 100	[%] Default (NaN) 自動上下限制限
viDPT	露点温度	REAL	Nan, -20.00 ~ +99.00	[CDP] Default (NaN) 自動上下限制限

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL	FALSE, TRUE	
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	計算結果(エンタルピ)	REAL	Nan, Floating Point	[kcal/kg]

(STATE)

viDB	viRH	viDPT	vo
NaN	-	-	NA
-	NaN	-	DB, DPT で演算
-	-	NaN	DB, RH で演算
-	-	-	DB, RH で演算 DPT は未使用

NaN には範囲外も含みます

(解説)

ENTHALPY の算出は

- 乾球温度(viDB)、相対湿度(viRH)
 - 乾球温度(viDB)、露点温度(viDPT)
- パラメータ指定により上記2式のうちどちらかを使用します。

FILTER [FB]

一次遅れフィルタ

FILTER

viEN	voENO
vi	voERRNO
viT1	vo
viRST	

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: vo=vi TRUE: vo=計算結果 Default (TRUE)
vi	入力値	REAL	Nan, Floating Point	
viT1	遅れ時間	INT	0 ~ 100	[sec] Default (0) 自動上下限制限
viRST	リセット	BOOL		Default (FALSE) (TRUE) vo = vi (FALSE) vo = 計算結果

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL	FALSE, TRUE	
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	計算結果	REAL	Nan, Floating Point	

(解説)

下記の計算式で結果を返します。

$$vo = voLast + TS / (TS + viT1) * (vi - voLast)$$

voLast: 前回出力値

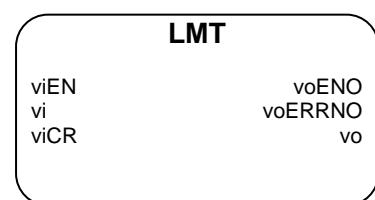
TS: 実行周期

(注意)

- 1) 初回の実行結果は $vo = voLast = vi$ とします。
- 2) vi が NaN であると結果 vo は NaN となります。
- 3) 実行周期(TS)>遅れ時間(viT1)であるときは $viT1=TS$ として計算します。

LMT [FB]**変化量制限**

入力 vi の変化に対して変化量制限を行い vo に出力します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: vo=vi TRUE: vo=計算結果 Default (TRUE)
vi	入力値	REAL	NaN, Floating Point	Default (NaN)
viCR	変化量	REAL	NaN, 0.0 ~ 99999	Default (NaN) 自動上下限制限

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	結果	REAL	NaN, Floating Point	

(解説)

入力 vi の変化量(前回出力値 voLast – 入力 vi)が±viCR 以上の場合には下記の値が出力されます。

上限値: voLast + viCR

下限値: voLast - viCR

voLast: 前回出力値

(注意)

1) 初回(viEN が FALSE→TRUE)の実行結果は vo = voLast = vi とします。

2) vi か viCR が NaN であると結果 vo は NaN となります。

LOAD_RESET [FB]**給気温度最適化制御**

入力(viMX、viMN)により一定周期(viCYC)毎に設定値を一定値(viSTP)だけ UP/DOWN させ出力します。

LOAD_RESET

viEN	viENO
viMX	viERRNO
:	vo

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viMX	UP/DOWN コマンド	BOOL		Default (FALSE)
viMN	DOWN/UP コマンド	BOOL		Default (FALSE)
viHC	冷暖切換入力	BOOL		FALSE: 暖房 TRUE: 冷房 Default (FALSE)
viRST	リセット	BOOL		TRUE: リセット Default(FALSE)
vi	初期値設定	REAL	NaN, -999.9 ~ 999.9	[%] Default (NaN) 自動上下限制限
viCYC	実行周期	INT	0 ~ 9999	[s] Default (600) 自動上下限制限 0: Program scan
viSTP	増加ステップ幅	REAL	NaN, -999.9 ~ 999.9	[%] Default (1.0) 自動上下限制限
viMAX	出力最大値	REAL	NaN, -999.9 ~ 999.9	[%] Default (100.0) 自動上下限制限
viMIN	出力最小値	REAL	NaN, -999.9 ~ 999.9	[%] Default (-100.0) 自動上下限制限

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	出力	REAL	NaN, -999.9 ~ 999.9	

(解説)

- ① 実行周期(viCYC)毎に UP/DOWN コマンド(viMX,viMN)の指令により出力を指定のステップ幅(viSTP)だけ増減します。

●暖房時 (viHC = FALSE)

viMX	viMN	出力
TRUE	FALSE	viCYC 毎に viSTP 分だけ出力を増加する。 出力 = 前回出力 + ステップ幅(viSTP)
FALSE	TRUE	viCYC 毎に viSTP 分だけ出力を減少する。 出力 = 前回出力 - ステップ幅(viSTP)
TRUE	TRUE	前回出力を保持
FALSE	FALSE	" "

●冷房時 (viHC = TRUE)

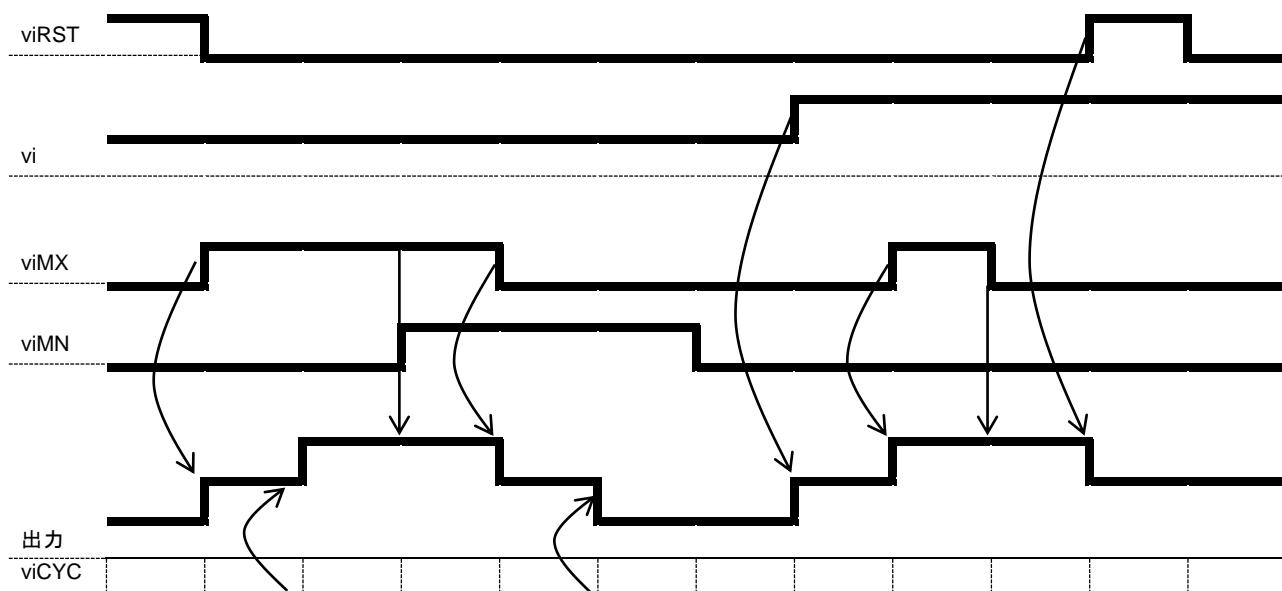
viMX	viMN	出力
TRUE	FALSE	viCYC 毎に viSTP 分だけ出力を減少する。 出力 = 前回出力 - ステップ幅(viSTP)
FALSE	TRUE	viCYC 毎に viSTP 分だけ出力を増加する。

1. ライブラリ詳細

		出力 = 前回出力 + ステップ幅(viSTP)
TRUE	TRUE	前回出力を保持
FALSE	FALSE	"

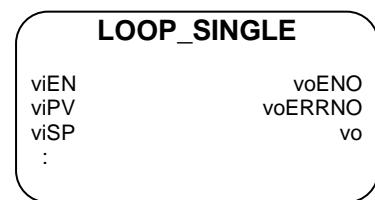
- ② リセット(viRST = TRUE)の時
初期値設定(vi)をそのまま出力します。
- ③ パワーオンリセット時
初期値設定(vi)をそのまま出力します。
- ④ 初期値設定(vi)の変化時
初期値設定(vi)をそのまま出力します。
- ⑤ 出力は最大値(viMAX)、最小値(viMIN)により値が制限されます。
- ⑥ vi,viSTP,viMAX,viMIN の何れかが NaN なら出力(vo)に NaN が出力されます。
- ⑦ 上記②、③、④、⑥は実行周期に関係なく評価されます。また実行周期の計測はその時点から再計算(リセット)されます。

「暖房時のタイミングチャート」



LOOP_SINGLE [FB]**PID 演算**

プロセス入力(viPV)と設定入力(viSP)との偏差により PID 演算を行います。

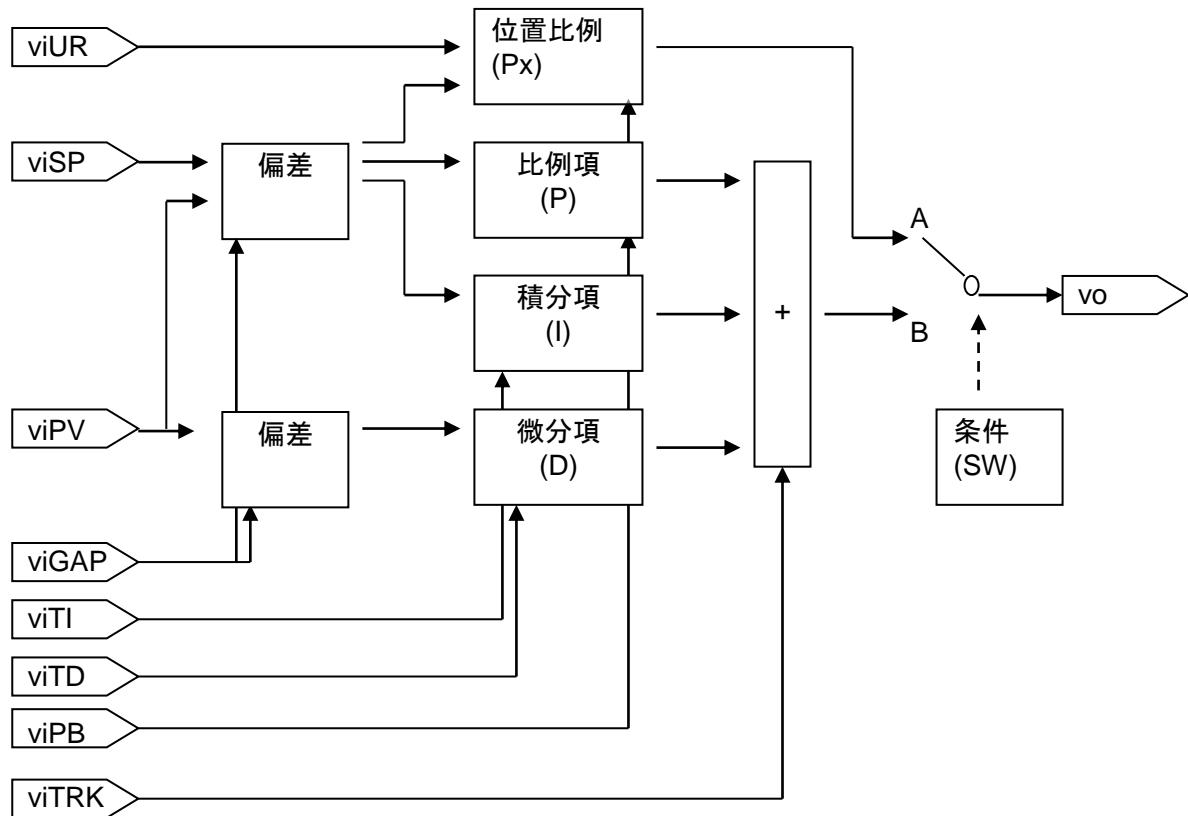
**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: vo=0.0 TRUE: vo=演算結果 Default (TRUE)
viPV	プロセス入力	REAL	NaN, -100.00 ~ +100.00	[°C] Default(NaN) 自動上下限制限
viSP	設定入力	REAL	" "	[°C] Default(0.0) 自動上下限制限
viTR	トラッキング入力	REAL	NaN, 0 ~ 100.0	[%] Default(NaN) NAN:Disable 自動上下限制限
viST	インターロック ロックアウト入力	BOOL		Default(FALSE) (TRUE) ロックアウト状態 vo=比例制御演算 (FALSE) vo=演算出力
viPB	比例帯	REAL	NaN, 0 ~ 999.9	[%] Default(5) 0 or NaN:Disable 自動上下限制限
viTI	積分時間	INT	0 ~ 9999	[sec] Default(900) 自動上下限制限 0:Disable 位置比例制御とする
viTD	微分時間	INT	0 ~ 9999	[sec] Default(0) 自動上下限制限 0:Disable
viUR	リセット値	REAL	NaN, 0 ~ 100.0	[%] Default(50) 自動上下限制限
viGAP	偏差ギャップ	REAL	NaN, 0 ~ 999.9	[%] Default(NaN) 自動上下限制限
viDIR	正逆動作選択	BOOL		Default(FALSE) FALSE:逆 TRUE:正
viSKP	演算スキップ回数	INT	0 ~ 9999	[回] Default(0) 自動上下限制限

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	計算結果	REAL	0 ~ 100.0	[%]

(解説)



■演算エラー時の voENO

以下の条件で voENO=False、vo=0.0 に設定されます。

- 1) viPV,viSP が NaN の場合
- 2) 比例帯(viPB)が 0.0% の場合
- 3) 演算結果 vo が +-999.9% を超えた場合

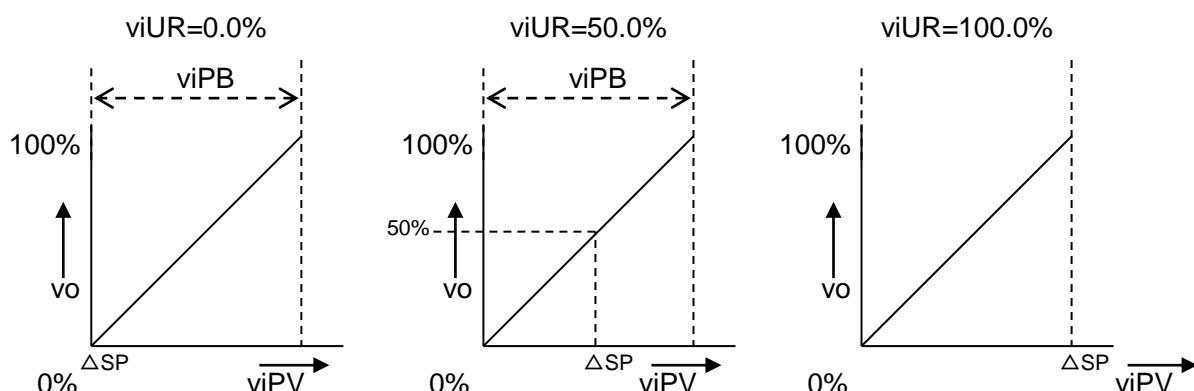
■条件 SW

以下のいずれかの条件で A(位置比例)を選択します。

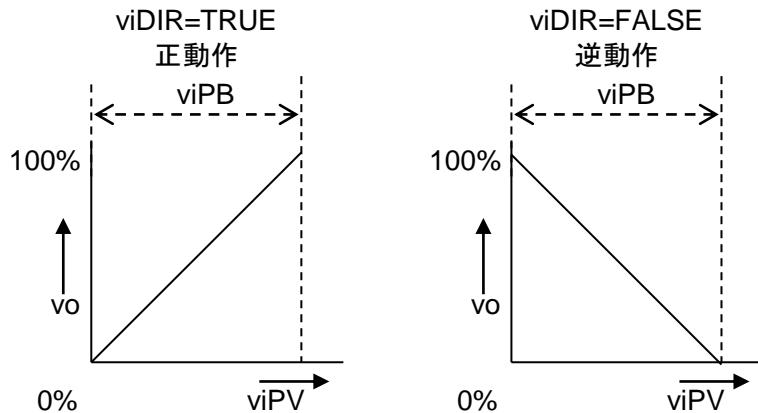
- 1) イニシャル時(初回演算または viPV,viSP の前回値が NaN である時)
- 2) 積分時間 viTI が 0
- 3) インターロック入力 viST が TRUE

■リセット値 viUR について

位置比例制御を行うときの出力オフセットを指定します。



■正逆動作選択 viDIR について
PID および位置比例制御の動作方向を指定します。



MOMENTARY_OUTPUT [FB]**モメンタリ出力****(解説)**

入力をモメンタリ出力に変換します。

MOMENTARY_OUTPUT

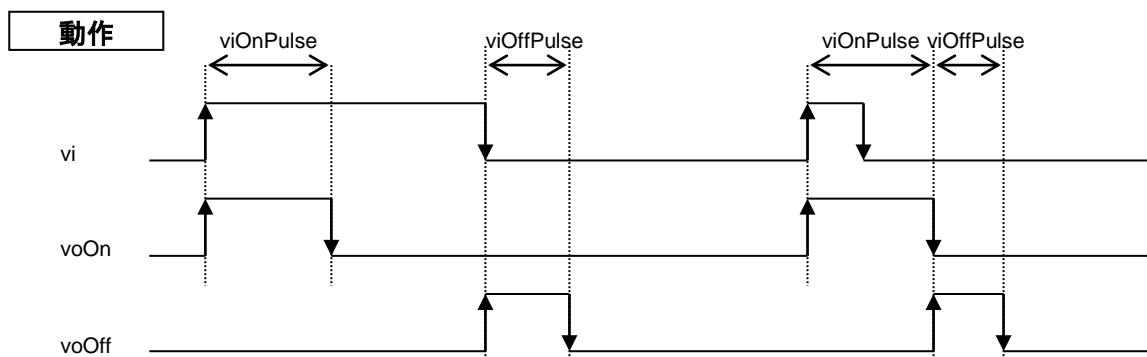
viEN	voENO
vi	voERRNO
:	voOn voOff

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
vi	入力	BOOL		
viOnPulse	On パルス時間	REAL	Nan, 0.1 ~ 10.0	[sec] Default (NaN) 自動上下限制限 NAN:no start pulse
viOffPulse	Off パルス時間	REAL	Nan, 0.1 ~ 10.0	[sec] Default (NaN) 自動上下限制限 Nan:no stop pulse

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		FALSE: NG TRUE: OK
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voOn	Start 出力	BOOL		
voOff	Stop 出力	BOOL		

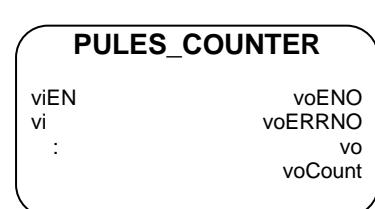
**(注意)**

- 初回(viEN=FALSE→TRUE)で既に vi=TRUE であった場合は FALSE→TRUE に変化したものとみなします。
- viOnPulse 中に vi が TRUE→FALSE あるいは viOffPulse 中に vi が FALSE→TRUE に変化しても Pulse 出力は継続します。このとき viOnPulse 出力完了時点の vi が FALSE であれば続いて viOffPulse が 出力されます。また、viOffPulse 出力完了時点の vi が TRUE であれば続けて viOnPulse が 出力されます。
- viEN が FALSE である場合は voOn,voOff は即 FALSE となります。

PULSE_COUNTER [FB]**パルスカウント****(解説)**

パルスを入力し演算結果を vo に出力します。

vo = viPulse * (1 / (t1 - t2))
 t1: 今回パルス FALSE→TRUE 時刻[sec]
 t2: 前回パルス FALSE→TRUE 時刻[sec]

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
vi	パルス入力	BOOL		Default (FALSE)
viReset	カウンタリセット	BOOL		Default (FALSE)
viPulse	パルスカウント定数	REAL	Floating Point	Default (0)

(OUTPUT)

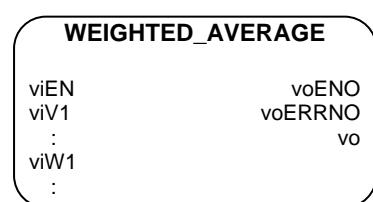
記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		FALSE: NG TRUE: OK
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	レート出力 [Pulse / sec]	REAL	Floating Point	
voCount	カウント出力	UDINT	0 ~ 999,999,999	上限を超えると 0 に戻る

- パルスは 50% duty サイクルであるものとして演算します。
- 最大パルスレートは (このファンクションブロックの置かれたプログラムのスキャン周期に依存します)
例えば 100ms スキャン周期であれば
 $1\text{sec} / (100\text{ms} * 1/(50\%)) = 5 \text{ [pulse / sec]}$
 が最大パルスレートとなります。
- 最小パルスレートは (このファンクションブロックの置かれたプログラムのスキャン周期に依存します)
 $1\text{pulse} / 5\text{min} (300\text{sec})$ です。
 この最小パルスを超える場合は vo に 0 が出力されます。
- viReset は TRUE の間カウント出力を 0 に設定します。
- カウント出力(voCount)は、パルス入力の FALSE→TRUE の検出で1加算されます。

WEIGHTED_AVERAGE [FB]**加重平均**

入力 viV1~4 の加重平均演算を行い戻り値(Return)に出力します。

$$vo = \frac{\sum_{n=1}^4 viVn \times viWn}{\sum_{n=1}^4 viWn}$$

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		(FALSE) vo : FALSE (TRUE) vo : 演算結果 Default (TRUE)
viV1 .. 4	入力 1~4	REAL	NaN, Floating point	Default (NaN)
viW1 .. 4	加重 1~4	REAL	NaN, Floating point	Default (NaN)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	結果	REAL	NaN, Floating point	

(解説)

入力 viV1 に対する加重は viW1 に入力します。対応する入力あるいは加重が NaN 値の場合は演算対象から除かれます。

(注意)

- 1) 演算対象が無い場合は結果として NaN 値を返します。
- 2) 加重合計が0の場合は結果として0を返します。

IS_REAL_NAN [FN]

REAL 型変数値の NaN 判定

(解説)

指定 REAL 型変数が NaN 値であるか判定し結果を返します。

IS_REAL_NAN

vi

(return)

(INPUT)

記号	解説	型	範囲	備考
vi	判定対象変数	REAL		

(OUTPUT)

記号	解説	型	範囲	備考
(return)	判定結果	BOOL		FALSE (Not NaN) TRUE (NaN)

SET_REAL_NAN [FN]

REAL 型変数値を NaN に設定

(解説)

指定 REAL 型変数を NaN 値に設定します。

SET_REAL_NAN

(return)

(INPUT)

記号	解説	型	範囲	備考
なし				

(OUTPUT)

記号	解説	型	範囲	備考
(return)	対象変数	REAL		

POINT_HISTORY [FB]**変数値の履歴書き込み****(解説)**

指定の値を履歴に書き込みます。

POINT_HISTORY

viEN	voENO
viUnit	voERRNO
viBoolValue1	voCount
:	

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viUnit	履歴領域番号	INT	1 ~ 50	Default(0) (*1)
viRealValue	対象の値(REAL)	REAL	Floating Point	Default(NaN)
viUSintValue	対象の値(USINT)	USINT		Default(0)
viBoolValue1	対象の値(BOOL)	BOOL		Default(FALSE)
viBoolValue2	対象の値(BOOL)	BOOL		Default(FALSE)
viBoolValue3	対象の値(BOOL)	BOOL		Default(FALSE)
viBoolValue4	対象の値(BOOL)	BOOL		Default(FALSE)
viRing	Enable RingBuffer	BOOL		Default(FALSE) (*1) TRUE:Ring Buffer
viReset	リセット(履歴削除)	BOOL		FALSE→TRUE Default(FALSE)
viSampling	サンプリング間隔(秒)	INT	0 ~ 1000	Default(0) (*1) 0:Program Scan
viTitle	履歴タイトル	STR	文字列	max. 32 characters (*1) Default("")

*1) viEN 立ち上がり時に記憶

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error
voCount	履歴件数	INT	0 ~ 100	

RTC_NOW [FB]

リアルタイムクロックの読み出し

(解説)

リアルタイムクロックより現在時刻を取得します。

RTC_NOW

viEN	voENO voERRNO voYear voHour :
------	---

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)

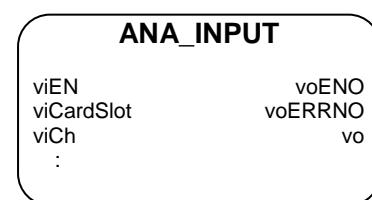
(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		=viEN
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voYear	年	INT	1900 ~ 2070	
voMonth	月	INT	1 ~ 12	
voDay	日	INT	1 ~ 31	
voHour	時	INT	0 ~ 23	
voMinute	分	INT	0 ~ 59	
voSecond	秒	INT	0 ~ 59	
voDayOfWeek	曜日	INT	0 ~ 6	0:Sunday, 6:Saturday
voDayOfYear	年間積算日	INT	1 ~ 366	

ANA_INPUT [FB]**ハードウェアからのアナログ入力****(解説)**

下記の計算式で結果を vo に出力します。

```
vo = Limit2( (Limit1(X + b) - viL_Raw) * a + viL_Scale + of)
X: ハードウェア入力値
a: (viH_Scale - viL_Scale) / (viH_Raw - viL_Raw)
b: viBase
of: viOffset
Limit1: viH_Raw ~ viL_Raw
Limit2: viH_Scale ~ viL_Scale
```

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viCardSlot	カードスロット番号	INT	0,1 ~ 16	Default (0) 0 は予約 (*1)
viCh	チャネル番号	INT	1 ~ 16	Default (0) (*1)
viBase	ハードウェア入力のベース値	REAL	Floating Point	Default (0.0)
viL_Raw	ハードウェア入力の下限値	REAL	Floating Point	Default (0.0)
viH_Raw	ハードウェア入力の上限値	REAL	Floating Point	Default (10000.0)
viL_Scale	ハードウェア入力の下限に割り当てる値	REAL	Floating Point	Default (0.0)
viH_Scale	ハードウェア入力の上限に割り当てる値	REAL	Floating Point	Default (100.00)
viOffset	内部計算結果に対してこのオフセットを加算し出力(vo) します。 (現物合わせ調整用)	REAL	Floating Point	Default (0.0)
viT1	ハードウェア入力の一次遅れフィルタの遅れ時間 [sec]	INT	0 ~ 100	Default (0) 自動上下限制限 (*1)

*1)viEN 立ち上がり時に記憶

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		FALSE: NG TRUE: OK
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	出力	REAL	Nan, Floating Point	演算エラーは Nan

(補足)**■温度データ(R3-TS, R3-RS など)**

温度データは単位が摂氏(°C)の場合、温度を 10 倍した値がデータとなります。

例えば 23.4°C の場合には 234(10 進)がデータ(Raw)となります。

■R3 カード(上記以外)からの入力データ

入力レンジに対して 0 ~ 100% が 0 ~ 10000 (10 進)が対応したデータ(Raw)となります。

例えば、この入力をプログラムで%値として使用するときは

viL_Raw = 0.0 , viH_Raw = 10000.0 , viL_Scale = 0.0 , viH_Scale = 100.0 または
viL_Raw = -1500.0 , viH_Raw = 11500.0 , viL_Scale = -15.0 , viH_Scale = 115.0 と指定します。

ANA_INPUT_UI32 [FB]

ハードウェアからの 32bit アナログ入力

(解説)

下記の計算式で結果を vo に出力します。

```
vo = Limit( vi + viOffset )
Limit: 0 ~ 0xffffffff
```

ANA_INPUT_UI32

viEN	voENO
viCardSlot	voERRNO
viCh	vo
:	

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viCardSlot	カードスロット番号	INT	0,1 ~ 16	Default (0) 0 は予約 (*1)
viCh	チャネル番号	INT	1 ~ 8	Default (0) (*1)
viOffset	内部計算結果に対してこのオフセットを加算し出力(vo) とします。 (現物合わせ調整用)	DINT		Default (0)

*1)viEN 立ち上がり時に記憶

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		FALSE: NG TRUE: OK
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	出力	UDINT		

(補足)

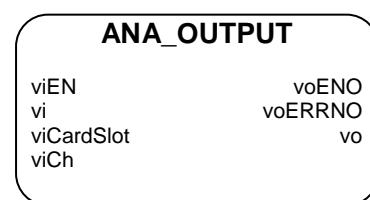
ANA_OUTPUT [FB]**ハードウェアへのアナログ出力****(解説)**

下記の計算式で結果をハードウェアに出力します。

```

vo = Limit1( (Limit2(vi - of) - viL_Scale) / a + viL_Raw + b)
a: (viH_Scale - viL_Scale) / (viH_Raw - viL_Raw)
b: viBase
of: viOffset
Limit1: viH_Raw ~ viL_Raw
Limit2: viH_Scale ~ viL_Scale

```

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
vi	出力値	REAL	Floating Point	Default (NaN) NaN の場合は 0.0 とする
viCardSlot	カードスロット番号	INT	0,1 ~ 16	Default (0) 0 は予約 (*1)
viCh	チャネル番号	INT	1 ~ 16	Default (0) (*1)
viBase	ハードウェア出力のベース値	REAL	Floating Point	Default (0.0)
viL_Raw	ハードウェア出力の下限値	REAL	Floating Point	Default (10000.0)
viH_Raw	ハードウェア出力の上限値	REAL	Floating Point	Default (0.0)
viL_Scale	ハードウェア出力の下限に割り当てる値	REAL	Floating Point	Default (0.0)
viH_Scale	ハードウェア出力の上限に割り当てる値	REAL	Floating Point	Default (100.00)
viOffset	内部計算結果に対して加算されていたオフセット (現物合わせ調整用)	REAL	Floating Point	Default (0.0)

*1) viEN 立ち上がり時に記憶

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		FALSE: NG TRUE: OK
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	ハードウェア出力値のコピー	REAL	NaN, Floating Point	演算エラーは NaN

(補足)**■R3 カードへの出力データ**

出力レンジに対して 0 ~ 100% が 0 ~ 10000 (10 進)が対応したデータ(Raw)となります。

例えば、プログラムから%値で出力するときは

viL_Raw = 0.0 , viH_Raw = 10000.0 , viL_Scale = 0.0 , viH_Scale = 100.0 と指定します。

ANA_OUTPUT_UI32 [FB]

ハードウェアへの 32bit アナログ出力

(解説)

下記の計算式で結果をハードウェアに出力します。

```
vo = Limit( vi + viOffset )
Limit: 0 ~ 0xffffffff
```

ANA_OUTPUT_UI32

viEN	voENO
vi	voERRNO
viCardSlot	vo
:	

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
vi	出力値	UDINT		Default (0)
viCardSlot	カーデスロット番号	INT	0,1 ~ 16	Default (0) 0 は予約 (*1)
viCh	チャネル番号	INT	1 ~ 8	Default (0) (*1)
viOffset	内部計算結果に対してこのオフセットを加算し出力(vo) とします。 (現物合わせ調整用)	DINT		Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		FALSE: NG TRUE: OK
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	ハードウェア出力値のコピー	UDINT		

(補足)

DIG_INPUT [FB]

ハードウェアからのデジタル入力

(解説)

ハードウェアから入力し結果を vo に出力します。

DIG_INPUT

viEN	voENO
viCardSlot	voERRNO
viCh	vo
:	

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viCardSlot	カードスロット番号	INT	0,1 ~ 16	Default (0) 0 は予約 (*1)
viCh	チャネル番号	INT	1 ~ 64	Default (0) (*1)
viType	入力信号タイプ	INT	0:Direct (N.O.) 1:Reverse (N.C.)	Default (0)

*1) viEN 立ち上がり時に記憶

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		FALSE: NG TRUE: OK
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	出力	BOOL		

DIG_OUTPUT [FB]**ハードウェアへのデジタル出力****(解説)**

ハードウェアに出力します。

DIG_OUTPUT

viEN	voENO
vi	voERRNO
viCardSlot	vo
viCh	:

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
vi	出力値	BOOL		Default (FALSE)
viCardSlot	カーデスロット番号	INT	0,1 ~ 16	Default (0) 0 は予約 (*1)
viCh	チャネル番号	INT	1 ~ 64	Default (0) (*1)
viType	出力信号タイプ	INT	0:Direct (N.O.), 1:Reverse (N.C.)	Default (0)

*1) viEN 立ち上がり時に記憶

(OUTPUT)

記号	解説	型	範囲	備考
voENO	Enable Output	BOOL		FALSE: NG TRUE: OK
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	ハードウェア出力値のコピー	BOOL		演算エラーは FALSE

LON_NV_INFO [FB]**LON 変数情報の取得****(解説)**

指定 LON 変数の情報を取得します。

LON_NV_INFO

viEN	voENO
viNvIndex	voERRNO
	vo_xxxx
	:

(INPUT)

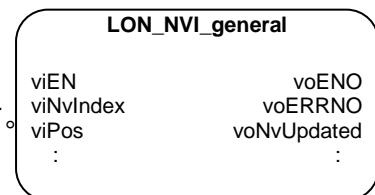
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 立上がり時演算 Default (TRUE)
viNvIndex	NV Index	INT	0 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo_type_program_ID	Program ID	STR	空	予約(固定値が返る)
vo_type_scope	Scope	USINT	0	予約(固定値が返る)
vo_type_index	Network variable Type	UINT	1 ~ 65,535	失敗は 0 として扱い ENO=FALSE
vo_type_category	Category	UINT	0	予約(固定値が返る)
vo_type_length	Length	USINT	1 ~ 31	失敗は 0 として扱い voENO=FALSE
vo_scaling_factor_a	Scaling factor a ScaledValue = a * 10b * (RawValue + c)	INT	0	予約(固定値が返る)
vo_scaling_factor_b	Scaling factor b	INT	0	予約(固定値が返る)
vo_scaling_factor_c	Scaling factor c	INT	0	予約(固定値が返る)

LON_NVI_general [FB]**LonWorks input variable 値取得****(解説)**

Lonworks 入力変数(最大 31 バイト)から指定位置の値を取得します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
viPos	変数(最大 31 バイト)内のデータ開始バイト位置	INT	0 ~ 30	Default(0)
viDataType	取得データ型	INT	1 ~ 4	1: BYTE (1 byte) 2: WORD (2 bytes) 3: DWORD (4 bytes) 4: FLOAT (Real) Default(0)
viTmpUpdate	実変数から作業領域への更新(コピー)指定 (複数要素で構成された変数アクセスで重要です)	BOOL		FALSE: 更新しない TRUE: 更新する Default (TRUE)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
voByteData	BYTE 指定時の取得結果	BYTE		
voWordData	WORD 指定時の取得結果	WORD		
voDWordData	DWORD 指定時の取得結果	DWORD		
voFloatData	FLOAT 指定時の取得結果	REAL		

(補足)**■viPos 指定範囲について**

指定の範囲は(viPos + viDataType のデータ長) <= 31 である必要があります。これを超える指定はエラーとなります。

■取得値のエンディアンについて

LonWorks 変数値のデータの並び順は、ビッグエンディアンです。このファンクションブロックでは、2バイト以上のデータ型を扱うために voWordData, voDWordData, voFloatData のメンバーを用意しています。

■取得値のビットについて

LonWorks 変数値でデータでビット要素が含まれるタイプがあります。LonWorks 変数値でのビット要素はバイトの最上位ビットから割り当てられますので、本コントローラでの取り扱いは注意が必要です。

例) SNVT_reg_val (136) の要素

```
unsigned short nr_decimals : 3;
```

この場合のデータは bit5,6,7 の3ビットとなり、nr_decimals の値1は 16#20、値7は 16#e0 となります。

例) SNVT_state (83) の要素

```
unsigned bit0 : 1;
```

```
unsigned bit1 : 1;
```

```
:
```

```
unsigned bit15 : 1;
```

この場合のデータは bit0 の値 1 は 16#80、bit1 の値 1 は 16#40、bit15 の値 1 は 16#01 となります。

SNVT_switch (95) の現在値 100:01 の value の取得を例に説明します。
 (Version 13 Revision 00 Janualy 2006 SNVT Master List より)

```
typedef struct {
    unsigned          value;
    signed           state;
} SNVT_switch;
```

value: Intensity as percentage of full scale, resolution 0.5%			
Field	Measurement	Field Type Category	Field Size
value	Value	Unsigned Short	1 byte
Valid Type Range	Type Resolution	Units	Invalid Value
0 .. 100	0.5	Percent of Full Scale	
Raw Range	Scale Factors	File Name	Default Value
0 .. 200 (0 .. 0xC8)	5, -1, 0 $S = a \cdot 10^b \cdot (R+c)$	N/A	N/A

state: This field can either be -1 (NULL), 0 (OFF), or 1 (ON)			
Field	Measurement	Field Type Category	Field Size
state	State	Signed Short	1 byte
Valid Type Range	Type Resolution	Units	Invalid Value
0 .. 1	1	N/A	-1 (0xFF)
Raw Range	Scale Factors	File Name	Default Value
0 .. 1	1, 0, 0 $S = a \cdot 10^b \cdot (R+c)$	N/A	N/A

Network 変数から値を取得する方法を示します。

ここでの例は Network 変数の中でも複雑な変数(複数要素で構成される Type Category : structure)で説明します。

① value の取得

viPos=0, viDataType=1(BYTE), viTmpUpdate=TRUE とすることで voByteData=200 が返ります。
 取得した値を上記表より演算して value の値を求めます: $200 * 0.5$ (Type Resolution) = 100

② state の取得

viPos=1, viDataType=1(BYTE), viTmpUpdate=FALSE とすることで voByteData=1 が返ります。
 取得した値を上記表より演算して state の値を求めます: $1 * 1$ (Type Resolution) = 1

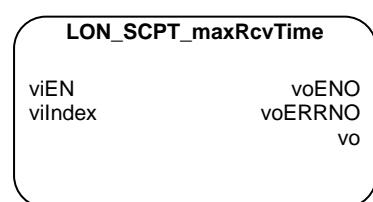
追記)

1) voNvUpdated は viTmpUpdate=TRUE のインスタンスに結果が反映されます。ただし、同一の viNvIndex を指定して、かつ viTmpUpdate=TRUE のインスタンスが複数存在する場合は、実変数が更新された直後に実行された最初のインスタンスの voNvUpdated が更新されます。その他のインスタンスの voNvUpdated には FALSE が返されます。

2) 上記①で viTmpUpdate=TRUE を指定し②では viTmpUpdate=FALSE を指定することは重要です。これは①で実変数から作業領域に値を取得し、次の②ではその作業領域から値を取得します。
 これは①と②の間に実変数の更新が有った場合に旧 value と新 state のように取得されることを防ぎます(この例では旧 value とその時の state、すなわち旧 state が取得されます)。

LON_SCPT_maxRcvTime [FB]**LonWorks CP variable 値取得****(解説)**

SCPT_maxRcvTime(48) 入力変数から値を取得します。

**(INPUT)**

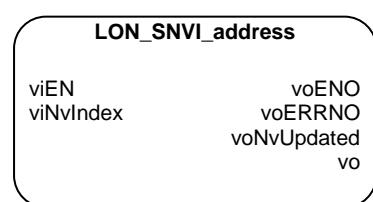
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
vilIndex	Index	INT	0 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
vo	Maximum Receive Time Resolution: IEEE 754	REAL	Floating point	0 ~ 6553.4, 6553.5

LON_SNVI_address [FB]**LonWorks input variable 値取得****(解説)**

SNVT_address(114) 入力変数から値を取得します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Neuron Chip Address Resolution: 1	UINT	16,384 ~ 64,767	失敗は 0 として扱い voENO=FALSE

LON_SNVI_count [FB]**LonWorks input variable 値取得****(解説)**

SNVT_count(8) 入力変数から値を取得します。

LON_SNVI_count

viEN	voENO
viNvIndex	voERRNO
	voNvUpdated
	vo

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Event Count Resolution: 1	UINT	0 ~ 65,535	失敗は 0 として扱い voENO=FALSE

LON_SNVI_count_inc [FB]**LonWorks input variable 値取得****(解説)**

SNVT_count_inc(9) 入力変数から値を取得します。

LON_SNVI_count_inc

viEN	voENO
viNvIndex	voERRNO
	voNvUpdated
	vo

(INPUT)

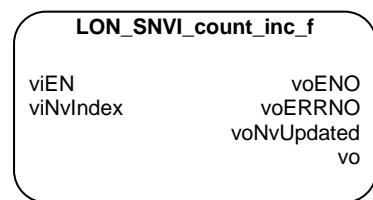
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Incremental Count Resolution: 1	INT	-32,768 ~ +32,767	失敗は 0 として扱い ENO=FALSE

LON_SNVI_count_inc_f [FB]**LonWorks input variable 値取得****(解説)**

SNVT_count_inc_f(52) 入力変数から値を取得します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

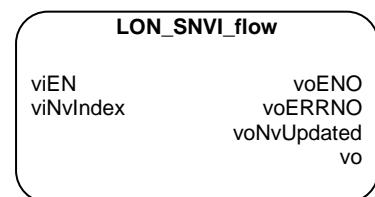
(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Incremental Count Resolution: IEEE 754	REAL	Floating Point	入力がNaNは0として扱い voENO=FALSE

注) LON 入力変数値が出力 vo の範囲外である場合は上限値あるいは下限値とし失敗 voENO=FALSE とする。

LON_SNVI_flow [FB]**LonWorks input variable 値取得****(解説)**

SNVT_flow(15) 入力変数から値を取得します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Flow Volume Resolution: 1	UINT	0 ~ 65,534	入力が 65,535 は 0 として扱い voENO=FALSE

LON_SNVI_hvac_emerg [FB]**LonWorks input variable 値取得****(解説)**

SNVT_hvac_emerg(103) 入力変数から値を取得します。

LON_SNVI_hvac_emergviEN
viNvIndexvoENO
voERRNO
voNvUpdated
vo**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	出力 Resolution: 1	USINT	Enumerated EMERG_xxx	失敗は 255 として扱い voENO=FALSE

LON_SNVI_hvac_mode [FB]**LonWorks input variable 値取得****(解説)**

SNVT_hvac_mode(108) 入力変数から値を取得します。

LON_SNVI_hvac_mode

viEN	voENO
viNvIndex	voERRNO
	voNvUpdated
	vo

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	出力 Resolution: 1	USINT	Enumerated HVAC_xxx	失敗は 255 として扱い voENO=FALSE

LON_SNVI_hvac_status [FB]**LonWorks input variable 値取得****(解説)**

SNVT_hvac_status(112) 入力変数から値を取得します。

LON_SNVI_hvac_status

viEN	voENO
viNvIndex	voERRNO
	voNvUpdated
	vo_xxxx

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo_mode	HVAC status mode Resolution: 1	USINT	Enumerated HVAC_xxx	失敗は 255 として扱い voENO=FALSE
vo_heat_output_primary	Primary heat output Resolution: 0.005	REAL	-163.840 ~ +163.830	入力が 32,767 は 0 として扱い voENO=FALSE
vo_heat_output_secondary	Secondary heat output Resolution: 0.005	REAL	-163.840 ~ +163.830	入力が 32,767 は 0 として扱い voENO=FALSE
vo_cool_output	Cooling output (primary) Resolution: 0.005	REAL	-163.840 ~ +163.830	入力が 32,767 は 0 として扱い voENO=FALSE
vo_econ_output	Economizer output Resolution: 0.005	REAL	-163.840 ~ +163.830	入力が 32,767 は 0 として扱い voENO=FALSE
vo_fan_output	Fan output Resolution: 0.005	REAL	-163.840 ~ +163.830	入力が 32,767 は 0 として扱い voENO=FALSE
vo_in_alarm	Alarm state	USINT	0 / 1 / 255	0:not in an alarm, 1 to 254:in Alarm, 255:disabled alarm

LON_SNVI_lev_cont [FB]**LonWorks input variable 値取得****(解説)**

SNVT_lev_cont(21) 入力変数から値を取得します。

LON_SNVI_lev_cont

viEN
viNvIndex

voENO
voERRNO
voNvUpdated
vo

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Continuous Level Resolution: 0.5	REAL	0.0 ~ 100.0	失敗は 0 として扱い voENO=FALSE

LON_SNVI_lev_cont_f [FB]**LonWorks input variable 値取得****(解説)**

SNVT_lev_cont_f(55) 入力変数から値を取得します。

LON_SNVI_lev_cont_f

viEN	voENO
viNvIndex	voERRNO
	voNvUpdated
	vo

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Countinuous Level Resolution: IEEE 754	REAL	0.0 ~ 100.0	入力がNaNは0として扱い voENO=FALSE

注) LON 入力変数値が出力 vo の範囲外である場合は上限値あるいは下限値とし失敗 voENO=FALSE とする。

LON_SNVI_lev_percent [FB]**LonWorks input variable 値取得****(解説)**

SNVT_lev_percent(81) 入力変数から値を取得します。

LON_SNVI_lev_percent

viEN	voENO
viNvIndex	voERRNO
	voNvUpdated
	vo

(INPUT)

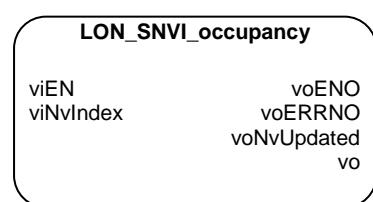
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Percent level Resolution: 0.005	REAL	-163.840 ~ +163.830	入力が 32,767 は 0 として扱い voENO=FALSE

LON_SNVI_occupancy [FB]**LonWorks input variable 値取得****(解説)**

SNVT_occupancy(109) 入力変数から値を取得します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	出力 Resolution: 1	USINT	Enumerated OC_xxx	失敗は 255 として扱い voENO=FALSE

LON_SNVI_ppm [FB]**LonWorks input variable 値取得****(解説)**

SNVT_ppm(29) 入力変数から値を取得します。

```

graph LR
    subgraph LON_SNVI_ppm [LON_SNVI_ppm]
        direction TB
        viEN[viEN] --- viNvIndex[viNvIndex]
        voENO[voENO] --- voERRNO[voERRNO] --- voNvUpdated[voNvUpdated] --- vo[vo]
    end

```

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

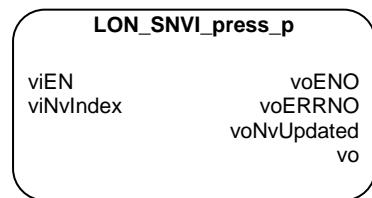
記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Concentration Resolution: 1	UINT	0 ~ 65,535	失敗は 0 として扱い voENO=FALSE

LON_SNVI_press_p [FB]

LonWorks input variable 值取得

(解説)

SNVT press p(113) 入力変数から値を取得します。



(INPUT)

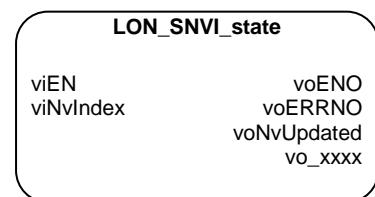
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Pressure (gauge) Resolution: 1	INT	-32,768 ~ +32,766	入力が 32,767 は 0 とし て扱い voENO=FALSE

LON_SNVI_state [FB]**LonWorks input variable 値取得****(解説)**

SNVT_state(83) 入力変数から値を取得します。

**(INPUT)**

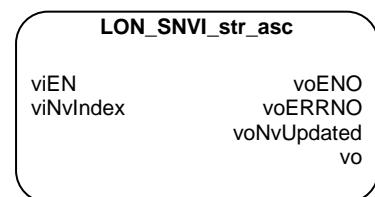
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo_bit0	bit 0	BOOL		0:FALSE, 1:TRUE
vo_bit1	bit 1	"		"
vo_bit2	bit 2	"		"
vo_bit3	bit 3	"		"
vo_bit4	bit 4	"		"
vo_bit5	bit 5	"		"
vo_bit6	bit 6	"		"
vo_bit7	bit 7	"		"
vo_bit8	bit 8	"		"
vo_bit9	bit 9	"		"
vo_bit10	bit 10	"		"
vo_bit11	bit 11	"		"
vo_bit12	bit 12	"		"
vo_bit13	bit 13	"		"
vo_bit14	bit 14	"		"
vo_bit15	bit 15	"		"

LON_SNVI_str_asc [FB]**LonWorks input variable 値取得****(解説)**

SNVT_str_asc(36) 入力変数から値を取得します。

**(INPUT)**

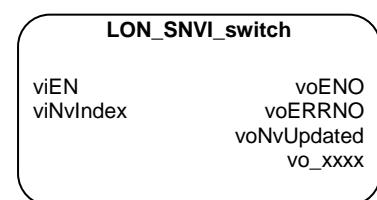
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	結果文字列	STRING	Max 30char	

LON_SNVI_switch [FB]**LonWorks input variable 値取得****(解説)**

SNVT_switch(95) 入力変数から値を取得します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

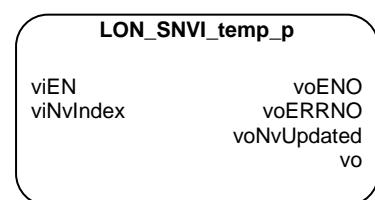
(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo_value	value Resolution: 0.5	REAL	0.0 ~ 100.0	
vo_state	state	USINT	0 / 1 / 255	失敗は 255 として扱い voENO=FALSE
vo	解釈	BOOL		

vo_value	vo_state	解釈 [vo]
any	0	off (0.0:0) [FALSE]
0	1	off (0.0:1) [FALSE]
>0	1	on (100.0:1) [TRUE]
any	255	invalid [FALSE] voENO=FALSE にし off (0:0) とする

LON_SNVI_temp_p [FB]**LonWorks input variable 値取得****(解説)**

SNVT_temp_p(105) 入力変数から値を取得します。

**(INPUT)**

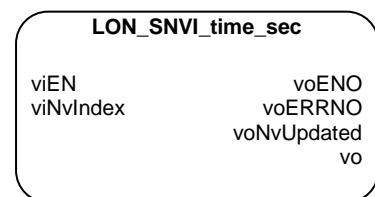
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Temperature Resolution: 0.01	REAL	-273.17 ~ +327.66	入力が 32,767 は 0 とし て扱い voENO=FALSE

LON_SNVI_time_sec [FB]**LonWorks input variable 値取得****(解説)**

SNVT_time_sec(107) 入力変数から値を取得します。

**(INPUT)**

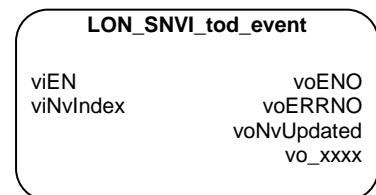
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Elapsed Time (seconds) Resolution: 0.1	REAL	0.0 ~ 6553.4	入力が 65,535 は 0 とし て扱い voENO=FALSE

LON_SNVI_tod_event [FB]**LonWorks input variable 値取得****(解説)**

SNVT_tod_event(128) 入力変数から値を取得します。

**(INPUT)**

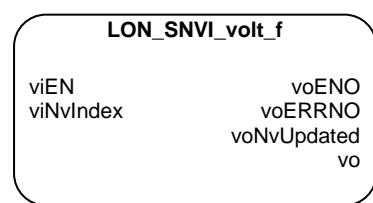
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo_current_state	Occupancy, current Resolution: 1	USINT	Enumerated OC_xxx	失敗は 255 として扱い voENO=FALSE
vo_next_state	Occupancy, next Resolution: 1	USINT	Enumerated OC_xxx	失敗は 255 として扱い voENO=FALSE
vo_time_to_ next_state	Time to next state Resolution: 1	UINT	0 ~ 65,535	

LON_SNVI_volt_f [FB]**LonWorks input variable 値取得****(解説)**

SNVT_volt_f(66) 入力変数から値を取得します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)

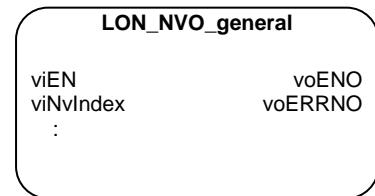
(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態(成功は TRUE, 失敗は FALSE)	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error
voNvUpdated	値更新	BOOL		更新検出時に TRUE
vo	Electric Voltage Resolution: IEEE 754	REAL	Floating Point	入力がNaNは0として扱い voENO=FALSE

注) LON 入力変数値が出力 vo の範囲外である場合は上限値あるいは下限値とし失敗 voENO=FALSE とする。

LON_NVO_general [FB]**LonWorks output variable 値設定****(解説)**

LonWorks 出力変数(最大 31 バイト)の指定位置へ値を設定します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE: 演算スキップ TRUE: 演算 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
viNvTmpClear	作業領域のクリア (viPos に関係なく作業領域の 31 バイト分をクリア)	BOOL		FALSE→TRUE でクリア Default (FALSE)
viPos	変数(最大 31 バイト)内のデータ開始バイト位置	INT	0 ~ 30	Default (0)
viDataType	設定データ型	INT	1 ~ 4	1: BYTE (1 byte) 2: WORD (2 bytes) 3: DWORD (4 bytes) 4: FLOAT (4 bytes) Default (0)
viByteData	BYTE 指定時の設定値	BYTE		Default (0)
viWordData	WORD 指定時の設定値	WORD		Default (0)
viDWordData	DWORD 指定時の設定値	DWORD		Default (0)
viFloatData	FLOAT 指定時の設定値	REAL		Default (0.0)
viNvUpdate	作業領域から実変数への更新(コピー)指定 (複数要素で構成された変数アクセスで重要です)	BOOL		FALSE→TRUE または 値変化で更新 Default (TRUE)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

(補足)**■値の設定と送出について**

LonWorks 変数は、複数の要素で構成されていることを考慮し、このファンクションブロックでは先ず作業領域に変数値を構築し最後に実変数へ更新します。

■設定値のエンディアンについて

LonWorks 変数値のデータの並び順は、ビッグエンディアンです。 このファンクションブロックでは、2バイト以上のデータ型を扱うために viWordData, viDWordData, viFloatData のメンバーを用意しています。

■設定値のビットについて

LonWorks 変数値でデータでビット要素が含まれるタイプがあります。 LonWorks 変数値でのビット要素はバイトの最上位ビットから割り当てられますので、本コントローラでの取り扱いは注意が必要です。

例) SNVT_reg_val (136) の要素

```
unsigned short    nr_decimals : 3;
```

この場合のデータは bit5,6,7 の3ビットとなり、nr_decimals の値1は 16#20、値7は 16#e0 となります。

例) SNVT_state (83) の要素

```
unsigned          bit0 : 1;
```

```
unsigned          bit1 : 1;
```

```
:
```

```
unsigned          bit15 : 1;
```

この場合のデータは bit0 の値 1 は 16#80、bit1 の値 1 は 16#40、bit15 の値 1 は 16#01 となります。

SNVT_switch (95) に 100:01 の設定を例に説明します。
 (Version 13 Revision 00 Janualy 2006 SNVT Master List より)

```
typedef struct {
    unsigned          value;
    signed           state;
} SNVT_switch;
```

value: Intensity as percentage of full scale, resolution 0.5%			
Field	Measurement	Field Type Category	Field Size
value	Value	Unsigned Short	1 byte
Valid Type Range	Type Resolution	Units	Invalid Value
0 .. 100	0.5	Percent of Full Scale	
Raw Range	Scale Factors	File Name	Default Value
0 .. 200 (0 .. 0xC8)	5, -1, 0 $S = a \cdot 10^b \cdot (R+c)$	N/A	N/A

state: This field can either be -1 (NULL), 0 (OFF), or 1 (ON).			
Field	Measurement	Field Type Category	Field Size
state	State	Signed Short	1 byte
Valid Type Range	Type Resolution	Units	Invalid Value
0 .. 1	1	N/A	-1 (0xFF)
Raw Range	Scale Factors	File Name	Default Value
0 .. 1	1, 0, 0 $S = a \cdot 10^b \cdot (R+c)$	N/A	N/A

Network 変数へ値を設定する方法を示します。

ここでの例は Network 変数の中でも複雑な変数(複数要素で構成される Type Category : structure)で説明します。

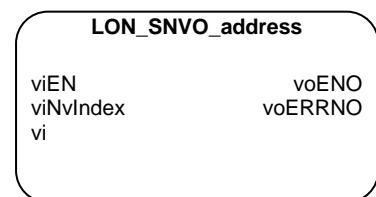
- ① 変数の作業領域のクリア準備
viNvTmpClear=FALSE
- ② 変数の作業領域のクリア
viNvTmpClear=TRUE
- ③ 変数の要素(value)の値を設定します。
viPos=0, viDataType=1(BYTE), viByteData=200
ここで viByteData に設定している値 200 は、設定する値(この例では 100:01 の 100)から Network 変数の value 部へ設定する値を上記表より求めます: $100 / 0.5$ (Type Resolution) = 200
- ④ 変数の要素(state)の値を設定します。
viPos=1, viDataType=1(BYTE), viByteData=1
ここで viByteData に設定している値 1 は、設定する値(この例では 100:01 の 01)から Network 変数の state 部へ設定する値を上記表より求めます: $1 / 1$ (Type Resolution) = 1
- ⑤ 作業領域から実変数へ更新(送出)
viNvUpdate=TRUE

追記)

- 1)⑤以降で手順③あるいは④以降を繰り返すことで変数の一部を変更して送出することができます。
- 2)値の設定のために③、④を別々のインスタンスで行うこともできます。例えば FB1 は value、FB2 は state を更新させ、実変数への更新は実行順 (FB1,FB2 の順として) の後 (FB2) のインスタンスで viNvUpdate=TRUE、それ以外(FB1)は viNvUpdate=FALSE とします。

LON_SNVO_address [FB]**LonWorks output variable 値設定****(解説)**

SNVT_address(114) 出力変数へ値を設定します。

**(INPUT)**

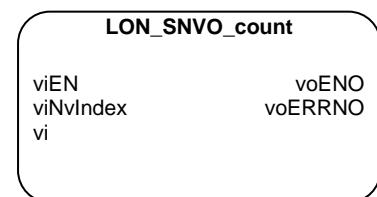
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Neuron Chip Address Resolution: 1	UINT	16,384 ~ 64,768	64,768:Invalid Value Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_count [FB]**LonWorks output variable 値設定****(解説)**

SNVT_count(8) 出力変数へ値を設定します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Event Count Resolution: 1	UINT	0 ~ 65,535	Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

LON_SNVO_count_inc [FB]**LonWorks output variable 値設定****(解説)**

SNVT_count_inc(9) 出力変数へ値を設定します。

LON_SNVO_count_inc
 viEN
 viNvIndex
 vi

 voENO
 voERRNO
(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Incremental Count Resolution: 1	INT	-32,768 ~ +32,767	Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

LON_SNVO_count_inc_f [FB]**LonWorks output variable 値設定****(解説)**

SNVT_count_inc_f(52) 出力変数へ値を設定します。

LON_SNVO_count_inc_f
 viEN
 viNvIndex
 vi

 voENO
 voERRNO
(INPUT)

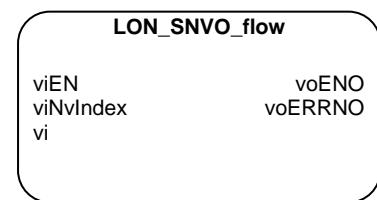
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Incremental Count Resolution: IEEE 754	REAL	float	Default(NaN)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

LON_SNVO_flow [FB]**LonWorks output variable 値設定****(解説)**

SNVT_flow(15) 出力変数へ値を設定します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Flow Volume Resolution: 1	UINT	0 ~ 65,534 or 65,535	65,535:Invalid Value Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

LON_SNVO_hvac_emerg [FB]**LonWorks output variable 値設定****(解説)**

SNVT_hvac_emerg(103) 出力変数へ値を設定します。

LON_SNVO_hvac_emerg

 viEN
 viNvIndex
 vi

 voENO
 voERRNO
(INPUT)

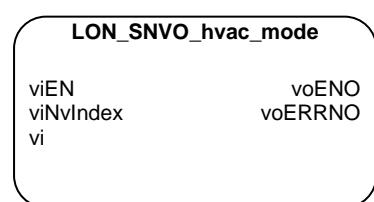
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	入力 Resolution: 1	USINT	Enumerated EMERG_xxx	255: Invalid value Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

LON_SNVO_hvac_mode [FB]**LonWorks output variable 値設定****(解説)**

SNVT_hvac_mode(108) 出力変数へ値を設定します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	入力 Resolution: 1	USINT	Enumerated HVAC_xxx	255:Invalid Value Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_hvac_status [FB]**LonWorks output variable 値設定****(解説)**

SNVT_hvac_status(112) 出力変数へ値を設定します。

LON_SNVO_hvac_status

viEN
viNvIndex
vi_xxxx

voENO
voERRNO

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi_mode	HVAC status mode Resolution: 1	USINT	Enumerated HVAC_xxx	255:Invalid Value Default(0)
vi_heat_output_primary	Primary heat output Resolution: 0.005	REAL	NaN, -163.840 ~ +163.830	自動上下限制限 NaN:Invalid Value Default(NaN)
vi_heat_output_secondary	Secondary heat output Resolution: 0.005	REAL	NaN, -163.840 ~ +163.830	自動上下限制限 NaN:Invalid Value Default(NaN)
vi_cool_output	Cooling output (primary) Resolution: 0.005	REAL	NaN, -163.840 ~ +163.830	自動上下限制限 NaN:Invalid Value Default(NaN)
vi_econ_output	Economizer output Resolution: 0.005	REAL	NaN, -163.840 ~ +163.830	自動上下限制限 NaN:Invalid Value Default(NaN)
vi_fan_output	Fan output Resolution: 0.005	REAL	NaN, -163.840 ~ +163.830	自動上下限制限 NaN:Invalid Value Default(NaN)
vi_in_alarm	Alarm state	USINT	0 / 1 / 255	0:not in an alarm, 1 to 254:in Alarm, 255:disabled alarm Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_lev_cont [FB]**LonWorks output variable 値設定****(解説)**

SNVT_lev_cont(21) 出力変数へ値を設定します。

LON_SNVO_lev_cont

viEN
viNvIndex
vivoENO
voERRNO**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Continuous Level Resolution: 0.5	REAL	0.0 ~ 100.0	自動上下限制限 Default(0.0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_lev_cont_f [FB]**LonWorks output variable 値設定****(解説)**

SNVT_lev_cont_f(55) 出力変数へ値を設定します。

LON_SNVO_lev_cont_fviEN
viNvIndex
vivoENO
voERRNO**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Continuous Level Resolution: IEEE754	REAL	0.0 ~ 100	自動上下限制限 Default(0.0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

LON_SNVO_lev_percent [FB]**LonWorks output variable 値設定****(解説)**

SNVT_lev_percent(81) 出力変数へ値を設定します。

LON_SNVO_lev_percent

viEN
viNvIndex
vi

voENO
voERRNO

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Percent level Resolution: 0.005	REAL	NaN, -163.840 ~ +163.830	自動上下限制限 NaN:Invalid value Default(NaN)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_occupancy [FB]**LonWorks output variable 値設定****(解説)**

SNVT_occupancy(109) 出力変数へ値を設定します。

LON_SNVO_occupancy

viEN
viNvIndex
vi

voENO
voERRNO

(INPUT)

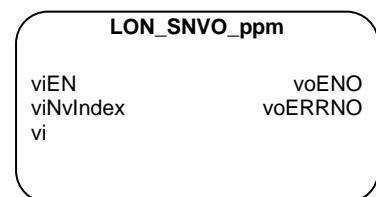
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	入力 Resolution: 1	USINT	Enumerated OC_xxx	255:Invalid value Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_ppm [FB]**LonWorks output variable 値設定****(解説)**

SNVT_ppm(29) 出力変数へ値を設定します。

**(INPUT)**

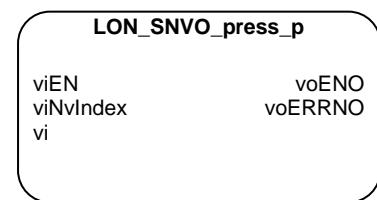
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Concentration Resolution: 1	UINT	0 ~ 65,535	Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

LON_SNVO_press_p [FB]**LonWorks output variable 値設定****(解説)**

SNVT_press_p(113) 出力変数へ値を設定します。

**(INPUT)**

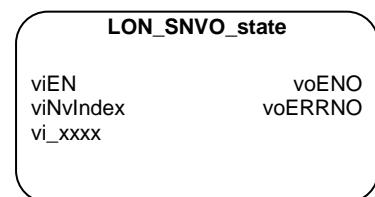
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Pressure (gauge) Resolution: 1	INT	-32,768 ~ +32,766 or 32,767	32,767:Invalid value Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

LON_SNVO_state [FB]**LonWorks output variable 値設定****(解説)**

SNVT_state(83) 出力変数へ値を設定します。

**(INPUT)**

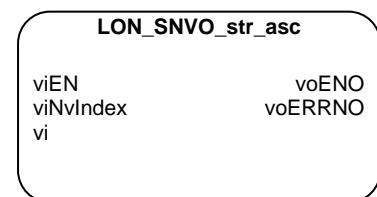
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi_bit0	bit 0	BOOL		0:FALSE, 1:TRUE Default(0)
vi_bit1	bit 1	"/"		"/"
vi_bit2	bit 2	"/"		"/"
vi_bit3	bit 3	"/"		"/"
vi_bit4	bit 4	"/"		"/"
vi_bit5	bit 5	"/"		"/"
vi_bit6	bit 6	"/"		"/"
vi_bit7	bit 7	"/"		"/"
vi_bit8	bit 8	"/"		"/"
vi_bit9	bit 9	"/"		"/"
vi_bit10	bit 10	"/"		"/"
vi_bit11	bit 11	"/"		"/"
vi_bit12	bit 12	"/"		"/"
vi_bit13	bit 13	"/"		"/"
vi_bit14	bit 14	"/"		"/"
vi_bit15	bit 15	"/"		"/"

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_str_asc [FB]**LonWorks output variable 値設定****(解説)**

SNVT_str_asc(36) 出力変数へ値を設定します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	文字列	STRING	Max 30char	先頭から 30char 有効 Default("")

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_switch [FB]**LonWorks output variable 値設定****(解説)**

SNVT_switch(95) 出力変数へ値を設定します。

LON_SNVO_switch

viEN
viNvIndex
vi_value
vi_state
vi

voENO
voERRNO

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi_value	value Resolution: 0.5	REAL	NaN, 0.0 ~ 100.0	自動上下限制限 NAN 時は vi を採用 Default (NaN)
vi_state	state	USINT	0 / 1 / 255	255:Invalid data 0,1 以外は 255 とみなす Default (255)
vi	入力	BOOL		Off:FALSE, On:TRUE Default(FALSE)

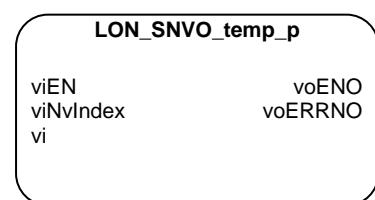
(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

vi_value	vi_state	vi	Raw Value
0.0 ~ 100.0	0	any	0:0 (0x0000)
NaN	any	FALSE or TRUE	0.0:0 (0x0000) or 100.0:1 (0xC801)
0.0 ~ 100.0	1	any	0.0:1 (0x0001) to 100.0:1 (0xC801)
0.0 ~ 100.0	255 (0,1 以外)	any	invalid (0xFFFF)

LON_SNVO_temp_p [FB]**LonWorks output variable 値設定****(解説)**

SNVT_temp_p(105) 出力変数へ値を設定します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Temperature Resolution: 0.01	REAL	NaN, -273.17 ~ +327.66	自動上下限制限 NAN:Invalid value Default(NaN)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_time_sec [FB]**LonWorks output variable 値設定****(解説)**

SNVT_time_sec(107) 出力変数へ値を設定します。

LON_SNVO_time_sec

viEN
viNvIndex
vi

voENO
voERRNO

(INPUT)

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Elapsed Time (seconds) Resolution: 0.1	REAL	NaN, 0.0 ~ 6553.4	自動上下限制限 NAN:Invalid Value Default(NaN)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0>No Error

LON_SNVO_tod_event [FB]**LonWorks output variable 値設定****(解説)**

SNVT_tod_event(128) 出力変数へ値を設定します。

LON_SNVO_tod_event
 viEN
 viNvIndex
 vi_xxxx

 voENO
 voERRNO
(INPUT)

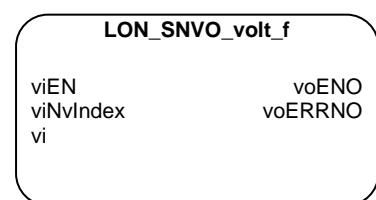
記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi_current_state	Occupancy, current Resolution: 1	USINT	Enumerated OC_xxx	255:Invalid value Default(0)
vi_next_state	Occupancy, next Resolution: 1	USINT	Enumerated OC_xxx	255:Invalid value Default(0)
vi_time_to_ next_state	Time to next state Resolution: 1	UINT	0 ~ 65,535	Default(0)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

LON_SNVO_volt_f [FB]**LonWorks output variable 値設定****(解説)**

SNVT_volt_f(66) 出力変数へ値を設定します。

**(INPUT)**

記号	解説	型	範囲	備考
viEN	Enable	BOOL		FALSE→TRUE または 値変化で送出 Default (TRUE)
viNvIndex	NV Index	INT	4 ~ 253	Default (0)
vi	Electric Voltage Resolution: IEEE 754	REAL	Floating Point	Default(NaN)

(OUTPUT)

記号	解説	型	範囲	備考
voENO	変換状態	BOOL		成功は TRUE, 失敗は FALSE
voERRNO	ERROR CODE	UINT		内部エラーコード 0:No Error

1.4. 型別ファンクションブロックを用意している SNVT

以下の型は、それぞれの型に対応した入力と出力ファンクションブロックを用意しています。

	型	Measurement	DataPoint の扱い
	Simple SNVT		
114	SNVT_address	Neuron Chip Address	16,384 to 64,767 Fixed point scalar, 2 bytes
36	SNVT_str_asc	ASCII Character string	8bit characters, up to 30 characters (zero-length string with a null terminator)
8	SNVT_count	Event Count	0 to 65,535 counts (1 count) Fixed point scalar, 2 bytes
9	SNVT_count_inc	Incremental Count	-32,768 to 32,767 (1 count) Fixed point scalar, 2 bytes
52	SNVT_count_inc_f		-1E38 to 1E38 counts Floating point scalar, 4 bytes
29	SNVT_ppm	Concentration	0 to 65,535 parts per million (1 ppm) Fixed point scalar, 2 bytes
105	SNVT_temp_p		-273.17 to +327.66 °C (0.1 °C) Fixed point scalar, 2 bytes
66	SNVT_volt_f		-1E38 to 1E38 V Floating point scalar, 4 bytes
15	SNVT_flow	Flow, l/s or ml/s (English: CFM)	0 to 65,534 liters/sec (1 l/s) Fixed point scalar, 2 bytes
21	SNVT_lev_cont	Continuous Level	0 to 100 % (0.5 %) Fixed point scalar, 1 byte
55	SNVT_lev_cont_f		0 to 100 % Floating point scalar, 4 byte
113	SNVT_press_p		-32,768 to +32,767 Pascals (1 Pa) Fixed point scalar, 2 bytes
107	SNVT_time_sec		0 to 65,535 hours (1 hour) Fixed point scalar, 2 bytes
81	SNVT_lev_percent	Level, percent	-163.84 % to 163.83 % (0.005 % or 50 ppm) Fixed point scalar, 2 bytes
	Enumerated SNVTs		
103	SNVT_hvac_emerg	Emergency Mode, HVAC	Enumerated value as follows: 0: (EMERG_NORMAL) – No emergency mode 1: (EMERG_PRESSURIZE) – Emergency pressurize mode 2: (EMERG_DEPRESSURIZE) – Emergency depressurize mode 3: (EMERG_PURGE) – Emergency purge mode 4: (EMERG_SHUTDOWN) – Emergency shutdown mode 255: (EMERG_NUL) – N/A (value at NA)
108	SNVT_hvac_mode	HVAC Mode	Enumerated values as follows: 0: (HVAC_AUTO) – Controller automatically changes mode. 1: (HVAC_HEAT) – Heating only 2: (HVAC_MRNG_WRMUP) – Application-specific morning warmup 3: (HVAC_COOL) – Cooling only 4: (HVAC_NIGHT_PURGE) – Application-specific night purge 5: (HVAC_PRE_COOL) – Application-specific pre-cool 6: (HVAC_OFF) – Controller not controlling outputs 7: (HVAC_TEST) – Equipment being tested 8: (HVAC_EMERG_HEAT) – Emergency heat mode (heat pump) 9: (HVAC_FAN_ONLY) – Fan only, air not conditioned 10: (HVAC_FREE_COOL) – Cooling with compressor not running 11: (HVAC_ICE) – Ice-making mode 255: (HVAC_NUL) – N/A (value at NA)
109	SNVT_occupancy	Occupancy	Enumerated values as follows: 0: (OC_OCCUPIED) – Area is occupied 1: (OC_UNOCCUPIED) – Area is unoccupied 2: (OC_BYPASS) – Area is temporary occupied for the bypass period 3: (OC_STANDBY) – Area is temporary unoccupied (standby) 255: (OC_NUL) – N/A (value at NA)

	Structured SNVTs																
83	SNVT_state	State	<p>Structured data (1 byte) using 16 bit-mapped fields: State0 (bit 0), State1 (bit 1), ..., State15 (bit 15)</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>1</td></tr> <tr><td>OFF</td><td>ON</td></tr> <tr><td>Inactive</td><td>Active</td></tr> <tr><td>Disabled</td><td>Enabled</td></tr> <tr><td>Low</td><td>High</td></tr> <tr><td>FALSE</td><td>TRUE</td></tr> <tr><td>Normal</td><td>Alarm</td></tr> </table> <p>No defined invalid (N/A) values exist. N/A input at nvo value is equivalent to 0 (OFF).</p>	0	1	OFF	ON	Inactive	Active	Disabled	Enabled	Low	High	FALSE	TRUE	Normal	Alarm
0	1																
OFF	ON																
Inactive	Active																
Disabled	Enabled																
Low	High																
FALSE	TRUE																
Normal	Alarm																
95	SNVT_switch	Switch	<p>Structured data (2 bytes) using 2 fields:</p> <ul style="list-style-type: none"> .Value – Intensity as percentage of full scale, 0 to 100 % (0.5 %) 255 means N/A (value at NA) .State – Binary in action, where 0 means OFF 1 means ON 255 means N/A (value at NA) 														
112	SNVT_hvac_status	HVAC Status	<p>Structured data (12 bytes) using these 7 fields:</p> <ul style="list-style-type: none"> .in_alarm – with a value of 0 or 1, where 1 means ALARM 255: N/A (value at NA) Following 5 fields have output values of -163.840 to 163.830% (0.005 %) as a percentage of full scale, where N/A = 163.835 (0x7FFF) .cool_output – Cooling Output .econ_output – Economizer Output .fan_output – Fan Output .heat_output_primary – Primary Heat Output .heat_output_secondary – Secondary Heat Output <p>.mode – Enumerated value as follows:</p> <ul style="list-style-type: none"> 0: (HVAC_AUTO) – Controller automatically changes mode. 1: (HVAC_HEAT) – Heating only 2: (HVAC_MRNG_WRMUP) – Application-specific morning warmup 3: (HVAC_COOL) – Cooling only 4: (HVAC_NIGHT_PURGE) – Application-specific night purge 5: (HVAC_PRE_COOL) – Application-specific pre-cool 6: (HVAC_OFF) – Controller not controlling outputs 7: (HVAC_TEST) – Equipment being tested 8: (HVAC_EMERG_HEAT) – Emergency heat mode (heat pump) 9: (HVAC_FAN_ONLY) – Fan only, air not conditioned 10: (HVAC_FREE_COOL) – Cooling with compressor not running 11: (HVAC_ICE) – Ice-making mode 255: (HVAC_NUL) – N/A (value at NA) 														
128	SNVT_tod_event	Supervisory Schedule Command	<p>Structured data (4 bytes) using 3 fields:</p> <ul style="list-style-type: none"> .current_state – Current scheduled occupancy state Enumerated as follows: <ul style="list-style-type: none"> 0: (OC_OCCUPIED) – Area is occupied 1: (OC_UNOCCUPIED) – Area is unoccupied 2: (OC_BYPASS) – Area is temporarily occupied for the bypass period 3: (OC_STANDBY) – Area is temporarily unoccupied (standby) 255: (OC_NUL) – N/A (value at NA) .next_state – next scheduled occupancy state Enumerated same as .current_state .time_to_next_state – Minutes until next schedule event change, 0 to 65,534 minutes (1 min.), 65.535 (0xFFFF) is NA 														

1.5. 型別ファンクションブロックが用意されていない Network Variable Type の扱い

全てのSNVTおよびUNVTは、汎用ファンクションブロック(LON_NVO_general, LON_NVI_general)を使用して入出力ができます。

■型別ファンクションブロックを使用する意味

それぞれの変数型に対応したファンクションブロックが用意されている場合は、そのファンクションブロックを使用することで変数型の持つ各要素に同時に、そして各要素の値は変換された値でアクセスできます。

■汎用ファンクションブロックを使用する意味

全ての変数型がアクセスできます。その代わりに、各要素の同時性と値の変換をユーザにて管理し値はSNVT Master Listに従い演算する必要があります。