

AJAN
システム監視コマンド
リファレンス

目 次

第 1 章	はじめに	4
1.1	概要	4
1.2	用語説明	5
第 2 章	機能説明	7
2.1	WDT	7
2.2	温度監視	8
2.3	電源電圧監視	9
2.4	RAS 出力	10
2.5	RAS 通知	10
2.6	デジタル入出力	11
2.7	割り込みイベント	12
2.8	瞬低対策用電源監視	13
2.9	LED 制御	14
2.10	ブザー制御	16
2.11	ひもスイッチ	17
2.12	FAN 監視	18
2.13	振動・衝撃監視	19
2.14	ストレージ監視	19
2.15	スロットイン電源モジュール監視	20
2.16	自己診断	20
2.17	遷移出力	21
2.18	製品情報取得	23
第 3 章	リファレンス	24
3.1	コマンド一覧	24
3.2	詳細個別説明 (WDT)	26
3.3	詳細個別説明 (温度監視)	28
3.4	詳細個別説明 (電源電圧監視)	32
3.5	詳細個別説明 (RAS 出力)	35
3.6	詳細個別説明 (RAS 通知)	37
3.7	詳細個別説明 (デジタル入出力)	39
3.8	詳細個別説明 (割り込みイベント)	42
3.9	詳細個別説明 (瞬低対策用電源監視)	47
3.10	詳細個別説明 (LED 制御)	50
3.11	詳細個別説明 (ブザー制御)	53
3.12	詳細個別説明 (ひもスイッチ)	54
3.13	詳細個別説明 (自己診断)	55
3.14	詳細個別説明 (FAN 監視)	57
3.15	詳細個別説明 (振動・衝撃監視)	59
3.16	詳細個別説明 (外部コネクタ出力)	61

3.17	詳細個別説明（ストレージ監視）	62
3.18	詳細個別説明（遷移出力）	63
3.19	詳細個別説明（スロットイン電源モジュール監視）	65
3.20	詳細個別説明（製品情報）	66
3.21	エラーコード一覧	67
第4章	サンプルプログラム	68

4.1	サンプルプログラム一覧	68
-----	-------------------	----

第5章	付録	70
------------	-----------------	-----------

5.1	システム監視ソフトウェア	70
5.2	機能対応表	71

第6章	索引	72
------------	-----------------	-----------

第7章	重要な情報	74
------------	--------------------	-----------

第1章 はじめに

本ドキュメントは、AJANのシステム監視制御コマンドの説明を記載しております。

本ドキュメントでは、説明で表現している表記として下記のように定義します。

- ・ コマンドの書式の説明において、[]内の引数は省略できます。
- ・ 文字の大小について
コマンドは大文字/小文字のどちらでも動作します。
変数名は大文字/小文字も同じものとして扱われます。
ファイルパス/ファイル名は大文字/小文字で区別されます。



本ドキュメント記載の、AJANはIoT用プログラミング言語です。
Interface Linux System上でのみ動作可能です。

1.1 概要

システム監視制御コマンドは、弊社産業用コンピュータを監視・制御する為のコマンドです。
システムの信頼性を高める機能を色々提供しています。

本コマンドは、C言語用ライブラリである「システム監視ソフトウェア」を利用することで実現しています。機能によっては、「システム監視ソフトウェア」のマニュアルに、より詳しく記載されていますので、更に説明が必要な場合は、そちらも併せてご参照ください。

又、コンピュータによって利用できる機能は異なります。
お使いの製品で、システム監視のどの機能が利用できるかは、「機能対応表」に記載がありますので、こちらをご参照ください。

これらの資料の参照方法は、『第5章 付録』に記載しております。

1.2 用語説明

本書で使用する用語について説明します。

No	用語	説明			
1	WDT	Watchdog Timer システムやプログラムの死活監視を行う機能。 タイマを定期的にクリアすることで正常と判断します。 システムのハングアップなどでクリアが行われなかった場合に異常と判断します。			
	CPU リセット	CPU のリセットによりシステムを再起動させる機能です。			
	S ステート (S0, S5, G3)	システムの状態を表しています。			
			システム	電源 LED	電源供給
		S0	稼動	点灯	あり
		S5	停止	点滅	あり
		G3	停止	消灯	なし
2	S. M. A. R. T	Self-Monitoring, Analysis And Reporting Technology HDD や SSD の自己診断機能。			
3	RAID	Redundant Arrays of Inexpensive Disk ストレージの冗長化等のために、複数のドライブをひとつのドライブとして扱う機能です。			
	スキップ・リコン	RAID ボードによるデータ再構築(コピー)中に、コピー元のドライブに読み込み異常が発生した状態のことです。RAID ボードはその異常が発生した部分をスキップしてコピー処理を継続します。			
	システム・ホルト	System fault 弊社製品に搭載されているハードウェア RAID ボードに致命的なエラーが発生し、RAID システムとして機能していない状態を表します。			
4	デジタル入出力	デジタル入力とリレー接点(1a)出力を備えた端子。			
5	外部コネクタ	電源 ON/OFF 入力と RAS 出力を備えた端子。			
6	ひもスイッチ	SuperCD に搭載されているプルスイッチ用の端子。			
7	瞬低対策電源装置	OS シャットダウンまでの時間を保証するものではなく、連続稼動中に瞬時電圧低下(瞬低)が発生する様な不安定な電源環境でも連続動作させるための装置です。			
	瞬低	「瞬時電圧低下」の略語です。 送電線への落雷などにより発生する、瞬間的な電圧低下を表します。			
8	RAS	Reliability, Availability and Serviceability の略語です。 弊社製品ではシステムの監視を行い、異常検出時に通知を行うことができます。			
9	遷移出力	システムの停止時と起動時にデジタル出力や外部コネクタ出力を制御する機能。			
		DAW	Disable After WDT 遷移出力を無効にする機能です。 次の条件を満たした場合に動作します。		
		機能	条件		
		遷移出力	DAW を有効に設定している。		
		RAS 通知	WDT 満了の通知を行う制御対象が遷移出力と同じである		
		WDT	WDT を動作させている。 WDT 満了時に CPU リセットを行う設定である。		
10	ヒステリシス	値が閾値を超えて異常となったとき、値が閾値を下回ってから正常とみなすまでの範囲。			
11	デジタルフィルタ	電源電圧異常検出時に、瞬間的な異常をフィルタするための機能を表します。 本ソフトウェアではフィルタの値として時間(ms)が設定できます。 設定した時間、異常が継続した場合に異常と判定します。			

システム監視制御コマンドリファレンス

No	用語	説明
12	警報ランプ	弊社製 SuperCD の筐体上部の LED。 本文書では、以下「エラーLED」として扱います。
13	拡張スロット 周辺温度	製品の拡張スロット周辺に搭載されているセンサが取得する温度。 センサの位置は製品によって異なります。詳しくは、製品の USER'S MANUAL をご確認ください。
14	機能対応表	システム監視ソフトウェアで利用できる機能・関数・識別子などを製 品ごとにまとめた文書です。 システム監視ソフトウェアに func_table.pdf のファイル名で付属し ています。 また、最新版を当社 Web site よりダウンロードすることができます。

第2章 機能説明



紹介している各機能は、サポートしているハードウェアのみ使用可能です。
機能の有無については、ハードウェアのマニュアルおよび機能対応表(「5.2 機能対応表 (P.71)」参照)をご確認ください。

2.1 WDT

下記の流れでWDTを使用します。

- 1.WDTの周期、周期満了アクション(CPUリセット、または割り込みイベント通知)を設定。
- 2.WDT開始。
- 3.設定した周期に達する前にWDTクリア。

WDTの周期、周期満了アクションの設定は、WDTCONFIG命令で行います。

```
' 50 秒間クリアされなければ CPU リセットを行う
WDTCONFIG 50, "ACTION=RESET"
```

WDTの開始は、WDTSTART命令で行います。

```
WDTSTART
```

WDTのクリアは、WDTCLEAR命令で行います。

```
WDTCLEAR
```

WDTの使用を停止する場合は、WDTSTOP命令を実行してください。

```
WDTSTOP
```

WDT周期満了時、CPUリセットの他に、割り込みイベントを通知することができます。
詳細は『2.7 割り込みイベント』を参照してください。

WDTの割り込みイベントとCPUリセットの用途は下記を参考にしてください。

WDT処理	用途
割り込みイベント	アプリケーションの動作異常によるプロセス停止検出を目的としています。 アプリケーションとは別に監視プロセスを作成し、アプリケーションでWDTの周期的クリアを実行します。監視プロセスにはWDT割り込みイベントを登録しておきます。 もしアプリケーションに異常が生じてWDTクリアが途絶えた場合、監視プロセスに割り込みイベントが発生し、アプリケーションの異常を検知できます。
CPUリセット	システム(OS)のフリーズ等からの自動復旧を目的としています。

2.2 温度監視

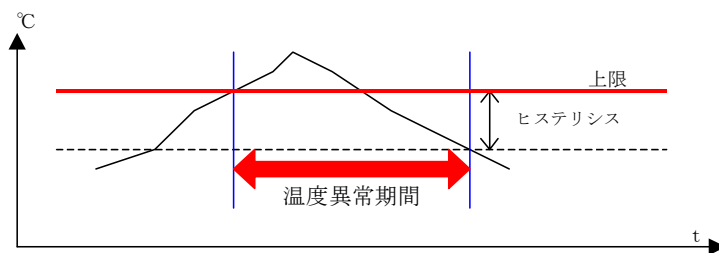
CPUコア温度と、CPU基板温度、メモリ周囲温度、拡張スロット周辺温度を取得することができます。いずれもTEMPGET関数を使用します。取得値は負数になることもあります。単位は℃になります。

```
' CPUコア温度の取得
CPUTEMP = TEMPGET(1)
' CPU基板温度の取得
BRDTEMP = TEMPGET(2)
' メモリ周囲温度の取得
MEMTEMP = TEMPGET(3)
' 拡張スロット周辺温度の取得
SLOT1 = TEMPGET(4)
SLOT2 = TEMPGET(5)
SLOT3 = TEMPGET(6)
```

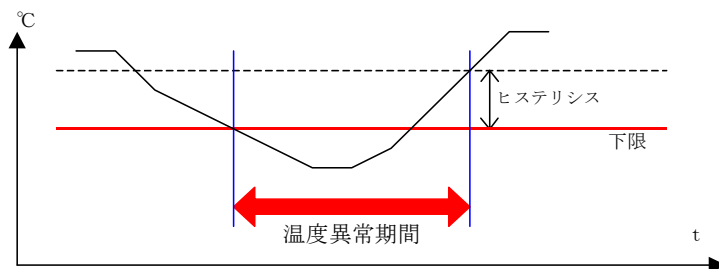
温度の上限と下限を設定することで、その範囲を超えた場合に、割り込みイベントを通知することができます。詳細は『2.7 割り込みイベント』を参照してください。

温度上限、下限異常の発生から復帰までの範囲（ヒステリシス）を設定することができます。設定した値は次回起動時も保持されます。

温度上限異常の場合は以下の図のようになります。



温度下限異常の場合は以下の図のようになります。



閾値の設定をするにはTEMPLIMIT命令を使用します。

```
' CPUコア温度の温度下限(60)、上限(80)異常検出の閾値とヒステリシス(5)を設定
TEMPLIMIT 1, 60, 80, 5, 5
```

温度警告検知をサポートしている製品では温度警告上限値、下限値、ヒステリシスを設定できます。設定は、TEMPUPPERLIMIT命令、TEMPLOWERLIMIT命令を使用します。

```
' CPUコア温度の温度上限警告、上限異常の閾値とヒステリシスを設定
TEMPUPPERLIMIT 1, 60, 80, 5, 5
' CPUコア温度の温度下限警告、下限異常の閾値とヒステリシスを設定
TEMPLOWERLIMIT 1, 10, 0, 5, 5
```


2.3 電源電圧監視

電源電圧を取得することができます。

POWERGET関数を使用します。

```
' 電源電圧の取得
POWER = POWERGET(1)
```

電源の電圧が異常を示した際に割り込みイベントを通知することができます。

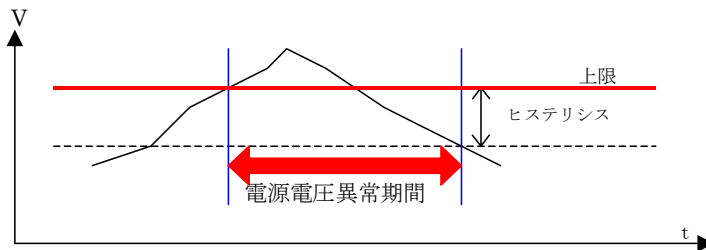
詳細は『2.7 割り込みイベント』を参照してください。

電源電圧上限、下限異常の発生から復帰までの範囲（ヒステリシス）を設定することができます。

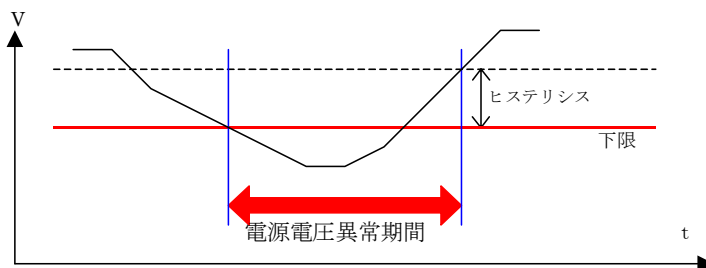
設定した値は次回起動時も保持されます。

※ ヒステリシスの変更はPOWERLIMIT命令から変更を行います。

電源電圧上限異常の場合は以下の図のようになります。



電源電圧下限異常の場合は以下の図のようになります。



閾値の設定をするにはPOWERLIMIT命令を使用します。

```
' 電源電圧上限、下限異常検出の閾値とヒステリシスを設定
POWERLIMIT 1, 25, 170, 10, 10
```

デジタルフィルタ値を設定できます。設定した期間異常が継続した場合に異常を検知します。

設定はDIGITALFILTER命令で行います。

```
' デジタルフィルタ値を 5 ミリ秒に設定
DIGITALFILTER 1, 5
```

2.4 RAS出力

異常検出時に、RAS出力を行うことができます。

あらかじめRAS出力を行う異常検知の種類と、異常時に出力する端子状態を設定しておくことで、異常検出時に自動的にRAS出力を行います。

```
' RAS出力をリレーOFF状態で初期化
ROOUTPUT "RASOUT=OFF"
' WDTタイマとCPU温度上限越え時にRAS出力をリレーON状態にするように設定
ROCONFIG &H3, "RASOUT=ON"
```

ROOUTPUT命令により、アプリケーションで自由にRAS出力を制御することも可能です。

RAS出力の状態をROSTATUS関数で取得することができます。

```
' RAS出力状態を取得
STAT = ROSTATUS
```

システムの電源投入時、RAS出力はリレーOFFになります。

RAS出力条件を設定した場合、次回起動時もその設定を保持します。

2.5 RAS通知

異常検出時に、RAS出力(デジタル出力、外部コネクタ出力)、ブザー、LEDにより通知を行うことができます。

あらかじめRAS通知を行う異常検知の種類と通知方法を設定しておくことで、異常検出時に自動的に通知を行います。

設定は、RASNOTIFY命令で行います。

```
' デジタル出力接点 1～4 を OFF 状態で初期化
OUTPUTUP([ 1 to 4 ]) = [0;0;0;0]
' WDT タイマ異常時にデジタル出力接点 1 を ON にするよう設定
RASNOTIFY "ENABLE=DOUT1", &H1, "ENABLE=ON"
' CPU 基板温度異常時にデジタル出力接点 2 を ON にするよう設定
RASNOTIFY "ENABLE=DOUT2", &H3, "ENABLE=ON"
' 電源電圧異常時にブザーを ON にするよう設定
RASNOTIFY "ENABLE=BUZZER", &H40
```

OUTPUTUP命令により、アプリケーションで自由にデジタル出力を制御することも可能です。

詳細は『2.6 デジタル入出力』を参照してください。

RASNOTIFY命令でRAS通知方法及び条件を設定した場合、次回起動時もその設定を保持します。

2.6 デジタル入出力

デジタル入出力制御を行うことができます。デジタル出力はRAS出力と兼用となります。
デジタル出力を行うにはOUTPUTUP命令を使用します。

```
' デジタル出力接点 1～4 を ON
OUTPUTUP([ 1 to 4 ]) = [1;1;1;1]
' デジタル出力接点 1～4 を OFF
OUTPUTUP([ 1 to 4 ]) = [0;0;0;0]
' デジタル出力接点 1 を ON
OUTPUTUP(1)= 1
' デジタル出力接点 1 を OFF
OUTPUTUP(1)= 0
```

INPUTUP関数でデジタル入力接点の状態を確認できます。

```
' デジタル入力端子 1～4 の状態を取得
Data = INPUTUP( [1 to 4] )
' デジタル入力端子 1 の状態を取得
Data1 = INPUTUP(1)
' デジタル入力端子 2 の状態を取得
Data2= INPUTUP(2)
```

デジタル入力された際に割り込みイベントを通知することができます。
詳細は『2.7 割り込みイベント』を参照してください。

デジタル入力1(DIN1)を電源スイッチに設定することができます。
(サポートしているハードウェアのみ使用可能です。機能の有無はコンピュータのマニュアルをご確認ください。)

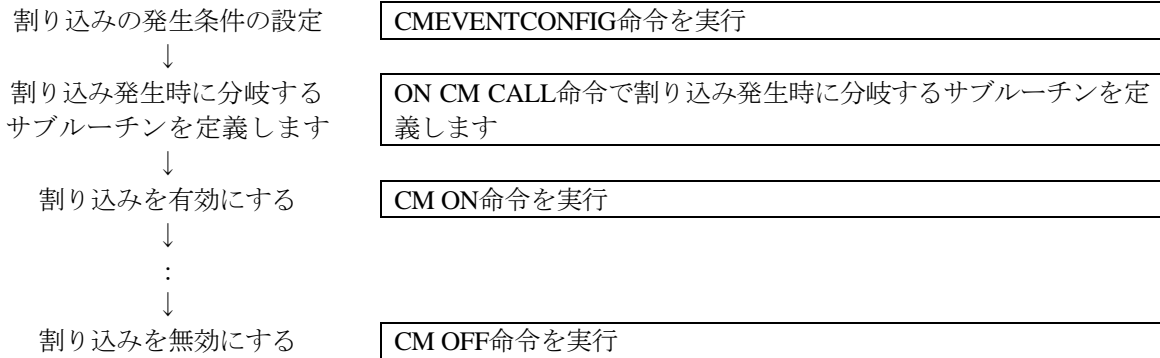
設定はUPCONFIG命令で行います。

```
' デジタル入力 1 を電源 ON/OFF スイッチに設定
UPCONFIG "DI1=PSWITCH"
```

デジタル入力1を電源スイッチに設定してもデジタル入力2～4は汎用入力として使用可能です。

2.7 割り込みイベント

割り込みイベントは下記の流れで使用します。



※「INCLUDE "SML001.AJN"」でSML001.AJNをインクルードする必要があります。

※発生した割り込み要因を取得する場合は、CMEVENT関数にて取得します。

イベント発生時にコールバック関数が呼び出される設定例を下記に示します。

(WDT、CPUコア温度上限、CPUコア温度下限、電源電圧異常を有効にします。)

```
' SML001.AJN をインクルード
INCLUDE "SML001.AJN"
' CPU コア温度下限(0℃)、上限(50℃)設定
TEMPLIMIT 0, 0, 50, 0, 0
' 割り込みの発生条件の設定(WDT、CPU コア温度上限・下限、電源電圧設定)
CMEVENTCONFIG "WDT=ON, CPTHS=ON, CPTLS=ON, VEXTA=ON"
' 割り込み発生時に分岐するサブルーチンを定義します
ON CM CALL SMCALLBACK
' 割り込みを有効にする
CM ON
...
```

割り込みイベントの使用終了時は、割り込みマスク設定でイベントを無効化し、CM OFF命令で割り込みイベントを無効にしてください。

```
' 全割り込みを無効化
CMEVENTCONFIG "WDT=OFF, CPTHS=OFF, CPTLS=OFF, VEXTA=OFF"
' 割り込みを無効にする
CM OFF
```

割り込みイベントが発生した場合に呼び出される処理は、以下の記述となります。

```
SUB SMCALLBACK()
' イベント要因取得
EVT = CMEVENT
IF EVT = CMEVTWDS THEN
    PRINT "WDT Event occur!"
END IF
IF EVT = CMEVTCPTHS THEN
    PRINT "CPTHS Event occur!"
END IF
IF EVT = CMEVTCPTLS THEN
    PRINT "CPTLS Event occur!"
END IF
IF EVT = CMEVTVEXTA THEN
    PRINT "VEXTA Event occur!"
END IF
END SUB
```

2.8 瞬低対策用電源監視

瞬低対策装置を接続すると、バッテリーステータスの取得および、停電が起きた際の動作を設定できます (バッテリーモジュール接続時のみ使用可能です)。

BATPOWEROFFTIME命令で、停電発生からシャットダウン開始までの時間と、シャットダウン開始から電源強制断までの時間を秒単位で設定できます。

停電発生～シャットダウン開始 16秒

シャットダウン開始～電源強制断 180秒に設定する場合

BATPOWEROFFTIME 16, 180

システムの電源投入時、停電発生～シャットダウン開始までの時間は16秒、シャットダウン開始～電源強制断までの時間は180秒です。

BATSTATUS関数でバッテリーステータスを取得できます。

' バッテリーステータスの取得 STATUS= BATSTATUS

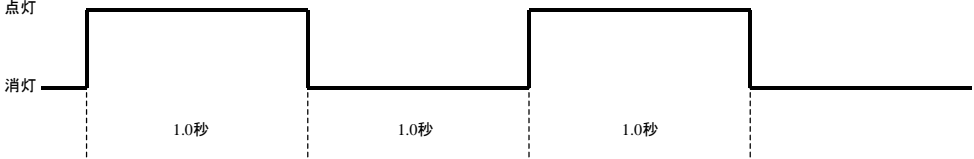
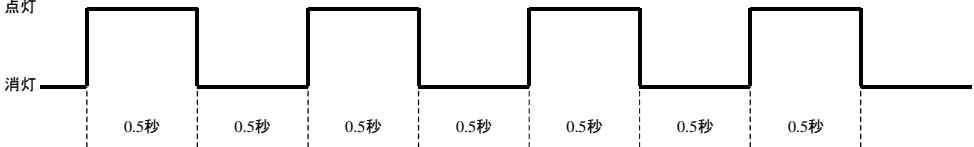
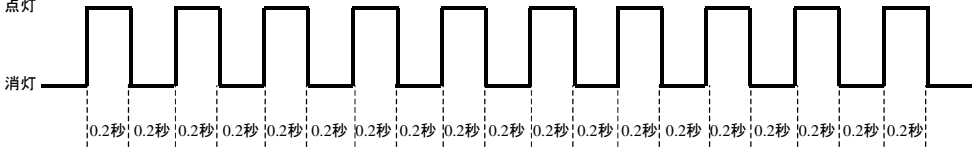
2.9 LED制御

2.9.1 エラー/アラームLED

エラー/アラームLEDの点灯パターンを設定することができます。
設定は、LEDPATTERN命令で行います。

```
' エラーLED の点灯パターンを設定
LEDPATTERN "ERRLED=PATTERN1"
```

点灯パターンは以下を設定可能です。

パターン	LED の動作
1	常時点灯します。
2	1 秒ごとに点灯と消灯を繰り返します。 
3	0.5 秒ごとに点灯と消灯を繰り返します。 
4	0.2 秒ごとに点灯と消灯を繰り返します。 

多色設定機能を持つ製品では、エラーLEDの発光色を設定することができます。
設定は、LEDPATTERN命令で行います。

```
' エラーLED の点灯パターンを 1 に、発光色を GREEN に設定
LEDPATTERN "ERRLED=PATTERN1,ERRLEDCOLOR=GREEN"
```



多色設定機能は有償オプションです。ご利用されたい方は、多色設定機能付きの製品をお買い求めください。

エラーLEDおよびアラームLEDを、単に常時点灯 / 消灯するだけならば、CMLED命令で簡単に制御できます。

```
' エラーLED を常時点灯し、発光色を GREEN に設定(多色機能を持たない製品では単に点灯)
CMLED 2
' エラーLED を消灯し、アラーム LED を点灯する
CMLED 0, 1
```

あらかじめLEDを点灯制御させる条件を設定しておくことで、異常検知時に自動的にLEDを制御させることができます。

設定は、RASNOTIFY命令で行います。

```
' あらかじめ LED を消灯
LEDCONTROL "ERRLED=OFF"
' WDT タイマ異常時に LED を点灯させるように設定
RASNOTIFY "ENABLE=ERRLED", &H1
```

LEDCONTROL命令により、アプリケーションで自由にLEDを制御することも可能です。

異常検知時に自動でLEDが制御開始した場合、LEDCONTROL命令またはCMLED命令でLEDを停止しても、異常状態が続いていれば再度制御します。

異常が続いている状態でLEDを停止させたい場合は、RASNOTIFY命令で一旦自動LED制御を無効化してからLEDCONTROL命令またはCMLED命令でLEDを停止してください。

また、異常復旧後もLEDは自動で停止しません。停止させたい場合、LEDCONTROL命令、またはCMLED命令で停止させてください。

2.9.2 汎用LED

LEDCONTROL命令で、自由に汎用LEDを制御する事ができます。

また、あらかじめ条件を設定しておくことで、異常検知時に自動的にLEDを制御させることができます。

設定は、RASNOTIFY命令で行います。

```
' あらかじめ LED を消灯
LEDCONTROL "LED1=OFF"

' WDT タイマ異常時に LED を点灯させるように設定
RASNOTIFY "ENABLE=LED1", &H1
```

CMLED命令も同じように汎用LEDを制御する事ができます。

```
' LEDCONTROL で、LED1、LED3、LED5 を点灯する
LEDCONTROL "LED1=ON, LED3=ON, LED5=ON"
' または、CMLED で LED1、LED3、LED5 を点灯する(エラーLED、アラーム LED は消灯)
CMLED 0, 0, &H1 or &H4 or &H10
```

異常検知時に自動でLEDが制御開始した場合、LEDCONTROL命令またはCMLED命令でLEDを停止しても、異常状態が続いていれば再度制御します。

異常が続いている状態でLEDを停止させたい場合は、RASNOTIFY命令で一旦自動LED制御を無効化してからLEDCONTROL命令またはCMLED命令でLEDを停止してください。

また、異常復旧後もLEDは自動で停止しません。停止させたい場合、異常発生原因をよくご確認の上、LEDCONTROL命令、またはCMLED命令で停止させてください。

2.10 ブザー制御

BUZZERCONTROL命令またはCMBUZZER命令により、自由にブザーを制御することができます。
また、あらかじめブザーをONさせる条件を設定しておくことで、異常検知時に自動的にブザーをONさせることができます。

設定は、RASNOTIFY命令で行います。

```
' あらかじめブザーを OFF
BUZZERCONTROL "CONTROL=OFF"
' または
CMBUZZER FALSE

' WDT タイマ異常時にブザーが ON するように設定
RASNOTIFY "ENABLE=BUZZER", &H1
```

異常検知時に自動でブザーがONした場合、BUZZERCONTROL命令またはCMBUZZER命令でブザーをOFFしても、異常状態が続いていれば再度ブザーがONします。

異常が続いている状態でブザーをOFFしたい場合は、RASNOTIFY命令で一旦ブザーを無効化してからBUZZERCONTROL命令またはCMBUZZER命令でブザーをOFFしてください。

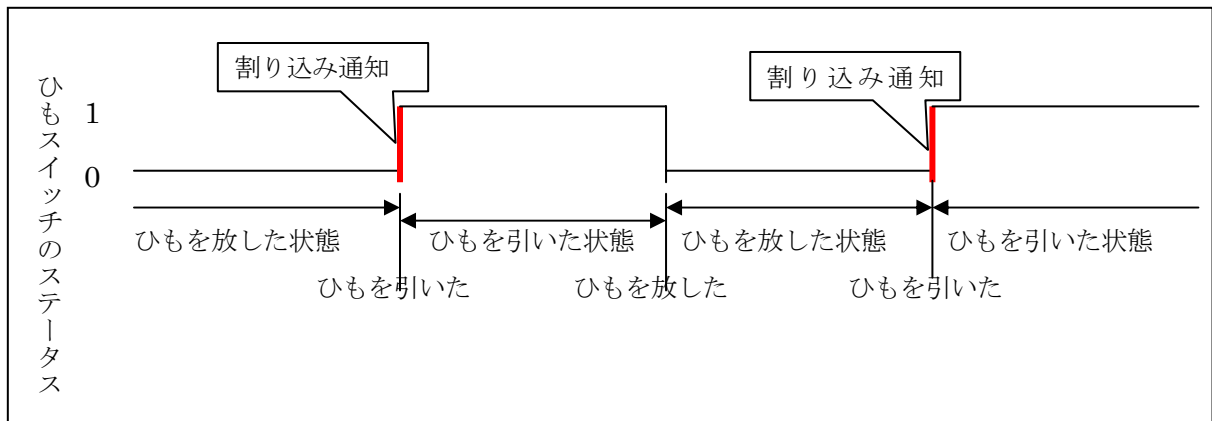
また、異常復旧後もブザーは自動でOFFしません。停止させたい場合、BUZZERCONTROL命令、またはCMBUZZER命令で停止させてください。

2.11 ひもスイッチ

ひもスイッチを引いた際に割り込み通知が行えます。
 詳細は『2.7 割り込みイベント』を参照してください。
 ひもスイッチによる割り込み通知は、ひもを引いた瞬間に行われます。

ひもスイッチは、現在スイッチを引かれているか放されているかのステータスを取得することができます。
 ステータスはHIMOSWITCHSTATUS関数で確認でき、引かれている場合は1、放されている場合は0のステータスが取得できます。

割り込み通知は以下のタイミングで発生します。



ステータスのポーリングによる使用例を下記に示します。

```
' ひもスイッチが引かれている場合、ブザーを鳴らす例

' あらかじめブザーをオフ
BUZZERCONTROL "CONTROL=OFF"
' ひもスイッチの状態をポーリング
DO WHILE TRUE
  ' ひもスイッチの状態を取得
  STATUS = HIMOSWITCHSTATUS
  IF STATUS = 1 THEN
    ' ひもスイッチが引かれているならブザーオン
    BUZZERCONTROL "CONTROL=ON"
  ELSE
    ' ひもスイッチが放されているならブザーオフ
    BUZZERCONTROL "CONTROL=OFF"
  END IF
LOOP
END
```

ひもスイッチは設定を行うことで電源スイッチとして利用することもできます。
 設定はHIMOSWITCHCONFIGコマンドで行います。

```
' ひもスイッチを電源スイッチに設定
HIMOSWITCHCONFIG "MODE=PSWITCH"
```

設定は次の起動後から有効になります。

2.12 FAN監視

FANの回転数を取得することができます。

FANROTATEGET関数を使用します。

```
' FAN1 の回転数を取得  
ROTATE = FANROTATEGET(1)
```

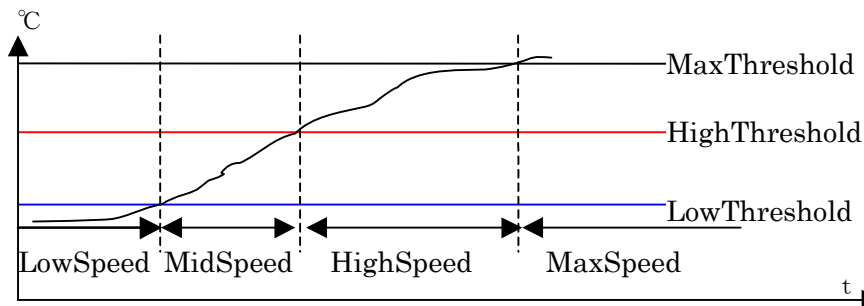
回転数は毎分の回転数(rpm)で取得できます。

回転数が閾値を下回った場合に割り込みイベントを通知することができます。

詳細は『2.7 割り込みイベント』を参照してください。

FANの回転数はCPUコア温度に応じて自動で制御されます。

以下の図のように変化します。



自動制御の温度閾値(LowThreshold, HighThreshold)と回転速度(LowSpeed, MidSpeed, HighSpeed)を設定できます。

※MaxThresholdとMaxSpeedは固定値となります。

設定はFANAUTOTHRESHOLD命令とFANAUTOSPEED命令で行います。

```
' FAN自動制御の閾値を設定  
FANAUTOTHRESHOLD 1, 20, 70  
' FAN自動制御の速度を設定  
FANAUTOSPEED 1, 30, 50, 80
```

※MaxThreshold、HighThreshold、LowThresholdには、-3℃～-5℃程度のヒステリシス(固定値)が設定されています。

CPUコア温度がHighThresholdを超えてFANの回転数がHighSpeedとなった場合、HighThresholdよりも3℃～5℃程度下回れば十分冷却されたとみなしてFANの回転数はMidSpeedとなります。

2.13 振動・衝撃監視



本機能は有償オプションです。ご利用されたい方は、加速度センサの機能付きの製品をお買い求めください。

製品に搭載されている加速度センサから重力加速度を取得できます。

GSENSEGET関数を使用します。

```
'X軸方向の重力加速度値を取得します。
Data = GSENSEGET("TARGET=SENSOR1, AXIS=X")
```

ここで得られる値は加速度センサの生の値となります。重力加速度値へ変換するには下記の式で計算を行います。

$$G = \frac{(x \times 0.01289) - 1.5}{0.42}$$

ここで、

G: 重力加速度値

x: 取得値

1秒間の加速度の最大値と最小値の差が閾値を超えた場合に割り込みイベントを通知することができます。詳細は『2.7 割り込みイベント』を参照してください。

閾値及びヒステリシスを1(0.03G)刻みで各方向ごとに指定することができます。設定には、GSENSELIMIT命令を使用します。

```
'X軸方向の閾値とヒステリシスの設定
GSENSELIMIT "TARGET=SENSOR1, AXIS=X", 10, 5
```

2.14 ストレージ監視

RAID状態の取得ができます。

RAIDSTATUS関数を使用します。

```
'RAID状態を取得します。
Data = RAIDSTATUS(0)
```

引数に「0」(DPG-0226使用時)を指定した場合、RAIDボードのステータスの取得が行えます。取得できるステータスは以下の通りです。

値	内容
0	通常動作
1	ドライブ1 異常
2	ドライブ2 異常
3	データ再構築（コピー）中
4	システム・ホルト
5	HDD2→HDD1へのデータ再構築（コピー）中にスキップ・リコン発生
6	HDD1→HDD2へのデータ再構築（コピー）中にスキップ・リコン発生

RAID異常(上記表の値1、2、4、5、6いずれか)の発生時に割り込み通知を行うことができます。詳細は『2.7 割り込みイベント』を参照してください。

2.15 スロットイン電源モジュール監視

スロットイン電源モジュールのIDを取得することができます。
CMPSMGET関数を使用します。

```
' スロット 1 のIDの取得  
ID = CMPSMGET("ID", 1)
```

スロットイン電源モジュールのスロット位置は、製品のパネル面から見て、左側からP1, P2, …です。

また、スロットイン電源モジュールの状態も取得することができます。
CMPSMGET関数を使用します。

```
' スロット 1 の電源状態の取得  
ID = CMPSMGET("STATUS", 1)
```

電源状態が警告 / 異常の際、割り込みイベントを通知することができます。
詳細は『2.7 割り込みイベント』を参照してください。

2.16 自己診断

システムの稼動状態を診断し、現在異常が発生していないか確認が行えます。
RASERRORSTATUS関数を使用します。

```
' 異常状態ステータスを取得。  
Data = RASERRORSTATUS
```

※温度や電源電圧などの各種異常は、コマンドで設定した閾値に依存します。設定されている閾値が適切でない場合、正しい自己診断結果とならない可能性があります。

2.17 遷移出力

遷移出力は、システム起動時と停止時に、デジタル出力/外部コネクタの出力状態を制御する機能です。遷移出力を使用するためには、予め設定を行う必要があります。



本機能の詳細については、システム監視ソフトウェア(DPG-0226)の「遷移出力」の章を参照ください。

遷移出力には以下の機能があります。

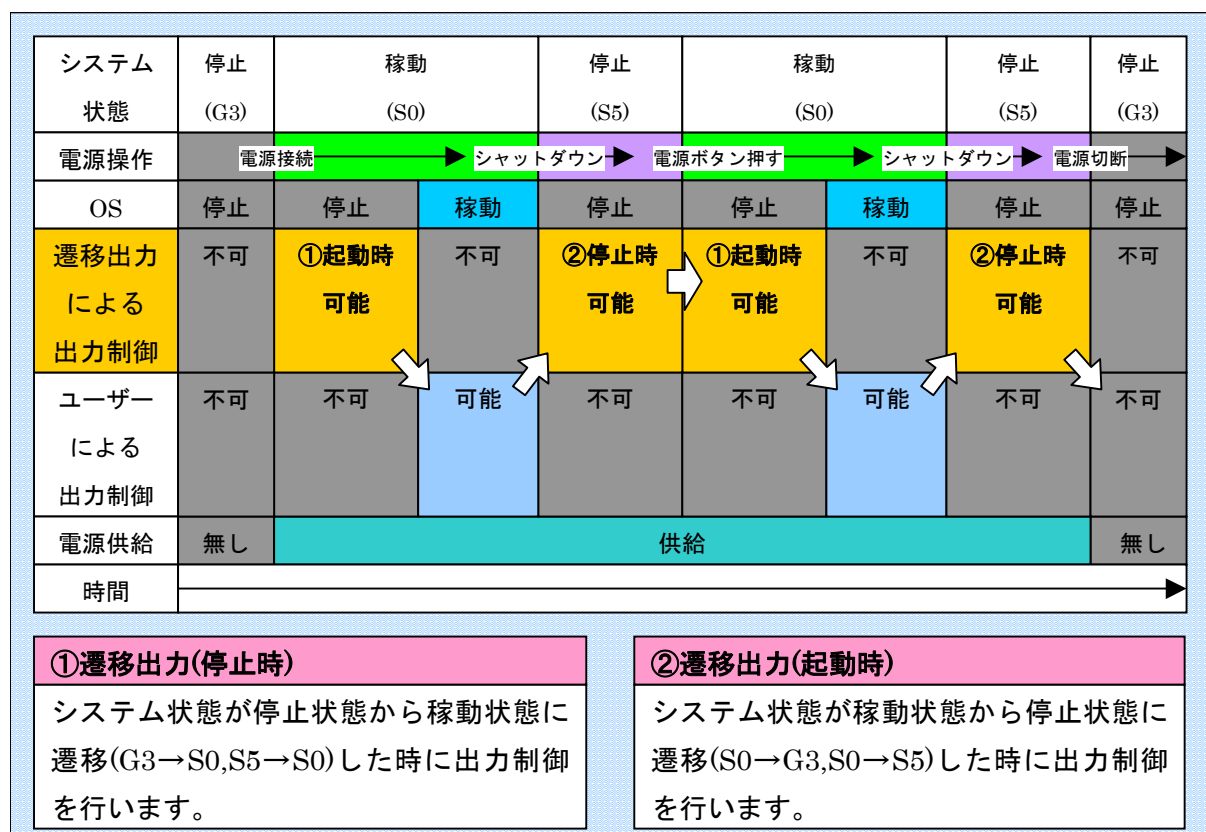
機能	説明
基本機能	システムの起動時と停止時に、設定に従って自動で出力状態を制御します。
DAW機能	WDT満了時にRAS通知かつCPUリセットが起きた時、再起動中の遷移出力を無効にします。

遷移出力は、以下のような時に使えます。

- ・電源をONにした直後やOSの稼動前に、出力状態を変更したい。
- ・シャットダウン後に、出力状態を変更したい。
- ・システム稼動の開始や停止を外部へ通知したい。

2.17.1 基本機能

制御対象の出力状態は、システムの起動時と停止時に設定に従って自動で制御されます。遷移出力の動作説明を下図に示します。



- ・BIOSの設定でState After G3を「S5 State」に設定した場合、外部電源接続による起動時の遷移出力は動作しません。
- ・外部電源が接続されていない停止(G3)状態では、出力状態はすべてOFFになります。

2.17.2 DAW(Disable After WDT)

WDT満了時にRAS通知かつCPUリセットが起きた時、再起動中の遷移出力を無効にする機能です。

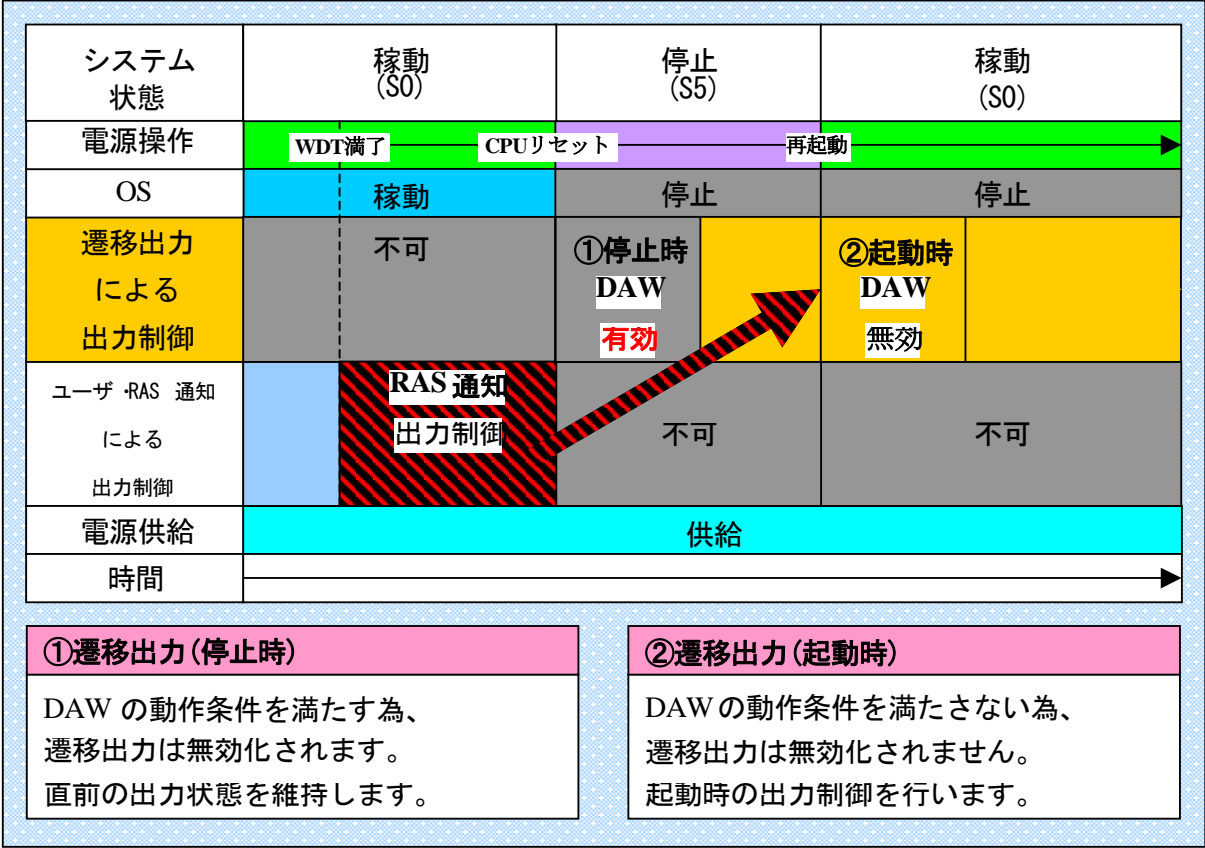
DAW機能は、以下の要求を満たす際に使用します。

- ・ WDT満了に対して、RAS通知かつCPUリセットを行う時に、遷移出力を無効にしたい。

WDT満了時にDAWを動作させるには、下表の設定条件を満たす必要があります。

機能	設定
遷移出力	・ DAWを有効に設定している。
RAS通知	・ WDT満了の通知を行う制御対象が、遷移出力と同じである。
WDT	・ WDTを動作させている。 ・ WDT満了時にCPUリセットを行う設定である。

動作事例として「停止時のDAWが有効、起動時のDAWが無効」である場合を、下図に示します。



デジタル出力の接点別に、DAWを設定することはできません。

2.17.3 コマンドを用いた設定方法

遷移出力の設定を行うには、CMTRANSCONFIG命令を使用します。
本命令により、以下の設定が可能です。

設定	説明
制御対象	デジタル出力(DOUT1-4)/外部コネクタ出力(EX_SWOUT)
制御タイミング	停止時/起動時
制御設定	有効/無効(接点別)
出力設定	ON/OFF(接点別)
DAW設定	有効/無効

以下の例は、デジタル出力をシステム停止時に全接点ON、DAWをONにする設定です。

```
CMTRANSCONFIG 0, 0, &HF, &HF, 1
```

詳しい説明は「CMTRANSCONFIG」を参照してください。

2.18 製品情報取得

2.18.1 ファームウェアバージョンの確認方法

CMVERGET命令でファームウェアバージョンを確認することができます。

RAS Controller・BIOSの初回バージョンは0、Ver.UPのたびに1, 2, 3,..となります。

弊社ドキュメントやBIOSのメニュー上では、バージョンはアルファベット表記となっており、以下の対応となります。

バージョン	表記
0	(空白)
1	A
2	B
...	...

第3章 リファレンス

使用できるシステム監視制御コマンドの使い方について記載します。

3.1 コマンド一覧

コマンド名	機能
●WDT	
WDTCONFIG	ウォッチドッグタイマの周期とアクションを設定します。
WDTSTART	ウォッチドッグタイマのカウントを開始します。
WDTSTOP	ウォッチドッグタイマのカウントを停止します。
WDTCLEAR	ウォッチドッグタイマのカウントをクリアします。
●温度監視	
TEMPGET	指定対象の温度を取得します。
TEMPLIMIT	温度の異常上限、下限、および温度に対するヒステリシスを設定します。
TEMPUPPERLIMIT	温度警告と温度異常検出の上限値及びヒステリシスを設定します。
TEMPLOWERLIMIT	温度警告と温度異常検出の下限値及びヒステリシスを設定します。
●電源電圧監視	
POWERGET	指定対象の電源電圧を取得します。
POWERLIMIT	電源電圧の上限、下限を設定します。
DIGITALFILTER	電源電圧異常検出時のデジタルフィルタを設定します。
●RAS出力	
ROOUTPUT	RAS出力端子(外部コネクタ出力端子)を制御します。
ROCONFIG	自動RAS出力条件を設定します。
ROSTATUS	RAS出力の状態を取得します。
●RAS通知	
RASNOTIFY	異常発生時の自動通知(RAS通知)方法と条件、通知条件を満たしたときのデジタル出力接点状態を設定します。
●デジタル入出力	
OUTPUTUP	デジタル出力接点を制御します。
INPUTUP	デジタル入力状態を読み込みます。
UPCONFIG	デジタル入力接点の機能を設定します。
●割り込みイベント	
CMEVENTCONFIG	システム監視割り込み発生要因を設定します。
ON CM CALL	指定したシステム監視要因発生時に、呼び出されるサブルーチンを定義します。
CMEVENT	発生したシステム監視割り込み要因を取得します。
CM ON	システム監視割り込みを有効にします。
CM OFF	システム監視割り込みを無効にします。
CM STOP	システム監視割り込みを保留にします。
●瞬低対策用電源監視	
BATSTATUS	バッテリーステータスを取得します。
BATPOWEROFFTIME	停電発生からシャットダウン/電源強制断までの時間を設定します。
●LED制御	
LEDCONTROL	LEDを制御します。
LEDPATTERN	LEDの点灯パターンを設定します。
CMLED	LEDを制御します。
●ブザー制御	
BUZZERCONTROL	ブザーを制御します。
CMBUZZER	ブザーを制御します。
●ひもスイッチ	
HIMOSWITCHSTATUS	現在のひもスイッチの状態を取得します。
HIMOSWITCHCONFIG	ひもスイッチの機能を設定します。

コマンド名	機能
●FAN監視	
FANROTATEGET	FANの回転数を取得します。
FANAUTOTHRESHOLD	FAN自動制御の温度閾値を設定します。
FANAUTOSPEED	FAN自動制御の速度を%で設定します。
●衝撃・振動監視	
GSENSEGET	指定対象の加速度センサ値を取得します。
GSENSELIMIT	振動・衝撃異常検出の閾値を設定します。
●外部コネクタ出力	
EXCONTROL	外部コネクタ出力(EX_SWOUT)を制御します。
●ストレージ監視	
RAIDSTATUS	RAIDステータスを取得します。
●自己診断	
RASERRORSTATUS	現在発生中の異常状態を取得します。
RASCONTROLSTATUS	現在の制御状態を取得します。
●遷移出力	
CMTRANSCONFIG	遷移出力の設定を行います。
●スロットイン電源モジュール監視	
CMPSMGET	スロットイン電源モジュールの情報を取得します。
●製品情報	
CMVERGET	ファームウェアバージョンを取得します。

3.2 詳細個別説明（WDT）

1.WDTCONFIG

命令			
機能	ウォッチドッグタイマの周期とアクションを設定します。		
書式	WDTCONFIG <①周期>,<②アクション>		
パラメータ	①	<周期>	
	数値		
	WDT の周期を設定します。設定できる範囲は 1～255 で、単位は秒です。 (デフォルト:30秒)		
	②	<アクション>	
	文字列		
	ウォッチドッグタイマ周期満了時のアクションを指定します。 「識別子=設定値」の形式で、文字列で指定します。		
	識別子	設定値	内容
	ACTION	EVENT	EVENT:周期満了時に割り込みイベントを発生させます。
		RESET	周期満了時にCPUリセットを起こします。
			デフォルト
			○
	BIOS設定でウォッチドッグタイマを有効にしている場合は、その設定となります。		
備考	・ WDT動作中に、本コマンドを実行すると、新しく設定した周期で、再度カウントし直します。 ・ 複数の異なるアプリケーションで本コマンドを実行した場合、最後に設定した周期、アクションが有効となります。 ・ BIOS設定でWDTを有効にしている場合は、OS起動時にその設定が引き継がれます。		
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmSetWDTConfig関数を使用しています。		
使用例	『2.1 WDT』を参照してください。		

2.WDTSTART

命令	
機 能	ウォッチドッグタイマのカウントを開始します。
書 式	WDTSTART
備 考	一度WDTを停止させた後、再度本コマンドを実行した場合、カウント数は最初からカウントし直されます。
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmStartWDT関数を使用しています。
使用例	『2.1 WDT』を参照してください。

3.WDTSTOP

命令	
機 能	ウォッチドッグタイマのカウントを停止します。
書 式	WDTSTOP
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmStopWDT関数を使用しています。
使用例	『2.1 WDT』を参照してください。

4.WDTCLEAR

命令	
機 能	ウォッチドッグタイマのカウントをクリアします。 クリアすると設定した周期の値でカウントし直します。
書 式	WDTCLEAR
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmClearWDT関数を使用しています。
使用例	『2.1 WDT』を参照してください。

3.3 詳細個別説明（温度監視）

1.TEMPGET

関数		
機 能	指定対象の温度を取得します。	
書 式	<(戻り値)温度> = TEMPGET(<①対象>)	
戻り値	戻り値	<温度> 数値
	指定した対象の温度を取得します。（単位は℃です。） 負数が得られることもあります。	
パラ メータ	①	<対象> 数値
	温度を取得する対象を指定します。	
	機能毎に指定できるパラメータは、『5.2 機能対応表』を参照してください。	
	対象値	指定対象(DPG-0226使用時) 指定対象(DPG-0200使用時)
	1	CPU温度 CPU温度
	2	基板温度 基板温度
	3	メモリ周辺温度 —
	4	拡張スロット周辺温度1 CPUコア2温度
	5	拡張スロット周辺温度2 CPUコア3温度
	6	拡張スロット周辺温度3 CPUコア4温度
備 考	弊社リアルタイム製品等により占有されているCPUコアを指定した場合、「ドライバ内部でエラーが発生しました」が発生します。	
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmGetTemp関数を使用しています。	
使用例	『2.2 温度監視』を参照してください。	

2. TEMPLIMIT

命令			
機能	温度の異常上限、下限、および温度に対するヒステリシスを設定します。		
書式	TEMPLIMIT <①対象>, <②下限>, <③上限>, <④下限復帰ヒステリシス>, <⑤上限復帰ヒステリシス>		
パラメータ	①	<対象>	数値
	温度監視を設定する対象を指定します。 機能毎に指定できるパラメータは、『5.2 機能対応表』を参照してください。		
	対象値	指定対象(DPG-0226使用時)	指定対象(DPG-0200使用時)
	1	CPU温度	CPU温度
	2	基板温度	基板温度
	3	メモリ周囲温度	—
	②	<下限>	数値
	監視温度の下限値です。-60～<上限>より小さい値が設定できます。 (-60 ≤ 設定可能値 < <上限>) デフォルトはCPU、基板共に-60です。単位は℃です。		
	③	<上限>	数値
	監視温度の上限値です。<下限>より大きい値～125が設定できます。 (<下限> < 設定可能値 ≤ 125) デフォルトはCPU、基板共に125です。単位は℃です。		
	④	<下限復帰ヒステリシス>	数値
	温度下限復帰ヒステリシスです。 CPUコア温度は0～15、CPU基板温度は0～31の間で設定できます。単位は℃です。		
	⑤	<上限復帰ヒステリシス>	数値
	温度上限復帰ヒステリシスです。 CPUコア温度は0～15、CPU基板温度は0～31の間で設定できます。 単位は℃です。		
注意	ここで設定できる値は、本ソフトウェアで監視できる値です。ここで設定できる範囲がハードウェアの保証温度ではありませんのでご注意ください。		
備考	<ul style="list-style-type: none"> 複数の異なるアプリケーションで本コマンドを実行した場合、最後に設定した上限、下限が有効となります。 マルチコアCPUの場合、各コアのいずれかの温度がリミットを越えると異常とみなします。 		
使用関数	本コマンドは、システム監視ソフトウェア (DPG-0226/0200) の、CmSetTempLimit、CmSetTempLimitEx関数を使用しています。		
使用例	『2.2 温度監視』を参照してください。		

3. TEMPUPPERLIMIT

命令									
機 能	温度警告と温度異常検出の上限値及びヒステリシスを設定します。								
書 式	TEMPUPPERLIMIT <①対象>, <②温度警告の上限値>, <③温度異常の上限値>, <④温度上限警告のヒステリシス>, <⑤温度上限異常のヒステリシス>								
パラ メータ	①	<対象>	数値						
	温度警告と温度異常検出を設定する対象を指定します。								
	<table><tr><td>設定値</td><td>指定対象</td></tr><tr><td>1</td><td>CPU温度</td></tr><tr><td>2</td><td>基板温度</td></tr></table>			設定値	指定対象	1	CPU温度	2	基板温度
	設定値	指定対象							
	1	CPU温度							
	2	基板温度							
	②	<温度警告の上限値>	数値						
温度警告の上限値を指定します。単位は℃です。 警告下限値～温度異常の上限値の範囲で設定可能です。 (警告下限値 < 設定可能値 < 温度異常の上限値)									
③	<温度異常の上限値>	数値							
温度異常の上限値を指定します。単位は℃です。 温度警告の上限値～125の範囲で設定可能です。 (温度警告の上限値 < 設定可能値 <= 125)									
④	<温度上限警告のヒステリシス>	数値							
温度警告の上限の復帰ヒステリシスを指定します。 CPU コア温度は 0～15、CPU 基板温度は 0～31 の間で設定できます。単位は℃です。									
⑤	<温度上限異常のヒステリシス>	数値							
温度異常の上限の復帰ヒステリシスを指定します。 CPU コア温度は 0～15、CPU 基板温度は 0～31 の間で設定できます。単位は℃です。									
注 意	ここで設定できる値は、本ソフトウェアで監視できる値です。ここで設定できる範囲がハードウェアの保証温度ではありませんのでご注意ください。								
備 考	・ 複数の異なるアプリケーションで本コマンドを実行した場合、最後に設定した上限、下限が有効となります。 ・ TEMPUPPERLIMIT命令とTEMPLOWERLIMIT命令の各パラメータの関係式は、以下のとおりです。 -60≦温度異常の下限値<温度警告の下限値<温度警告の上限値<温度異常の上限値≦125								
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmSetTempUpperLimit関数を使用しています。								
使用例	『2.2 温度監視』を参照してください。								

4. TEMPLOWERLIMIT

命令									
機能	温度警告と温度異常検出の下限値及びヒステリシスを設定します。								
書式	TEMPLOWERLIMIT <①対象>, <②温度警告の下限値>, <③温度異常の下限値>, <④温度下限警告のヒステリシス>, <⑤温度下限異常のヒステリシス>								
パラメータ	①	<対象>	数値						
	温度警告と温度異常検出を指定する対象を指定します。								
	<table><tr><td>対象値</td><td>指定対象</td></tr><tr><td>1</td><td>CPU温度</td></tr><tr><td>2</td><td>基板温度</td></tr></table>			対象値	指定対象	1	CPU温度	2	基板温度
	対象値	指定対象							
	1	CPU温度							
	2	基板温度							
	②	<温度警告の下限値>	数値						
温度警告の下限値を指定します。単位は℃です。 温度異常の下限値～警告上限値の範囲で設定可能です。 (温度異常の下限値< 設定可能値 < 警告上限値)									
③	<温度異常の下限値>	数値							
温度異常の下限値を指定します。単位は℃です。 -60～温度警告の下限値の範囲で設定可能です。 (-60<= 設定可能値 < 温度警告の下限値)									
④	<温度下限警告のヒステリシス>	数値							
温度警告の下限の復帰ヒステリシスを指定します。 CPU コア温度は 0～15、CPU 基板温度は 0～31 の間で設定できます。単位は℃です。									
⑤	<温度下限異常のヒステリシス>	数値							
温度異常の下限の復帰ヒステリシスを指定します。 CPU コア温度は 0～15、CPU 基板温度は 0～31 の間で設定できます。単位は℃です。									
注意	ここで設定できる値は、本ソフトウェアで監視できる値です。ここで設定できる範囲がハードウェアの保証温度ではありませんのでご注意ください。								
備考	・ 複数の異なるアプリケーションで本コマンドを実行した場合、最後に設定した上限、下限が有効となります。 ・ TEMPUPPERLIMIT 命令と TEMPLOWERLIMIT 命令の各パラメータの関係式は、以下のとおりです。 -60≦温度異常の下限値<温度警告の下限値<温度警告の上限値<温度異常の上限値≦125								
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmSetTempLowerLimit関数を使用しています。								
使用例	『2.2 温度監視』を参照してください。								

3.4 詳細個別説明（電源電圧監視）

1. POWERGET

関数			
機 能	指定対象の電源電圧を取得します。		
書 式	<(戻り値)電圧> = POWERGET(<①対象>)		
戻り値	戻り値	<電圧>	数値
	指定した対象の電圧(単位:V)を取得します。		
パラ メータ	①	<対象>	数値
	電圧を取得する対象を指定します。		
	設定値	指定対象	
	1	外部電源電圧	
備 考	SuperCD PoE受電モデルでは、本機能は非サポートとなります。DC電源モデルと互換のため本コマンドは実行できますが、取得できる値は実際の値とは異なります。		
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmGetPower関数を使用しています。		
使用例	『2.3 電源電圧監視』を参照してください。		

2.POWERLIMIT

命令		
機 能	電源電圧の上限、下限を設定します。 ここで設定した範囲を超えた場合に、割り込みを通知することができます。 電源電圧異常発生から異常復帰までの範囲も設定できます。	
書 式	POWERLIMIT <①対象>, <②下限値>, <③上限値>, <④下限異常のヒステリシス>, <⑤上限異常のヒステリシス>	
パラ メータ	①	<対象> 数値
	電源の監視電圧を設定する対象を指定します。	
	設定値	指定対象
	1	外部電源電圧
	②	<下限値> 数値
	電源の監視電圧下限値を設定します。下限値は設定値に 0.22(V)を掛けたものになります。 設定範囲は 25～上限値より小さい値です。 例) 25と設定した場合、下限値は $25 \times 0.22 = 5.50(\text{V})$ となります。	
	③	<上限値> 数値
	電源の監視電圧上限値を設定します。上限は設定値に 0.22(V)を掛けたものになります。 設定範囲は下限値より大きい値～170 です。 例) 170と設定した場合、上限値は $170 \times 0.22 = 37.4(\text{V})$ となります。	
	④	<下限異常のヒステリシス> 数値
	電源の監視電圧下限異常のヒステリシスです。0～31 の間で設定できます。 範囲は設定値に 0.22(V)を掛けたものになります。 例) 1と設定した場合、範囲は0.22(V)となります。	
	⑤	<上限異常のヒステリシス> 数値
	電源の監視電圧上限異常のヒステリシスです。0～31 の間で設定できます。 範囲は設定値に 0.22(V)を掛けたものになります。 例) 1と設定した場合、範囲は0.22(V)となります。	
備 考	複数の異なるアプリケーションで本コマンドを実行した場合、最後に設定した上限、下限が有効となります。	
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmSetPowerLimit関数を使用しています。	
使用例	『2.3 電源電圧監視』を参照してください。	

3.DIGITALFILTER

命令			
機 能	電源電圧異常検出時のデジタルフィルタを設定します。		
書 式	DIGITALFILTER <①対象>, <②フィルタ値>		
パラ メータ	①	<対象>	数値
	デジタルフィルタを設定する対象を指定します。		
	設定値	意味	
	1	外部電圧異常検出のデジタルフィルタ	
	②	<フィルタ値>	数値
異常を検知し、設定期間異常状態が継続した場合のみに異常と判断します。 単位はmsです。			
	フィルタ値	内容	
	0	フィルタ無し	
	1～255	フィルタ値	
備 考	複数の異なるアプリケーションで本関数を実行した場合、最後に設定したフィルタ値が有効になります。		
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmSetDigitalFilter関数を使用しています。		
使用例	『2.3 電源電圧監視』を参照してください。		

3.5 詳細個別説明 (RAS出力)

1. ROOUTPUT

命令			
機 能	RAS出力端子(外部コネクタ出力端子)を制御します。		
書 式	ROOUTPUT <①制御値>		
パラ メータ	①	<制御値>	文字列
	RAS出力の制御値を指定します。 「識別子=設定値」の形式で、文字列で指定します。		
	識別子	設定値	内容
	RASOUT	ON OFF	RAS出力をON RAS出力をOFF
備 考	<ul style="list-style-type: none"> システムの電源投入時、RAS出力はリレーOFFになります。OS再起動、およびシャットダウン時でも、電源が待機状態であれば、RAS出力状態は保持されます。 		
使用関数	<ul style="list-style-type: none"> 本コマンドは、システム監視ソフトウェア(DPG-0200)の、CmControlRO関数を使用しています。 		
使用例	『2.4 RAS出力』を参照してください。		

2.ROCONFIG

命令																			
機能	自動RAS出力条件を設定します。																		
書式	ROCONFIG <①制御値>, <②端子状態>																		
パラメータ	①	<制御値>																	
	RAS出力の制御値を指定します。																		
	<table><tr><th>制御値</th><th>内容</th></tr><tr><td>0</td><td>出力条件なし(出荷時設定)</td></tr><tr><td>&H1</td><td>WDT周期満了時にRAS出力を行います。</td></tr><tr><td>&H2</td><td>CPU温度上限超え時にRAS出力を行います。</td></tr><tr><td>&H4</td><td>CPU温度下限超え時にRAS出力を行います。</td></tr><tr><td>&H8</td><td>基板温度上限超え時にRAS出力を行います。</td></tr><tr><td>&H10</td><td>基板温度下限超え時にRAS出力を行います。</td></tr><tr><td>&H20</td><td>電源電圧異常時にRAS出力を行います。</td></tr></table>			制御値	内容	0	出力条件なし(出荷時設定)	&H1	WDT周期満了時にRAS出力を行います。	&H2	CPU温度上限超え時にRAS出力を行います。	&H4	CPU温度下限超え時にRAS出力を行います。	&H8	基板温度上限超え時にRAS出力を行います。	&H10	基板温度下限超え時にRAS出力を行います。	&H20	電源電圧異常時にRAS出力を行います。
	制御値	内容																	
	0	出力条件なし(出荷時設定)																	
	&H1	WDT周期満了時にRAS出力を行います。																	
	&H2	CPU温度上限超え時にRAS出力を行います。																	
	&H4	CPU温度下限超え時にRAS出力を行います。																	
	&H8	基板温度上限超え時にRAS出力を行います。																	
	&H10	基板温度下限超え時にRAS出力を行います。																	
&H20	電源電圧異常時にRAS出力を行います。																		
複数の条件をORで設定することもできます。																			
②	<端子状態>																		
異常時にRAS出力する端子状態を指定します。																			
<table><tr><th>識別子</th><th>設定値</th><th>内容</th></tr><tr><td rowspan="2">RASOUT=</td><td>ON</td><td>RAS出力をON</td></tr><tr><td>OFF</td><td>RAS出力をOFF</td></tr></table>			識別子	設定値	内容	RASOUT=	ON	RAS出力をON	OFF	RAS出力をOFF									
識別子	設定値	内容																	
RASOUT=	ON	RAS出力をON																	
	OFF	RAS出力をOFF																	
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0200)の、CmSetROConfig関数を使用しています。																		
使用例	『2.4 RAS出力』を参照してください。																		

3.ROSTATUS

関数			
機 能	RAS出力の状態を取得します。		
書 式	<(戻り値)ステータス>= ROSTATUS		
戻り値	戻り値	<ステータス>	数値
	RAS出力の動作状態 0 : リレーOFF 1 : リレーON		
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0200)の、CmGetSpecifiedStatus関数を使用しています。		
使用例	『2.4 RAS出力』を参照してください。		

3.6 詳細個別説明 (RAS通知)

1.RASNOTIFY

命令

機能	異常発生時の自動通知(RAS通知)方法と条件、通知条件を満たしたときのデジタル出力接点状態を設定します。																																												
書式	RASNOTIFY <①方法>, <②条件> [, <③出力>]																																												
パラメータ	①	<方法>	文字列																																										
	RAS通知の方法を指定します。 「識別子=設定値」の形式で、文字列で指定します。																																												
		<table><tr><th>識別子</th><th>設定値</th><th>内容</th></tr><tr><td rowspan="19">ENABLE</td><td>BUZZER</td><td>ブザーをON</td></tr><tr><td>ERRLED</td><td>エラーLEDを制御</td></tr><tr><td>ALMLED</td><td>アラームLEDを制御</td></tr><tr><td>DOUT1</td><td>デジタル出力接点1を制御</td></tr><tr><td>DOUT2</td><td>デジタル出力接点2を制御</td></tr><tr><td>DOUT3</td><td>デジタル出力接点3を制御</td></tr><tr><td>DOUT4</td><td>デジタル出力接点4を制御</td></tr><tr><td>EXO</td><td>外部コネクタ出力端子を制御</td></tr><tr><td>LED1</td><td>汎用LED1を制御</td></tr><tr><td>LED2</td><td>汎用LED2を制御</td></tr><tr><td>LED3</td><td>汎用LED3を制御</td></tr><tr><td>LED4</td><td>汎用LED4を制御</td></tr><tr><td>LED5</td><td>汎用LED5を制御</td></tr><tr><td>LED6</td><td>汎用LED6を制御</td></tr><tr><td>LED7</td><td>汎用LED7を制御</td></tr><tr><td>LED8</td><td>汎用LED8を制御</td></tr><tr><td>LED9</td><td>汎用LED9を制御</td></tr><tr><td>LED10</td><td>汎用LED10を制御</td></tr></table>	識別子	設定値	内容	ENABLE	BUZZER	ブザーをON	ERRLED	エラーLEDを制御	ALMLED	アラームLEDを制御	DOUT1	デジタル出力接点1を制御	DOUT2	デジタル出力接点2を制御	DOUT3	デジタル出力接点3を制御	DOUT4	デジタル出力接点4を制御	EXO	外部コネクタ出力端子を制御	LED1	汎用LED1を制御	LED2	汎用LED2を制御	LED3	汎用LED3を制御	LED4	汎用LED4を制御	LED5	汎用LED5を制御	LED6	汎用LED6を制御	LED7	汎用LED7を制御	LED8	汎用LED8を制御	LED9	汎用LED9を制御	LED10	汎用LED10を制御			
識別子	設定値	内容																																											
ENABLE	BUZZER	ブザーをON																																											
	ERRLED	エラーLEDを制御																																											
	ALMLED	アラームLEDを制御																																											
	DOUT1	デジタル出力接点1を制御																																											
	DOUT2	デジタル出力接点2を制御																																											
	DOUT3	デジタル出力接点3を制御																																											
	DOUT4	デジタル出力接点4を制御																																											
	EXO	外部コネクタ出力端子を制御																																											
	LED1	汎用LED1を制御																																											
	LED2	汎用LED2を制御																																											
	LED3	汎用LED3を制御																																											
	LED4	汎用LED4を制御																																											
	LED5	汎用LED5を制御																																											
	LED6	汎用LED6を制御																																											
	LED7	汎用LED7を制御																																											
	LED8	汎用LED8を制御																																											
	LED9	汎用LED9を制御																																											
	LED10	汎用LED10を制御																																											
		②	<条件>	数値																																									
	RAS通知する条件を指定します。 下記の値が指定できます。複数の値をORで指定することも可能です。																																												
		<table><tr><th>値</th><th>意味</th></tr><tr><td>0</td><td>自動通知を行いません。</td></tr><tr><td>&H1</td><td>WDT 周期満了の際に RAS 通知を行います。</td></tr><tr><td>&H4</td><td>CPU コア温度異常上限超えの際に RAS 通知を行います。</td></tr><tr><td>&H8</td><td>CPU コア温度異常下限超えの際に RAS 通知を行います。</td></tr><tr><td>&H10</td><td>CPU 基板温度異常上限超えの際に RAS 通知を行います。</td></tr><tr><td>&H20</td><td>CPU 基板温度異常下限超えの際に RAS 通知を行います。</td></tr><tr><td>&H40</td><td>外部電源電圧異常の際に RAS 通知を行います。 ※AC 電源タイプの製品では内部 DC 出力電圧異常時に RAS 通知を行います。</td></tr><tr><td>&H100</td><td>メモリ周辺温度異常上限超えの際に RAS 通知を行います。</td></tr><tr><td>&H200</td><td>メモリ周辺温度異常下限超えの際に RAS 通知を行います。</td></tr><tr><td>&H10000</td><td>CPU コア温度警告上限超えの際に RAS 通知を行います。</td></tr><tr><td>&H20000</td><td>CPU コア温度警告下限超えの際に RAS 通知を行います。</td></tr><tr><td>&H40000</td><td>CPU 基板温度警告上限超えの際に RAS 通知を行います。</td></tr><tr><td>&H80000</td><td>CPU 基板温度警告下限超えの際に RAS 通知を行います。</td></tr><tr><td>&H100000</td><td>X 軸振動・衝撃異常検知の際に RAS 通知を行います。</td></tr><tr><td>&H200000</td><td>Y 軸振動・衝撃異常検知の際に RAS 通知を行います。</td></tr><tr><td>&H400000</td><td>Z 軸振動・衝撃異常検知の際に RAS 通知を行います。</td></tr><tr><td>&H1000000</td><td>FAN1 異常の際に RAS 通知を行います。</td></tr><tr><td>&H2000000</td><td>FAN2 異常の際に RAS 通知を行います。</td></tr><tr><td>&H4000000</td><td>FAN3 異常の際に RAS 通知を行います。</td></tr><tr><td>&H40000000</td><td>RAID 異常の際に RAS 通知を行います。</td></tr></table>	値	意味	0	自動通知を行いません。	&H1	WDT 周期満了の際に RAS 通知を行います。	&H4	CPU コア温度異常上限超えの際に RAS 通知を行います。	&H8	CPU コア温度異常下限超えの際に RAS 通知を行います。	&H10	CPU 基板温度異常上限超えの際に RAS 通知を行います。	&H20	CPU 基板温度異常下限超えの際に RAS 通知を行います。	&H40	外部電源電圧異常の際に RAS 通知を行います。 ※AC 電源タイプの製品では内部 DC 出力電圧異常時に RAS 通知を行います。	&H100	メモリ周辺温度異常上限超えの際に RAS 通知を行います。	&H200	メモリ周辺温度異常下限超えの際に RAS 通知を行います。	&H10000	CPU コア温度警告上限超えの際に RAS 通知を行います。	&H20000	CPU コア温度警告下限超えの際に RAS 通知を行います。	&H40000	CPU 基板温度警告上限超えの際に RAS 通知を行います。	&H80000	CPU 基板温度警告下限超えの際に RAS 通知を行います。	&H100000	X 軸振動・衝撃異常検知の際に RAS 通知を行います。	&H200000	Y 軸振動・衝撃異常検知の際に RAS 通知を行います。	&H400000	Z 軸振動・衝撃異常検知の際に RAS 通知を行います。	&H1000000	FAN1 異常の際に RAS 通知を行います。	&H2000000	FAN2 異常の際に RAS 通知を行います。	&H4000000	FAN3 異常の際に RAS 通知を行います。	&H40000000	RAID 異常の際に RAS 通知を行います。	
値	意味																																												
0	自動通知を行いません。																																												
&H1	WDT 周期満了の際に RAS 通知を行います。																																												
&H4	CPU コア温度異常上限超えの際に RAS 通知を行います。																																												
&H8	CPU コア温度異常下限超えの際に RAS 通知を行います。																																												
&H10	CPU 基板温度異常上限超えの際に RAS 通知を行います。																																												
&H20	CPU 基板温度異常下限超えの際に RAS 通知を行います。																																												
&H40	外部電源電圧異常の際に RAS 通知を行います。 ※AC 電源タイプの製品では内部 DC 出力電圧異常時に RAS 通知を行います。																																												
&H100	メモリ周辺温度異常上限超えの際に RAS 通知を行います。																																												
&H200	メモリ周辺温度異常下限超えの際に RAS 通知を行います。																																												
&H10000	CPU コア温度警告上限超えの際に RAS 通知を行います。																																												
&H20000	CPU コア温度警告下限超えの際に RAS 通知を行います。																																												
&H40000	CPU 基板温度警告上限超えの際に RAS 通知を行います。																																												
&H80000	CPU 基板温度警告下限超えの際に RAS 通知を行います。																																												
&H100000	X 軸振動・衝撃異常検知の際に RAS 通知を行います。																																												
&H200000	Y 軸振動・衝撃異常検知の際に RAS 通知を行います。																																												
&H400000	Z 軸振動・衝撃異常検知の際に RAS 通知を行います。																																												
&H1000000	FAN1 異常の際に RAS 通知を行います。																																												
&H2000000	FAN2 異常の際に RAS 通知を行います。																																												
&H4000000	FAN3 異常の際に RAS 通知を行います。																																												
&H40000000	RAID 異常の際に RAS 通知を行います。																																												
	③																																												

	③	<出力>		文字列
	条件を満たした際のデジタル出力の状態を設定します。 RAS通知の方法をデジタル出力接点もしくは外部コネクタ出力端子にした場合のみ指定してください。			
		識別子	設定値	内容
		ENABLE	ON	出力をON
			OFF	出力をOFF
備 考	<ul style="list-style-type: none">・エラーLED、アラームLEDはLEDPATTERN命令で設定した点灯パターンで制御されます。・<①方法>にアラームLEDを指定した場合、<②条件>には以下の値のみ指定できます。 CPUコア温度上限警告 (&H10000) CPUコア温度下限警告 (&H20000) CPU基板温度上限警告 (&H40000) CPU基板温度下限警告 (&H80000) <①方法>にアラームLED以外を指定した場合、<②条件>には上記の値は指定できません。・通知条件はOR条件での指定のみ可能です。AND条件での指定はできません。・RAIDボード検出途中は、RAID異常によるRAS通知は発生しません。詳しくは、RAIDボードのUSER'S MANUALをご覧ください。			
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmSetRasNotify関数を使用しています。			
使用例	『2.5 RAS通知』を参照してください。			

3.7 詳細個別説明（デジタル入出力）

1.OUTPUTUP

命令																					
機 能	デジタル出力接点を制御します。																				
書 式	OUTPUTUP(<①ポート番号>)=(<②出力値>)																				
パラ メータ	①	<ポート番号>	数値、配列																		
	デジタル出力のポート番号を設定します。 ポート番号を数値型変数で指定した場合（1点だけ指定した場合）、指定した1点のポートからデジタルデータを出力します。 ポート番号を数値配列型変数で指定した場合、指定したポート番号の数に応じたデジタルデータを出力します。																				
	<table><tr><th>ポート番号値</th><th>デジタル出力接点名</th></tr><tr><td>1</td><td>DOUT1</td></tr><tr><td>2</td><td>DOUT2</td></tr><tr><td>3</td><td>DOUT3</td></tr><tr><td>4</td><td>DOUT4</td></tr><tr><td>5</td><td>DOUT5</td></tr><tr><td>6</td><td>DOUT6</td></tr><tr><td>7</td><td>DOUT7</td></tr><tr><td>8</td><td>DOUT8</td></tr></table>			ポート番号値	デジタル出力接点名	1	DOUT1	2	DOUT2	3	DOUT3	4	DOUT4	5	DOUT5	6	DOUT6	7	DOUT7	8	DOUT8
	ポート番号値	デジタル出力接点名																			
	1	DOUT1																			
2	DOUT2																				
3	DOUT3																				
4	DOUT4																				
5	DOUT5																				
6	DOUT6																				
7	DOUT7																				
8	DOUT8																				
②	<出力値>	数値、配列																			
デジタル出力接点データを指定します。 0 : OFF 1 : ON																					
備 考	・ 出力値の配列とポート番号の配列は、同一の要素数を持つものを指定する必要があります。 ・ 出力値を指定しなかったポートの出力は、以前の出力を継続します。 ・ デジタル出力のポート番号の最大数は、実際に制御するハードウェアに依存します。 （製品によって、4点あるものと8点あるものがあります） ・ ポート番号や出力値の指定で複数点を指定する時、配列変数を指定する他、[] で値を列挙して指定可能です。 例： OUTPUTUP([1 to 4])= [1; 0; 1; 0]																				
使用関数	・ 本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmOutputUP関数を使用しています。																				
使用例	『2.6 デジタル入出力』を参照してください。																				

2.INPUTUP

関数																					
機 能		デジタル入力状態を読み込みます。																			
書 式		<(戻り値)入力値> = INPUTUP(<①ポート番号>)																			
戻り値	戻り値	<入力値>	配列																		
	デジタル入力接点情報の配列が得られます。 0 : OFF 1 : ON																				
パラ メータ	①	<ポート番号>	数値、配列																		
	デジタル入力のポート番号を設定します。 ポート番号を数値型変数で指定した場合（１点だけ指定した場合）、指定した１点のポートからデジタルデータを入力します。 ポート番号を数値配列型変数で指定した場合、指定したポート番号の数に応じたデジタルデータを入力します。																				
		<table><tr><th>ポート番号値</th><th>デジタル入力接点名</th></tr><tr><td>1</td><td>DIN1</td></tr><tr><td>2</td><td>DIN2</td></tr><tr><td>3</td><td>DIN3</td></tr><tr><td>4</td><td>DIN4</td></tr><tr><td>5</td><td>DIN5</td></tr><tr><td>6</td><td>DIN6</td></tr><tr><td>7</td><td>DIN7</td></tr><tr><td>8</td><td>DIN8</td></tr></table>	ポート番号値	デジタル入力接点名	1	DIN1	2	DIN2	3	DIN3	4	DIN4	5	DIN5	6	DIN6	7	DIN7	8	DIN8	
ポート番号値	デジタル入力接点名																				
1	DIN1																				
2	DIN2																				
3	DIN3																				
4	DIN4																				
5	DIN5																				
6	DIN6																				
7	DIN7																				
8	DIN8																				
備 考		・ 入力値の配列とポート番号の配列は、同一の要素数を持つものを指定する必要があります。 ・ デジタル入力のポート番号の最大数は、実際に制御するハードウェアに依存します。 （製品によって、4点あるものと8点あるものがあります） ・ ポート番号の指定で複数点を指定する時、配列変数を指定する他、[] で値を列挙して指定可能です。 例： ARY = INPUTUP([1 to 4])																			
使用関数		・ 本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmInputUP関数を使用しています。																			
使用例		『2.6 デジタル入出力』を参照してください。																			

3.UPCONFIG

命令										
機 能	デジタル入力接点の機能を設定します。									
書 式	UPCONFIG <①機能>									
パラ メータ	①	<機能> 文字列								
	割り当てる機能を設定します。									
	「識別子=設定値」の形式で、文字列で指定します。									
	<table><tr><td>識別子</td><td>設定値</td><td>内容</td></tr><tr><td rowspan="2">DI1</td><td>DI</td><td>デジタル入力接点1を汎用入力</td></tr><tr><td>PSWITCH</td><td>デジタル入力接点1を電源スイッチ</td></tr></table>		識別子	設定値	内容	DI1	DI	デジタル入力接点1を汎用入力	PSWITCH	デジタル入力接点1を電源スイッチ
	識別子	設定値	内容							
DI1	DI	デジタル入力接点1を汎用入力								
	PSWITCH	デジタル入力接点1を電源スイッチ								
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmSetDIConfig関数を使用しています。									
使用例	『2.6 デジタル入出力』を参照してください。									

3.8 詳細個別説明（割り込みイベント）

1.CMEVENTCONFIG

命令

機能	システム監視割り込み発生要因を設定します。			
書式	CMEVENTCONFIG <①割り込み定義>			
パラメータ	①	<割り込み定義>	文字列	
	システム監視割り込みの定義を設定します。(文字列で指定します) 「識別子=設定値」の形式で、文字列で指定します。複数列挙する場合は「,」で区切ります。			
■対応表				
	識別子	設定値	内容	デフォルト
	WDT	OFF	WDT周期満了割り込み無効	○
		ON	WDT周期満了割り込み有効	
	CPTH	OFF	CPUコア温度異常上限超え割り込み無効	○
		ON	CPUコア温度異常上限超え割り込み有効	
	CPTL	OFF	CPUコア温度異常下限超え割り込み無効	○
		ON	CPUコア温度異常下限超え割り込み有効	
	LCTH	OFF	CPU基板温度異常上限超え割り込み無効	○
		ON	CPU基板温度異常上限超え割り込み有効	
	LCTL	OFF	CPU基板温度異常下限超え割り込み無効	○
		ON	CPU基板温度異常下限超え割り込み有効	
	METH	OFF	メモリ周囲温度異常上限超え割り込み無効	○
		ON	メモリ周囲温度異常上限超え割り込み有効	
	METL	OFF	メモリ周囲温度異常下限超え割り込み無効	○
		ON	メモリ周囲温度異常下限超え割り込み有効	
	VEXTA	OFF	外部電源電圧異常割り込み無効	○
		ON	外部電源電圧異常割り込み有効	
	SIG1	OFF	DIN1割り込み無効	○
		FALL	DIN1 OFF→ON検出割り込み有効	
		RISE	DIN1 ON→OFF検出割り込み有効	
	SIG2	OFF	DIN2割り込み無効	○
		FALL	DIN2 OFF→ON検出割り込み有効	
		RISE	DIN2 ON→OFF検出割り込み有効	
	SIG3	OFF	DIN3 割り込み無効	○
		FALL	DIN3 OFF→ON検出割り込み有効	
		RISE	DIN3 ON→OFF検出割り込み有効	
	SIG4	OFF	DIN4割り込み無効	○
		FALL	DIN4 OFF→ON検出割り込み有効	
		RISE	DIN4 ON→OFF検出割り込み有効	
	SIG5	OFF	DIN5割り込み無効	○
		FALL	DIN5 OFF→ON検出割り込み有効	
		RISE	DIN5 ON→OFF検出割り込み有効	
	SIG6	OFF	DIN6割り込み無効	○
		FALL	DIN6 OFF→ON検出割り込み有効	
		RISE	DIN6 ON→OFF検出割り込み有効	
	SIG7	OFF	DIN7割り込み無効	○
		FALL	DIN7 OFF→ON検出割り込み有効	
		RISE	DIN7 ON→OFF検出割り込み有効	
	SIG8	OFF	DIN8割り込み無効	○
		FALL	DIN8 OFF→ON検出割り込み有効	
		RISE	DIN8 ON→OFF検出割り込み有効	
	CPTHSA	OFF	CPUコア温度警告上限超え割り込み無効	○
		ON	CPUコア温度警告上限超え割り込み有効	
	CPTLSA	OFF	CPUコア温度警告下限超え割り込み無効	○
		ON	CPUコア温度警告下限超え割り込み有効	
	LCTHSA	OFF	CPU基板温度警告上限超え割り込み無効	○

		ON	CPU基板温度警告上限超過割り込み有効	
	LCTLSA	OFF	CPU基板温度警告下限超過割り込み無効	○
		ON	CPU基板温度警告下限超過割り込み有効	
	VSSEX	OFF	X軸振動・衝撃検出割り込み無効	○
		ON	X軸振動・衝撃検出割り込み有効	
	VSSEY	OFF	Y軸振動・衝撃検出割り込み無効	○
		ON	Y軸振動・衝撃検出割り込み有効	
	VSSEZ	OFF	Z軸振動・衝撃検出割り込み無効	○
		ON	Z軸振動・衝撃検出割り込み有効	
	FAN1ST	OFF	FAN1異常検出割り込み無効	○
		ON	FAN1異常検出割り込み有効	
	FAN2ST	OFF	FAN2異常検出割り込み無効	○
		ON	FAN2異常検出割り込み有効	
	FAN3ST	OFF	FAN3異常検出割り込み無効	○
		ON	FAN3異常検出割り込み有効	
	RAIDSE	OFF	RAID異常検出割り込み無効	○
		ON	RAID異常検出割り込み有効	
	HIMOSW	OFF	ひもスイッチ割り込み無効	○
		ON	ひもスイッチ割り込み有効	
	PSMP1	OFF	P1スロットイン電源モジュール割り込み無効	○
		FAL	P1スロットイン電源モジュール異常(FAL信号検出)割り込み有効	
		DEG	P1スロットイン電源モジュール警告(DEG信号検出)割り込み有効	
	PSMP2	OFF	P2スロットイン電源モジュール割り込み無効	○
		FAL	P2スロットイン電源モジュール異常(FAL信号検出)割り込み有効	
		DEG	P2スロットイン電源モジュール警告(DEG信号検出)割り込み有効	
	PSMP3	OFF	P3スロットイン電源モジュール割り込み無効	○
		FAL	P3スロットイン電源モジュール異常(FAL信号検出)割り込み有効	
		DEG	P3スロットイン電源モジュール警告(DEG信号検出)割り込み有効	
	PSMP4	OFF	P4スロットイン電源モジュール割り込み無効	○
		FAL	P4スロットイン電源モジュール異常(FAL信号検出)割り込み有効	
		DEG	P4スロットイン電源モジュール警告(DEG信号検出)割り込み有効	
	PSMP5	OFF	P5スロットイン電源モジュール割り込み無効	○
		FAL	P5スロットイン電源モジュール異常(FAL信号検出)割り込み有効	
		DEG	P5スロットイン電源モジュール警告(DEG信号検出)割り込み有効	
	PSMP6	OFF	P6スロットイン電源モジュール割り込み無効	○
		FAL	P6スロットイン電源モジュール異常(FAL信号検出)割り込み有効	
		DEG	P6スロットイン電源モジュール警告(DEG信号検出)割り込み有効	
使用関数	本コマンドは、システム監視ソフトウェア (DPG-0226/0200) の、CmSetEventMaskEx、CmSetEventMask関数を使用しています。			
使用例	『2.7 割り込みイベント』を参照してください。			

2.ON CM CALL

命令			
機能	指定したシステム監視要因発生時に、呼び出されるサブルーチンを定義します。 本命令の呼び出し後、CM ONを呼び出すと割り込み有効になります。		
書式	ON CM CALL <①サブルーチン名>		
パラメータ	①	<サブルーチン名>	サブルーチン名
	処理ルーチンのサブルーチン名で指定します。 サブルーチンは、以下の定義に従います。		
	<div><div>SUB サブルーチン名</div><div>処理内容</div><div>END SUB</div></div>		
備考	<ul style="list-style-type: none">・本コマンドは、デバッグトレース実行中には使用できません。・CM ONでエラー検出を有効にしている間は、本命令を使用することはできません。一度CM OFFで無効にしてから使用してください。・本命令を呼び出し後、CM ON で割り込みが有効になります。		
使用関数	<ul style="list-style-type: none">・本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmSetEvent関数を使用しています。		
使用例	『2.7 割り込みイベント』を参照してください。		

3.CM ON

命令	
機 能	システム監視割り込みを有効にします。
書 式	CM ON
備 考	<ul style="list-style-type: none">・割り込みを有効にし、保留にしていた割り込みを発生させます。・本コマンドの実行後、CMEVENTCONFIG命令で有効化した割り込み要因が発生すると、ON CM CALL命令の実行時に指定されたサブルーチンが実行されます。
使用関数	・本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmSetEvent関数を使用しています。
使用例	『2.7 割り込みイベント』を参照してください。

4.CM OFF

命令	
機 能	システム監視割り込みを無効にします。
書 式	CM OFF
備 考	・本コマンドの呼び出し以降、割り込みが検知されても、サブルーチンへの分岐は起こりません。
使用関数	・本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmKillEvent関数を使用しています。
使用例	『2.7 割り込みイベント』を参照してください。

5.CM STOP

命令	
機 能	システム監視割り込みを保留にします。
書 式	CM STOP
備 考	<ul style="list-style-type: none">・本コマンドの呼び出し以降、割り込みが保留されます。サブルーチンへの分岐は起こりません。・CM ONコマンドで割り込みが許可されると、保留されていた割り込みでサブルーチンに分岐します。
使用例	『2.7 割り込みイベント』を参照してください。

6.CMEVENT

関数			
機 能 書 式		発生したシステム監視割り込み要因を取得します。	
		<(戻り値)発生要因> = CMEVENT	
戻り値	戻り値	<発生要因>	数値
	発生要因が返ります。		
■ 対応表			
	識別子	設定値	内容
	CMEVTWDT	1	WDT周期満了割り込み検出
	CMEVTCPTHSA	2	CPUコア温度異常上限超過割り込み検出
	CMEVTCPTLSA	3	CPUコア温度異常下限超過割り込み検出
	CMEVTLCTHSA	4	CPU基板温度異常上限超過割り込み検出
	CMEVTLCTLSA	5	CPU基板温度異常下限超過割り込み検出
	CMEVTVEXTA	6	外部電源電圧異常割り込み検出
	CMEVTSIG1	7	DIN1割り込み検出
	CMEVTSIG2	8	DIN2割り込み検出
	CMEVTSIG3	9	DIN3割り込み検出
	CMEVTSIG4	10	DIN4割り込み検出
	CMEVTCPTHSA	11	CPUコア温度警告上限超過割り込み検出
	CMEVTCPTLSA	12	CPUコア温度警告下限超過割り込み検出
	CMEVTLCTHSA	13	CPU基板温度警告上限超過割り込み検出
	CMEVTLCTLSA	14	CPU基板温度警告下限超過割り込み検出
	CMEVTVSSEX	15	X軸振動・衝撃検出割り込み検出
	CMEVTVSSEY	16	Y軸振動・衝撃検出割り込み検出
	CMEVTVSSEZ	17	Z軸振動・衝撃検出割り込み検出
	CMEVTFAN1ST	18	FAN1異常検出割り込み検出
	CMEVTRAIDSE	19	RAID異常検出割り込み検出
	CMEVTHIMOSW	20	ひもスイッチ割り込み検出
	CMEVTSIG5	21	DIN5割り込み検出
	CMEVTSIG6	22	DIN6割り込み検出
	CMEVTSIG7	23	DIN7割り込み検出
	CMEVTSIG8	24	DIN8割り込み検出
	CMEVTFAN2ST	25	FAN2異常検出割り込み検出
	CMEVTFAN3ST	26	FAN3異常検出割り込み検出
	CMEVTMETHS	27	メモリ周囲温度異常上限超過割り込み検出
	CMEVTMETLS	28	メモリ周囲温度異常下限超過割り込み検出
	CMEVTPSMP1FAL	29	P1スロットイン電源モジュールFAL信号割り込み検出
	CMEVTPSMP2FAL	30	P2スロットイン電源モジュールFAL信号割り込み検出
	CMEVTPSMP3FAL	31	P3スロットイン電源モジュールFAL信号割り込み検出
	CMEVTPSMP4FAL	32	P4スロットイン電源モジュールFAL信号割り込み検出
	CMEVTPSMP5FAL	33	P5スロットイン電源モジュールFAL信号割り込み検出
	CMEVTPSMP6FAL	34	P6スロットイン電源モジュールFAL信号割り込み検出
	CMEVTPSMP1DEG	35	P1スロットイン電源モジュールDEG信号割り込み検出
	CMEVTPSMP2DEG	36	P2スロットイン電源モジュールDEG信号割り込み検出
	CMEVTPSMP3DEG	37	P3スロットイン電源モジュールDEG信号割り込み検出
	CMEVTPSMP4DEG	38	P4スロットイン電源モジュールDEG信号割り込み検出
	CMEVTPSMP5DEG	39	P5スロットイン電源モジュールDEG信号割り込み検出
	CMEVTPSMP6DEG	40	P6スロットイン電源モジュールDEG信号割り込み検出
	識別子は、SML001.AJN にて、定義されています。		
備 考	・ 本コマンドを使用する際は、INCLUDEコマンドでSML001.AJNをインクルードしてください。 ・ 割り込み要因を取得すると、要因はクリアされます(本コマンドを呼び出して要因を取得した後、何も要因が満たされていない状態で再度本コマンドを呼び出すと、2回目は要因として0が返されます)。		
使用関数	・ 本コマンドは、システム監視ソフトウェア (DPG-0226/0200) の、CmGetEventFactor, CmGetEventFactorEx関数を使用しています。		

3.9 詳細個別説明（瞬低対策用電源監視）

1.BATSTATUS

関数

機 能	バッテリーステータスを取得します。																		
書 式	<(戻り値)ステータス>=BATSTATUS																		
戻り値	戻り値	<ステータス>																	
	数値																		
	バッテリーステータスを取得します。																		
	各ビットは、以下のように定義されています。																		
	<table><tr><td colspan="8">bit31 ~ bit16</td></tr><tr><td colspan="8">予約</td></tr></table>			bit31 ~ bit16								予約							
	bit31 ~ bit16																		
	予約																		
	<table><tr><td colspan="4">bit15 ~ bit10</td><td colspan="2">bit9</td><td colspan="2">bit8</td></tr><tr><td colspan="4">予約</td><td colspan="2">BAT2</td><td colspan="2">BAT1</td></tr></table>			bit15 ~ bit10				bit9		bit8		予約				BAT2		BAT1	
	bit15 ~ bit10				bit9		bit8												
	予約				BAT2		BAT1												
<table><tr><td>bit7</td><td>bit 6</td><td>bit 5</td><td>bit 4</td><td>bit 3</td><td>bit 2</td><td>bit 1</td><td>bit 0</td></tr><tr><td>ERR</td><td>BLALM</td><td>LALM</td><td>DCHG</td><td>CHG</td><td>RSOC2</td><td>RSOC1</td><td>RSOC0</td></tr></table>			bit7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	ERR	BLALM	LALM	DCHG	CHG	RSOC2	RSOC1	RSOC0	
bit7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0												
ERR	BLALM	LALM	DCHG	CHG	RSOC2	RSOC1	RSOC0												
BAT2 : バッテリーの接続状態																			
BAT1 : バッテリーが複数接続されるモデルでは以下のように接続状態を取得 できます。																			
<table><tr><td>BAT2</td><td>BAT1</td><td></td></tr><tr><td>1</td><td>1</td><td>正常</td></tr><tr><td>1</td><td>0</td><td>接続異常</td></tr><tr><td>0</td><td>1</td><td>接続異常</td></tr><tr><td>0</td><td>0</td><td>接続なし</td></tr></table>			BAT2	BAT1		1	1	正常	1	0	接続異常	0	1	接続異常	0	0	接続なし		
BAT2	BAT1																		
1	1	正常																	
1	0	接続異常																	
0	1	接続異常																	
0	0	接続なし																	
※接続なしの場合、「3.21 エラーコード一覧(P.67)」のERRSUB関数で得 られる値で、&HC0000009 が返されます。																			
※接続可能なバッテリーが一つの製品ではBAT1のみ有効です。BAT2は無視し て下さい。																			
※接続なし、接続異常が発生した場合、弊社サポート窓口までお問い合わせせ 下さい。																			
ERR : バッテリー異常																			
※バッテリー異常が発生した場合、弊社サポート窓口までお問い合わせ下さ い。																			
0 : エラーなし																			
1 : バッテリー異常が発生																			
BLALM : バッテリー残容量低下アラーム																			
0 : 正常																			
1 : バッテリーの残容量が20%以下																			
LALM : 寿命アラーム																			
※寿命アラームが発生した場合、バッテリーモジュール交換時期です。																			
0 : 正常																			
1 : 寿命アラームが発生																			
DCHG : 放電情報																			
0 : 放電していない																			
1 : 放電中																			
CHG : 充電情報																			
0 : 充電していない																			
1 : 充電中																			
RSOC2-0 : バッテリー残容量																			
111 : 100%~80%																			

システム監視制御コマンドリファレンス

	<p>011 : 79%～30% 001 : 29%～1% 000 : 0%</p> <p>JPS-BUSTC / JPS-BUETCA / JPS-BUETC2 / JPS-BUVAC / JPS-BUVAC2 / JPS-BUC1308 / JPS-BUC1316 の場合</p> <p>BLALM : バッテリ残容量低下アラーム 0 : 正常 1 : バッテリの残容量が30%以下</p> <p>RSOC2-1 : バッテリ残容量 11 : 100%～80% 01 : 79%～30% 00 : 29%～0%</p> <p>RSOC0 : 外部バッテリー接続状態 (外部電池入力端子(BU_IN2)接続状態) 外部電池を使用しない場合は無視してください。 1 : 接続 0 : 未接続</p> <p>※JPS-BUSTC / JPS-BUETCA のみ取得できます。</p>
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmGetBatStatus関数を使用しています。
使用例	『2.8 瞬低対策用電源監視』を参照してください。

2.BATPOWEROFFTIME

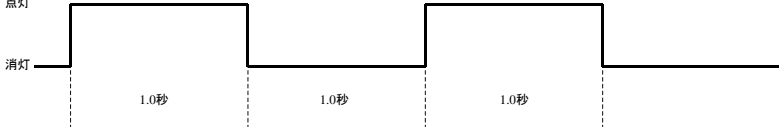
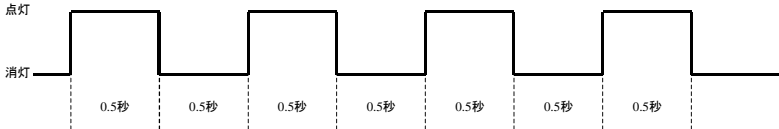
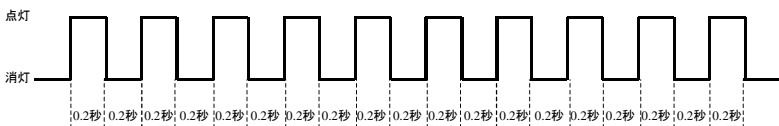
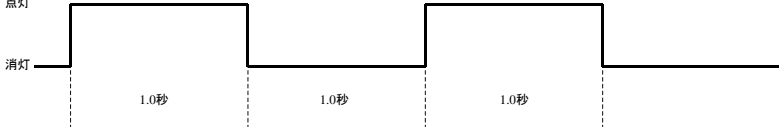
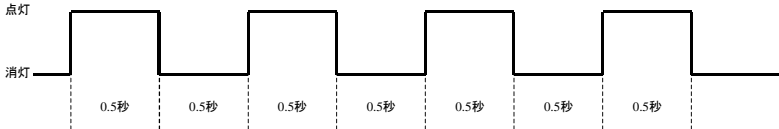
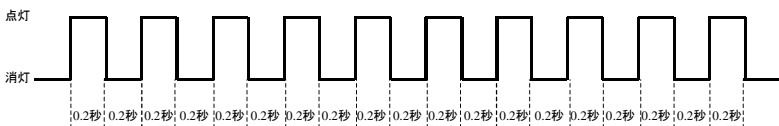
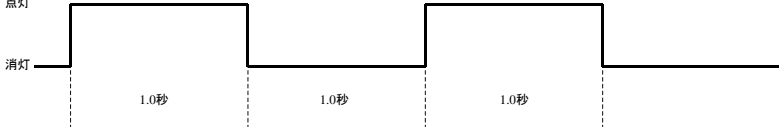
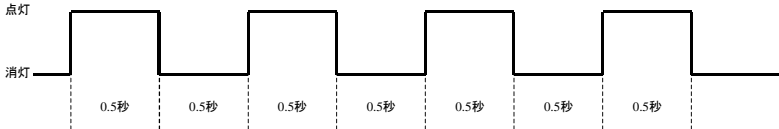
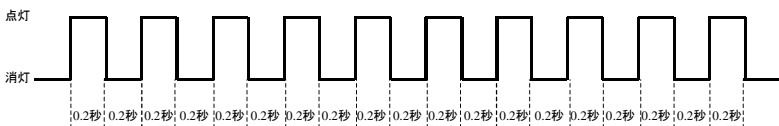
命令				
機 能		停電発生からシャットダウン/電源強制断までの時間を設定します。		
書 式		BATPOWEROFFTIME <①制御時間>, <②時間>		
パラ メータ		①	<制御時間>	数値
		停電発生からシャットダウン開始までの時間を設定します。 設定できる範囲は0～255で、単位は秒です。 デフォルトは16秒です。 0を指定すると、シャットダウンを行いません。		
		②	<時間>	数値
		シャットダウン開始から電源強制断までの時間を設定します。 設定できる範囲は0～255で、単位は秒です。 デフォルトは180秒です。 0を指定すると、電源強制断を行いません。		
備 考		・ バッテリモジュール接続時のみ使用できます。 ・ 停電後に本関数を実行すると、新しく設定した時間秒後にシャットダウンを開始します。 ・ 複数の異なるアプリケーションで本関数を実行した場合、最後に設定した時間が有効となります。 ・ OS 再起動、およびシャットダウン時でも、電源が待機状態であれば、設定は保持されます。 ・ バッテリモジュールはオプションです。 詳しくは弊社サポート窓口までお問い合わせ下さい。		
使用関数		・ 本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmSetPowerOffTime関数を使用しています。		
使用例		『2.8 瞬低対策用電源監視』を参照してください。		

3.10 詳細個別説明 (LED制御)

1.LEDCONTROL

命令		
機 能	LEDを制御します。 CMLED命令の別版です。	
書 式	LEDCONTROL <①制御>	
パラ メータ	①	<制御>
	文字列	
	LED制御を指定します。 「識別子=設定値」の形式で、文字列で指定します。複数列挙する場合は「,」で区切ります。	
	識別子	設定値 内容
	ALMLED	ON アラームLEDをON
		OFF アラームLEDをOFF
	ERRLED	ON エラーLEDをON
		OFF エラーLEDをOFF
	LED1	ON LED1をON
		OFF LED1をOFF
	LED2	ON LED2をON
		OFF LED2をOFF
	LED3	ON LED3をON
		OFF LED3をOFF
	LED4	ON LED4をON
		OFF LED4をOFF
	LED5	ON LED5をON
		OFF LED5をOFF
	LED6	ON LED6をON
		OFF LED6をOFF
	LED7	ON LED7をON
		OFF LED7をOFF
	LED8	ON LED8をON
		OFF LED8をOFF
	LED9	ON LED9をON
		OFF LED9をOFF
	LED10	ON LED10をON
		OFF LED10をOFF
使用関数	本コマンドは、システム監視ソフトウェア (DPG-0226) の、CmGetControlStatus、CmControlNoticeLed、CmControlGPLed関数を使用しています。	
使用例	『2.9 LED制御』を参照してください。	

2.LEDPATTERN

命令																																						
機 能	LEDの点灯パターンを設定します。 CMLED命令の別版です。																																					
書 式	LEDPATTERN <①制御>																																					
パラ メータ	①	<制御>文字列																																				
	点灯パターンを設定します。 「識別子=設定値」の形式で、文字列で指定します。複数列挙する場合は「,」で区切ります。																																					
	<table><tr><th>識別子</th><th>設定値</th><th>内容</th></tr><tr><td rowspan="4">ALMLED</td><td>PATTERN1</td><td>アラームLEDに対して点灯パターン1</td></tr><tr><td>PATTERN2</td><td>アラームLEDに対して点灯パターン2</td></tr><tr><td>PATTERN3</td><td>アラームLEDに対して点灯パターン3</td></tr><tr><td>PATTERN4</td><td>アラームLEDに対して点灯パターン4</td></tr><tr><td rowspan="4">ERRLED</td><td>PATTERN1</td><td>エラーLEDに対して点灯パターン1</td></tr><tr><td>PATTERN2</td><td>エラーLEDに対して点灯パターン2</td></tr><tr><td>PATTERN3</td><td>エラーLEDに対して点灯パターン3</td></tr><tr><td>PATTERN4</td><td>エラーLEDに対して点灯パターン4</td></tr><tr><td rowspan="7">ERRLEDCOLOR</td><td>RED</td><td>発光色を RED に</td></tr><tr><td>GREEN</td><td>発光色を GREEN に</td></tr><tr><td>YELLOW</td><td>発光色を YELLOW に</td></tr><tr><td>BLUE</td><td>発光色を BLUE に</td></tr><tr><td>MAGENTA</td><td>発光色を MAGENTA に</td></tr><tr><td>CYAN</td><td>発光色を CYAN に</td></tr><tr><td>WHITE</td><td>発光色を WHITE に</td></tr></table>		識別子	設定値	内容	ALMLED	PATTERN1	アラームLEDに対して点灯パターン1	PATTERN2	アラームLEDに対して点灯パターン2	PATTERN3	アラームLEDに対して点灯パターン3	PATTERN4	アラームLEDに対して点灯パターン4	ERRLED	PATTERN1	エラーLEDに対して点灯パターン1	PATTERN2	エラーLEDに対して点灯パターン2	PATTERN3	エラーLEDに対して点灯パターン3	PATTERN4	エラーLEDに対して点灯パターン4	ERRLEDCOLOR	RED	発光色を RED に	GREEN	発光色を GREEN に	YELLOW	発光色を YELLOW に	BLUE	発光色を BLUE に	MAGENTA	発光色を MAGENTA に	CYAN	発光色を CYAN に	WHITE	発光色を WHITE に
	識別子	設定値	内容																																			
	ALMLED	PATTERN1	アラームLEDに対して点灯パターン1																																			
PATTERN2		アラームLEDに対して点灯パターン2																																				
PATTERN3		アラームLEDに対して点灯パターン3																																				
PATTERN4		アラームLEDに対して点灯パターン4																																				
ERRLED	PATTERN1	エラーLEDに対して点灯パターン1																																				
	PATTERN2	エラーLEDに対して点灯パターン2																																				
	PATTERN3	エラーLEDに対して点灯パターン3																																				
	PATTERN4	エラーLEDに対して点灯パターン4																																				
ERRLEDCOLOR	RED	発光色を RED に																																				
	GREEN	発光色を GREEN に																																				
	YELLOW	発光色を YELLOW に																																				
	BLUE	発光色を BLUE に																																				
	MAGENTA	発光色を MAGENTA に																																				
	CYAN	発光色を CYAN に																																				
	WHITE	発光色を WHITE に																																				
点灯パターンは、下記の通りです。																																						
<table><tr><th>パターン</th><th>LEDの動作</th></tr><tr><td>1</td><td>常時点灯します。</td></tr><tr><td>2</td><td>1秒ごとに点灯と消灯を繰り返します。 <div><div>点灯</div><div>消灯</div><div></div></div></td></tr><tr><td>3</td><td>0.5秒ごとに点灯と消灯を繰り返します。 <div><div>点灯</div><div>消灯</div><div></div></div></td></tr><tr><td>4</td><td>0.2秒ごとに点灯と消灯を繰り返します。 <div><div>点灯</div><div>消灯</div><div></div></div></td></tr></table>		パターン	LEDの動作	1	常時点灯します。	2	1秒ごとに点灯と消灯を繰り返します。 <div><div>点灯</div><div>消灯</div><div></div></div>	3	0.5秒ごとに点灯と消灯を繰り返します。 <div><div>点灯</div><div>消灯</div><div></div></div>	4	0.2秒ごとに点灯と消灯を繰り返します。 <div><div>点灯</div><div>消灯</div><div></div></div>																											
パターン	LEDの動作																																					
1	常時点灯します。																																					
2	1秒ごとに点灯と消灯を繰り返します。 <div><div>点灯</div><div>消灯</div><div></div></div>																																					
3	0.5秒ごとに点灯と消灯を繰り返します。 <div><div>点灯</div><div>消灯</div><div></div></div>																																					
4	0.2秒ごとに点灯と消灯を繰り返します。 <div><div>点灯</div><div>消灯</div><div></div></div>																																					
備 考	ERRLEDCOLORによる発行色設定は、多色設定機能付きの製品(有償オプション)でのみ使用可能です。																																					
使用関数	本コマンドは、システム監視ソフトウェア (DPG-0226) の、CmSetPatternNoticeLed、CmSetColorNoticeLed関数を使用しています。																																					
使用例	『2.9 LED制御』を参照してください。																																					

3.CMLED

命令																									
機 能	LEDを制御します。 LEDCONTROL, LEDPATTERN命令の別版です。																								
書 式	CMLED <①エラーLED制御> [, <②アラームLED制御> [, <③汎用LED制御>]]																								
パラ メータ	①	<エラーLED制御> 数値																							
	エラーLEDのON / OFF および、多色指定可能な製品の場合は発光色を制御します。 LEDをONにしたとき、常時点灯です。																								
	多色機能が無い製品の場合：																								
	<table><tr><th>設定値</th><th>内容</th></tr><tr><td>0</td><td>エラーLEDをOFF</td></tr><tr><td>0以外</td><td>エラーLEDをON</td></tr></table>		設定値	内容	0	エラーLEDをOFF	0以外	エラーLEDをON																	
	設定値	内容																							
	0	エラーLEDをOFF																							
	0以外	エラーLEDをON																							
	多色機能が有る製品の場合：																								
	<table><tr><th>設定値</th><th>内容</th></tr><tr><td>0</td><td>エラーLEDをOFF</td></tr><tr><td>1</td><td>エラーLEDをONにし、発光色をRED</td></tr><tr><td>2</td><td>エラーLEDをONにし、発光色をGREEN</td></tr><tr><td>3</td><td>エラーLEDをONにし、発光色をYELLOW</td></tr><tr><td>4</td><td>エラーLEDをONにし、発光色をBLUE</td></tr><tr><td>5</td><td>エラーLEDをONにし、発光色をMAGENTA</td></tr><tr><td>6</td><td>エラーLEDをONにし、発光色をCYAN</td></tr><tr><td>7</td><td>エラーLEDをONにし、発光色をWHITE</td></tr></table>		設定値	内容	0	エラーLEDをOFF	1	エラーLEDをONにし、発光色をRED	2	エラーLEDをONにし、発光色をGREEN	3	エラーLEDをONにし、発光色をYELLOW	4	エラーLEDをONにし、発光色をBLUE	5	エラーLEDをONにし、発光色をMAGENTA	6	エラーLEDをONにし、発光色をCYAN	7	エラーLEDをONにし、発光色をWHITE					
	設定値	内容																							
0	エラーLEDをOFF																								
1	エラーLEDをONにし、発光色をRED																								
2	エラーLEDをONにし、発光色をGREEN																								
3	エラーLEDをONにし、発光色をYELLOW																								
4	エラーLEDをONにし、発光色をBLUE																								
5	エラーLEDをONにし、発光色をMAGENTA																								
6	エラーLEDをONにし、発光色をCYAN																								
7	エラーLEDをONにし、発光色をWHITE																								
②	<アラームLED制御> 真偽値																								
アラームLEDのON(TRUE) / OFF(FALSE)を制御します。 LEDをONにしたとき、常時点灯です。省略時、アラームLEDはOFFです。																									
<table><tr><th>設定値</th><th>内容</th></tr><tr><td>FALSE</td><td>アラームLEDをOFF</td></tr><tr><td>TRUE</td><td>アラームLEDをON</td></tr></table>		設定値	内容	FALSE	アラームLEDをOFF	TRUE	アラームLEDをON																		
設定値	内容																								
FALSE	アラームLEDをOFF																								
TRUE	アラームLEDをON																								
③	<汎用LED制御> 数値																								
汎用LEDの各点のON / OFF を制御します。 各設定値は、OR 可能です。省略時、汎用LEDは全点OFFです。																									
<table><tr><th>設定値</th><th>内容</th></tr><tr><td>0</td><td>汎用LEDはOFF</td></tr><tr><td>&H1</td><td>汎用LED1をON</td></tr><tr><td>&H2</td><td>汎用LED2をON</td></tr><tr><td>&H4</td><td>汎用LED3をON</td></tr><tr><td>&H8</td><td>汎用LED4をON</td></tr><tr><td>&H10</td><td>汎用LED5をON</td></tr><tr><td>&H20</td><td>汎用LED6をON</td></tr><tr><td>&H40</td><td>汎用LED7をON</td></tr><tr><td>&H80</td><td>汎用LED8をON</td></tr><tr><td>&H100</td><td>汎用LED9をON</td></tr><tr><td>&H200</td><td>汎用LED10をON</td></tr></table>		設定値	内容	0	汎用LEDはOFF	&H1	汎用LED1をON	&H2	汎用LED2をON	&H4	汎用LED3をON	&H8	汎用LED4をON	&H10	汎用LED5をON	&H20	汎用LED6をON	&H40	汎用LED7をON	&H80	汎用LED8をON	&H100	汎用LED9をON	&H200	汎用LED10をON
設定値	内容																								
0	汎用LEDはOFF																								
&H1	汎用LED1をON																								
&H2	汎用LED2をON																								
&H4	汎用LED3をON																								
&H8	汎用LED4をON																								
&H10	汎用LED5をON																								
&H20	汎用LED6をON																								
&H40	汎用LED7をON																								
&H80	汎用LED8をON																								
&H100	汎用LED9をON																								
&H200	汎用LED10をON																								
備 考	多色設定機能は有償オプションです。ご利用されたい方は、多色設定機能付きの製品をお買い求めください。																								
使用関数	本コマンドは、システム監視ソフトウェア (DPG-0226) の、CmGetControlStatus、CmControlNoticeLed、CmControlGPLed、CmSetColorNoticeLed、CmGetColorNoticeLed関数を使用しています。																								
使用例	『2.9 LED制御』を参照してください。																								

3.11 詳細個別説明（ブザー制御）

1.BUZZERCONTROL

命令										
機 能	ブザーを制御します。 CMBUZZER命令の別版です。									
書 式	BUZZERCONTROL <①制御値>									
パラ メータ	①	<制御値> 文字列								
	ブザーの制御値を指定します。 「識別子=設定値」の形式で、文字列で指定します。									
	<table><tr><th>識別子</th><th>設定値</th><th>内容</th></tr><tr><td rowspan="2">CONTROL</td><td>ON</td><td>ブザーをON</td></tr><tr><td>OFF</td><td>ブザーをOFF</td></tr></table>		識別子	設定値	内容	CONTROL	ON	ブザーをON	OFF	ブザーをOFF
	識別子	設定値	内容							
CONTROL	ON	ブザーをON								
	OFF	ブザーをOFF								
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmControlBuzzer関数を使用しています。									
備 考	呼び出しが簡素な、CMBUZZER命令が推奨されます。									
使用例	『2.10 ブザー制御』を参照してください。									

2.CMBUZZER

命令							
機 能	ブザーを制御します。 BUZZERCONTROL命令の別版です。						
書 式	CMBUZZER <①ブザー指定>						
パラ メータ	①	<ブザー指定> 真偽値					
	ブザーのON(TRUE) / OFF(FALSE)を指定します。						
	<table><tr><th>設定値</th><th>内容</th></tr><tr><td>TRUE</td><td>ブザーをON</td></tr><tr><td>FALSE</td><td>ブザーをOFF</td></tr></table>		設定値	内容	TRUE	ブザーをON	FALSE
設定値	内容						
TRUE	ブザーをON						
FALSE	ブザーをOFF						
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmControlBuzzer関数を使用しています。						
使用例	『2.10 ブザー制御』を参照してください。						

3.12 詳細個別説明（ひもスイッチ）

1.HIMOSWITCHSTATUS

関数			
機 能	現在のひもスイッチの状態を取得します。		
書 式	<(戻り値)ステータス>= HIMOSWITCHSTATUS		
戻り値	戻り値	<ステータス>	数値
	ひもスイッチの状態の格納先です。 0：ひもスイッチを引いていない状態です。 1：ひもスイッチを引いている状態です。		
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmGetHimoSwitchStatus関数を使用しています。		
使用例	『2.11 ひもスイッチ』を参照してください。		

2.HIMOSWITCHCONFIG

命令										
機 能	ひもスイッチの機能を設定します。									
書 式	HIMOSWITCHCONFIG <①機能>									
パラ メータ	①	<機能>	文字列							
	割り当てる機能を設定します。									
	「識別子=設定値」の形式で、文字列で指定します。									
	<table><tr><th>識別子</th><th>設定値</th><th>内容</th></tr><tr><td rowspan="2">MODE</td><td>GSWITCH</td><td>汎用スイッチとして使用</td></tr><tr><td>PSWITCH</td><td>電源スイッチとして使用</td></tr></table>			識別子	設定値	内容	MODE	GSWITCH	汎用スイッチとして使用	PSWITCH
識別子	設定値	内容								
MODE	GSWITCH	汎用スイッチとして使用								
	PSWITCH	電源スイッチとして使用								
備 考	・ 割り込み通知を使用する場合は汎用スイッチに設定してください。 ・ 設定は次回の起動時から有効になります。									
使用関数	・ 本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmSetHimoSwitchConfig関数を使用しています。									
使用例	『2.11 ひもスイッチ』を参照してください。									

3.13 詳細個別説明（自己診断）

1.RASERRORSTATUS

関数	
機 能	現在発生中の異常状態を取得します。
書 式	<(戻り値)ステータス> = RASERRORSTATUS
戻り値	戻り値

2.RASCONTROLSTATUS

関数	
機 能	現在の制御状態を取得します。
書 式	<(戻り値)ステータス> = RASCONTROLSTATUS
戻り値	戻り値

3.14 詳細個別説明 (FAN監視)

1.FANROTATEGET

関数			
機 能	FAN の回転数を取得します。 回転数は毎分の回転数(rpm)が取得できます。		
書 式	<(戻り値)回転数> = FANROTATEGET(<①対象>)		
戻り値	戻り値	<回転数>	数値
指定した対象の回転数を取得します。(単位 : rpm)			
パラ メータ	①	<対象>	数値
	回転数を取得する対象を指定します。		
	設定値	内容	
	1	FAN1の回転数を取得します	
	2	FAN2の回転数を取得します	
	3	FAN3の回転数を取得します	
備 考	ご利用の製品によっては指定できないFAN番号があります。		
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmGetFanRotate関数を使用しています。		
使用例	『2.12 FAN監視』を参照してください。		

2.FANAUTOTHRESHOLD

命令			
機 能	FAN 自動制御の温度閾値を設定します。 温度閾値はCPUコア温度の閾値となります。		
書 式	FANAUTOTHRESHOLD <①対象>, <②低温閾値>, <③高温閾値>		
パラ メータ	①	<対象>	数値
	温度閾値を設定する対象を指定します。		
	設定値	内容	
	1	FAN1	
	2	FAN2	
	3	FAN3	
	②	<低温閾値>	数値
	低温閾値を設定します。(単位 : °C) -60～高温閾値の値を設定できます。		
	③	<高温閾値>	数値
	高温閾値を設定します。(単位 : °C) 低温閾値～125の値を設定できます。		
備 考	ご利用の製品によっては指定できないFAN番号があります。		
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmSetFanAutoThreshold関数を使用しています。		
使用例	『2.12 FAN監視』を参照してください。		

3.FANAUTOSPEED

命令		
機 能	FAN自動制御の速度を%で設定します。	
書 式	FANAUTOSPEED <①対象>,<②低温速度>,<③正常速度>,<④高温速度>	
パラ メータ	①	<対象> 数値
	速度を設定する対象を指定します。	
	設定値	内容
	1	FAN1の自動制御の速度を設定します。
	2	FAN2の自動制御の速度を設定します。
	3	FAN3の自動制御の速度を設定します。
	②	<低温速度> 数値
低温閾値以下の場合の速度を設定します。（単位：%） 30～100の値を10刻みで設定できます。一桁目の数は切り捨てられます。		
③	<正常速度> 数値	
低温閾値～高温閾値間の場合の速度を設定します。（単位：%） 30～100の値を10刻みで設定できます。一桁目の数は切り捨てられます。		
④	<高温速度> 数値	
高温閾値～最大閾値間の場合の速度を設定します。（単位：%） 30～100の値を10刻みで設定できます。一桁目の数は切り捨てられます。		
備 考	ご利用の製品によっては指定できないFAN番号があります。	
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmSetFanAutoSpeed関数を使用しています。	
使用例	『2.12 FAN監視』を参照してください。	

3.15 詳細個別説明（振動・衝撃監視）

1.GSENSEGET

関数														
機 能	指定対象の加速度センサ値を取得します。													
書 式	<(戻り値)加速度センサ値> = GSENSEGET(<①対象>)													
戻り値	戻り値	<加速度センサ値>												
	数値													
	<p>加速度センサ値を取得します。 ここで得られる値は加速度センサの生の値となります。 下記の式で重力加速度値へ変換できます。</p> $G = \frac{(x \times 0.01289) - 1.5}{0.42}$ <p>ここで、 G: 重力加速度値 x: 取得値</p>													
パラ メータ	①	<対象>												
	<p>文字列</p> <p>加速度センサ値を取得する、加速度センサと軸を指定します。 「識別子=設定値」の形式で、文字列で指定します。</p> <table border="1"> <thead> <tr> <th>識別子</th><th>設定値</th><th>内容</th></tr> </thead> <tbody> <tr> <td>TARGET</td><td>SENSOR1</td><td>加速度センサ1のセンサ値を取得します。</td></tr> <tr> <td rowspan="3">AXIS</td><td>X</td><td>X軸方向のセンサ値を取得します。</td></tr> <tr> <td>Y</td><td>Y軸方向のセンサ値を取得します。</td></tr> <tr> <td>Z</td><td>Z軸方向のセンサ値を取得します。</td></tr> </tbody> </table>		識別子	設定値	内容	TARGET	SENSOR1	加速度センサ1のセンサ値を取得します。	AXIS	X	X軸方向のセンサ値を取得します。	Y	Y軸方向のセンサ値を取得します。	Z
識別子	設定値	内容												
TARGET	SENSOR1	加速度センサ1のセンサ値を取得します。												
AXIS	X	X軸方向のセンサ値を取得します。												
	Y	Y軸方向のセンサ値を取得します。												
	Z	Z軸方向のセンサ値を取得します。												
備 考	<p>加速度センサ機能は有償オプションです。ご利用されたい方は、加速度センサの機能付きの製品をお買い求めください。</p>													
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmGetGSense関数を使用しています。													
使用例	『2.13 振動・衝撃監視』を参照してください。													

2.GSENSELIMIT

命令																
機能		振動・衝撃異常検出の閾値を設定します。														
書式		GSENSELIMIT <①対象>, <②閾値>, <③ヒステリシス>														
パラメータ	①	<対象>	文字列													
	振動・衝撃異常検出の閾値を設定する、加速度センサと軸を指定します。 「識別子=設定値」の形式で、文字列で指定します。															
	<table><tr><td>識別子</td><td>設定値</td><td>内容</td></tr><tr><td>TARGET</td><td>SENSOR1</td><td>加速度センサ1の閾値を設定します。</td></tr><tr><td rowspan="3">AXIS</td><td>X</td><td>X軸方向の閾値を設定します。</td></tr><tr><td>Y</td><td>Y軸方向の閾値を設定します。</td></tr><tr><td>Z</td><td>Z軸方向の閾値を設定します。</td></tr></table>			識別子	設定値	内容	TARGET	SENSOR1	加速度センサ1の閾値を設定します。	AXIS	X	X軸方向の閾値を設定します。	Y	Y軸方向の閾値を設定します。	Z	Z軸方向の閾値を設定します。
	識別子	設定値	内容													
	TARGET	SENSOR1	加速度センサ1の閾値を設定します。													
	AXIS	X	X軸方向の閾値を設定します。													
		Y	Y軸方向の閾値を設定します。													
		Z	Z軸方向の閾値を設定します。													
	②	<閾値>	数値													
	振動・衝撃異常検出の閾値を指定します。 閾値は設定値に 0.03(G)を掛けた値になります。 例) 10と設定した場合、閾値は10×0.03= 0.3(G)となります。															
③	<ヒステリシス>	数値														
振動・衝撃異常検出の復帰ヒステリシスを指定します。 ヒステリシスは設定値に0.03(G)を掛けた値になります。 例) 10と設定した場合、閾値は10×0.03= 0.3(G)となります。																
備考																
加速度センサ機能は有償オプションです。ご利用されたい方は、加速度センサの機能付きの製品をお買い求めください。																
使用関数		本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmSetGSenseLimit関数を使用しています。														
使用例		『2.13 振動・衝撃監視』を参照してください。														

3.16 詳細個別説明（外部コネクタ出力）

1.EXCONTROL

命令										
機 能	外部コネクタ出力(EX_SWOUT)を制御します。									
書 式	EXCONTROL <①制御値>									
パラ メータ	①	<制御値>								
	文字列									
	制御値を指定します。 「識別子=設定値」の形式で、文字列で指定します。									
	<table><tr><th>識別子</th><th>設定値</th><th>内容</th></tr><tr><td rowspan="2">EXO</td><td>ON</td><td>リレーONに設定します。</td></tr><tr><td>OFF</td><td>リレーOFFに設定します。</td></tr></table>			識別子	設定値	内容	EXO	ON	リレーONに設定します。	OFF
識別子	設定値	内容								
EXO	ON	リレーONに設定します。								
	OFF	リレーOFFに設定します。								
備 考	<ul style="list-style-type: none">・システムの電源投入時、EX_SWOUTはリレーOFFになります。・OS再起動、およびシャットダウン時でも、電源が待機状態であればEX_SWOUTの状態は保持されます。									
使用関数	<ul style="list-style-type: none">・本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmControlEXO関数を使用しています。									
使用例	' 外部コネクタ出力をリレーONに設定します。 EXCONTROL "EXO=ON"									

3.17 詳細個別説明（ストレージ監視）

1.RAIDSTATUS

関数			
機 能 書 式	RAIDステータスを取得します。		
	<(戻り値)ステータス>=RAIDSTATUS(<①対象>)		
戻り値	戻り値	<ステータス>	数値
	RAIDステータスを取得します。		
	対象に「0」を指定した場合(DPG-0226使用時)：		
	ステータス値	内容	
	0	通常動作	
	1	ドライブ1 異常	
	2	ドライブ2 異常	
	3	データ再構築（コピー）中	
	4	システム・ホルト	
	5	HDD2→HDD1へのデータ再構築（コピー）中にスキップ・リコン(※)発生	
	6	HDD1→HDD2へのデータ再構築（コピー）中にスキップ・リコン(※)発生	
	対象に「0」を指定した場合(DPG-0200使用時)：		
	ステータス値	内容	
	0	正常動作	
	1	RAID構成が破損し、RAIDボリュームが利用できない	
	2	RAID1で異常が発生し、保護が働いていない	
	3	RAID1の再構築中	
	対象に「&H100」を指定した場合(DPG-0200使用時)：		
	ステータス値	内容	
	0～99	RAID1の再構築状況を数値で取得します。(単位は%)	
	※データ再構築(コピー)中に、コピー元のドライブに読み込み異常が発生すると、RAIDボードはその異常が発生した部分をスキップして処理を継続します。この操作をスキップ・リコンと呼びます		
パラメータ	①	<対象>	数値
	RAIDステータスを取得する対象を指定します。		
	設定値	内容	
	0	(DPG-0226使用時)RAIDボードのステータス値を取得します (DPG-0200使用時)RAIDボリュームの状態を取得します。	
	&H100	(DPG-0200使用時)RAID1の再構築状況を取得します。	
備 考	・RAIDボード検出中に 本関数を実行した場合、「3.21 エラーコード一覧(P.67)」のERRSUB関数で得られる値で、&HC0000004 が返されます。 その場合、時間をおいて再実行してください。		
使用関数	・本コマンドは、システム監視ソフトウェア(DPG-0226/0200)の、CmGetRaidStatus関数を使用しています。		
使用例	『2.14 ストレージ監視』を参照してください。		

3.18 詳細個別説明（遷移出力）

1. CMTRANSCONFIG

命令

機能	遷移出力の設定を行います。 制御対象の制御タイミング別に、機能の有効／無効、出力設定、DAW設定を行います。		
書式	CMTRANSCONFIG <①制御対象>, <②制御タイミング>, <③出力制御対象>, <④出力状態>, <⑤DAW設定>		
パラメータ	①	<制御対象>	数値
	設定を行う制御対象を指定します。		
	設定値	内容	
	0	デジタル出力(DOUT)	
	1	外部コネクタ出力(EX_SWOUT)	
②	<制御タイミング>	数値	
	設定を行う制御タイミングを指定します。		
	設定値	内容	
	0	停止時	
	1	起動時	
③	<出力制御対象>	数値	
	遷移出力の出力制御(有効 / 無効)を指定します。 <制御対象> の設定値により、設定値が変わります。		
	■デジタル出力時 下記設定値を OR できます。		
	設定値 有効	設定値 無効	内容
	0	1	DOUT1
	0	2	DOUT2
	0	4	DOUT3
	0	8	DOUT4
	■外部コネクタ出力時		
	設定値 有効	設定値 無効	内容
	0	1	EX_SWOUT
④	<出力状態>	数値	
	接点別の出力状態(ON / OFF)を指定します。 <制御対象> の設定値により、設定値が変わります。		
	■デジタル出力時 下記設定値を OR できます。		
	設定値 OFF出力	設定値 ON出力	内容
	0	1	DOUT1
	0	2	DOUT2
	0	4	DOUT3
	0	8	DOUT4
	■外部コネクタ出力時		
	設定値 OFF出力	設定値 ON出力	内容
	0	1	EX SWOUT

システム監視制御コマンドリファレンス

	⑤	<div><DAW設定></div> <div>数値</div>					
	DAW(有効 / 無効)を指定します。						
	<table><tr><td>設定値</td><td>内容</td></tr><tr><td>0</td><td>DAW無効</td></tr><tr><td>1</td><td>DAW有効</td></tr></table>	設定値	内容	0	DAW無効	1	DAW有効
設定値	内容						
0	DAW無効						
1	DAW有効						
備考	・『2.17 遷移出力』を参照してください。						
使用関数	・本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmSetTransOutputConfig関数を使用しています。						
使用例	CMTRANSCONFIG 0, 0, &HF, &HF, 1 ' デジタル出力のシステム停止時に、DOUT1～4をON出力し、DAW有効とします。						

3.19 詳細個別説明（スロットイン電源モジュール監視）

1. CMPSMGET

関数			
機 能	スロットイン電源モジュールの情報を取得します。		
書 式	<(戻り値)情報値> = CMPSMGET(<①取得ID> [, <②スロット位置>])		
戻り値	戻り値	<情報値> 数値	
	取得IDで指定した、スロットイン電源モジュールの情報を取得します。 スロット位置を指定すると単一の値が、省略すると全スロットの値が配列形式で得られます。		
	取得IDに "ID" を指定した時に得られる値は、以下の通りです。		
	戻り値	型式	内容
	&H05	JPS-2057	DC+12V入力モジュール(絶縁80W)
	&H06	JPS-2055	DC+24V入力モジュール(絶縁80W)
	&H08	JPS-2101, JPS-2101N	ATX出力モジュール(100W)
	&H09	JPS-2001	AC100V/200V入力モジュール(絶縁130W)
	&H0A	JPS-2053, JPS-2053N	DC+100V入力モジュール(絶縁130W)
	&H0B	JPS-2051, JPS-2051N	DC+48V入力モジュール(絶縁130W)
&H0C	JPS-1059	DC+12-24V入力モジュール(非絶縁130W)	
&H0D	JPS-105701	DC12V電源バス専用 DC+12V入力モジュール(非絶縁50W)	
&H0E	JPS-105501 JPS-205501	DC+24V入力モジュール(非絶縁150W) DC+24V入力モジュール(絶縁150W)	
&H0F	未実装	スロットイン電源モジュールが実装されていません	
取得IDに "STATUS" を指定した時に得られる値は、以下の通りです。			
戻り値	内容		
&H01	警告(DEG信号検出)		
&H02	異常(FAL信号検出)		
パラメータ	①	<取得ID> 文字列	
	情報を取得する対象を、文字列で指定します。		
	取得ID	内容	
	ID	スロットイン電源モジュールのIDを取得します	
	STATUS	スロットイン電源モジュールの状態を取得します	
	②	<スロット位置> 数値	
	情報を取得したいスロットイン電源モジュールのスロット位置を指定します。 スロット位置 P1は1、スロット位置 P2は2、…という順に指定します。 省略すると、全スロットの情報を取得します。		
	備 考	・スロット位置省略時のスロット数は 6です。	
	使用関数	・本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmGetPsmId、CmGetPsmStatus関数を使用しています。	
	使用例	PRINT CMPSMGET("ID", 1) 'P1スロットのスロット電源モジュールのID値を得ます。	

3.20 詳細個別説明（製品情報）

2. CMVERGET

関数								
機 能	ファームウェアバージョンを取得します。							
書 式	<(戻り値)バージョン値>=CMVERGET(<①対象>)							
戻り値	戻り値	<バージョン値> 数値						
	対象で指定した、ファームウェアバージョンを取得します。 値は、0始まりで取得できます。 バージョンアップの度に、1,2,3, … となります。 数値とバージョン表記の対応は、『2.18.1 ファームウェアバージョンの確認方法』を参照してください。							
パラメータ	①	<対象> 数値						
	情報を取得する対象を指定します。							
	<table><tr><th>設定値</th><th>内容</th></tr><tr><td>0</td><td>RASコントローラのファームウェアバージョンを取得します。</td></tr><tr><td>1</td><td>BIOSバージョンを取得します。</td></tr></table>		設定値	内容	0	RASコントローラのファームウェアバージョンを取得します。	1	BIOSバージョンを取得します。
設定値	内容							
0	RASコントローラのファームウェアバージョンを取得します。							
1	BIOSバージョンを取得します。							
使用関数	本コマンドは、システム監視ソフトウェア(DPG-0226)の、CmGetFirmwareVersion関数を使用しています。							
使用例	PRINT CMVERGET(0) 'RASコントローラのバージョンを取得します。							

3.21 エラーコード一覧

エラーコードが&h01000012(=ドライバまたはライブラリの呼び出しに失敗した) の時、ERRSUB関数で ドライバ内部のエラーコード値を取得できます。エラーコードの詳細は、以下の表のようになります。

ERRSUB関数で値を受け取るには、ON ERROR CALL命令で、エラー発生時の割り込みルーチン内で、以下のよう記述可能です。

```
' エラーが発生したら、CB_ERR サブルーチンを呼び出すように定義します
ON ERROR CALL CB_ERR
ERROR ON

SUB CB_ERR(I_ERR, I_ERM$, I_ERL)
' ドライバまたはライブラリの呼び出しに失敗した
IF I_ERR = &H01000012 THEN
    NUM& = ERRSUB      ' ドライバ内部のエラーコード値を取得します
    PRINT "システム監視内部のエラーコード値="; HEX$(NUM&)
END IF
END SUB
```

値	意味	対処方法
&HC0000001 (-1073741823)	・ 内部エラー	・ ハードウェア制御中の内部エラーです。 どのような状況でエラーが発生したかを 弊社サポート窓口にご連絡ください。
&HC0000002 (-1073741822)	・ 入力パラメータが不正	・ 設定可能範囲をご確認の上、パラメータを 設定してください。
&HC0000003 (-1073741821)	・ 既にWDTが動作している ・ 既にイベントが登録されている	・ WDT停止中に実行してください。 ・ イベントが登録されていない状態で実行 してください。
&HC0000004 (-1073741820)	・ まだWDTが動作していない ・ まだイベントが登録されていない ・ まだRAIDボードを検出中	・ WDT動作中に実行してください。 ・ イベントが登録されている状態で実行し てください。 ・ 時間をおいて再実行してください。
&HC0000005 (-1073741819)	・ NULLポインタ検出	・ 引数にNULLが渡されています。 有効な変数を指定してください。
&HC0000006 (-1073741818)	・ メモリ確保失敗	・ メモリの使用を抑えてください。
&HC0000007 (-1073741817)	・ 内部エラー	・ 共有ライブラリ初期化時の内部エラーで す。 ドライバが組み込まれているかご確認く ださい。
&HC0000008 (-1073741816)	・ 未サポート	・ 本機能はサポートしていません。 サポートの有無はマニュアルをご確認く ださい。
&HC0000009 (-1073741815)	・ デバイス接続異常	・ 対象のデバイスが正しく接続されてい るかご確認ください。
&HC000000Ah (-1073741814)	・ 現在はホット・スワップを要求できない	・ RAID状態を確認してください。必要な場 合はコールド・スワップを検討してくださ い

第4章 サンプルプログラム

サンプルプログラムは「/usr/share/interface/AJANPro/samples/SML/」に格納されています。

AJAN統合開発環境を起動すると、左ペインのエクスプローラウィンドウ内の「Samples/SML/」に、ファイルが取り込まれて配置されます。

4.1 サンプルプログラム一覧

ファイル名	内 容
●WDT	
MGRWDT.AJN	WDTの周期を5秒、アクションをCPUリセットに設定し、3秒ごとにWDTクリアを行うサンプルプログラムです。
●温度監視	
MGRTEMPERATURE.AJN	3秒ごとにCPUコア温度とCPU基板温度を取得し、表示するサンプルプログラムです。
●電源電圧監視	
MGRPOWER.AJN	3秒ごとに外部電源電圧を取得し、表示するサンプルプログラムです。
●RAS出力	
MGRRASOUT.AJN	異常時の自動RAS出力を行うサンプルプログラムです。 WDTの周期を3秒に設定し、WDTエラーが発生した時点でRAS出力をリレーONに設定するプログラムです。
●RAS通知	
MGRRASNOTIFY.BAS	WDT異常時にブザーをONするサンプルプログラムです。 WDT異常発生時にブザーが自動でONとなる設定を行い、WDT周期を3秒に設定して、WDTをスタートさせて5秒間待機します。 5秒後、WDTを停止させてブザーの自動通知を無効化し、ブザーを停止させてプログラムを終了します。
●デジタル入出力	
MGRDIO.AJN	汎用デジタル入出力を行うサンプルプログラムです。 デジタル出力後、1秒待機してデジタル入力を行い、入力データを表示します。出力データをシフトさせながら4回繰り返します。
MGRDICONFIG.AJN	デジタル入力接点1を電源スイッチに設定するサンプルプログラムです。 デジタル入力接点1を電源スイッチに設定後10秒間待機し、デジタル入力接点1を汎用入力に設定して終了します。
●割り込みイベント	
MGREVENT.AJN	WDT周期満了イベントをコールバック関数で通知するサンプルプログラムです。 WDT周期を3秒に設定し、WDTをスタートさせます。 3秒おきにWDT周期満了イベントが発生し、10秒後にWDTを停止させてプログラムを終了します。
●LED制御	
MGRERRORLED.AJN	エラーLEDを直接制御するサンプルプログラムです。 エラーLEDの点灯パターンをパターン1(常時点灯)に設定し、1秒ごとに点滅させ、10回点滅すると終了します。
MGRALAMLED.AJN	アラームLEDを直接制御するサンプルプログラムです。 アラームLEDの点灯パターンをパターン1(常時点灯)に設定し、1秒ごとに点滅させ、10回点滅すると終了します。
MGRCOLORLED.AJN	LEDの発光色を直接制御するサンプルプログラムです。 エラーLEDの点灯パターンをパターン1(常時点灯)に設定しつつ、発光色を1秒ごとに切り替えて、全ての発光色を切り替えると終了します。 ※本機能は有償オプションです。ご利用されたい方は、多色設定機能付きの製品をお買い求めください。
●ブザー制御	
MGRBUZZER.AJN	ブザー制御のサンプルプログラムです。
●ひもスイッチ	
MGRHIMOSWITCH.AJN	3秒ごとにひもスイッチの状態を表示するサンプルプログラムです。
●FAN監視	
MGRFANROTATE.AJN	ファンの回転数を表示するサンプルプログラムです。
●衝撃・振動監視	

ファイル名	内 容
MGRGSENSE.AJN	重力加速度値を表示するサンプルプログラムです。 ※本機能は有償オプションです。ご利用されたい方は、加速度センサの機能付きの製品をお買い求めください。
●遷移出力	
MGRTRANSOUTPUT.AJN	遷移出力対応製品に対して、停止時に全接点の出力状態をON、起動時に全接点の出力状態をOFFに設定するサンプルプログラムです。 プログラム実行後、シャットダウンを行うとシステム停止時に出力が全てON、起動時に出力が全てOFFになります。

第5章 付録

5.1 システム監視ソフトウェア

ここでは、システム監視制御の各機能の詳細、およびハードウェア仕様、各機能の使用可否について知る方法を述べます。

まず、お使いのコンピュータで、どのようなシステム監視機能があり、ハードウェア仕様、制限事項を知りたい場合は、弊社Web siteの、製品情報ページのユーザーズマニュアルを参照ください。

下図は、2021/10月時点で、SuperCD(型式：VAC-G019(L8XA)M04)の製品情報ページの一部を抜粋したものです。

ソフトウェア				
Linux	システム監視Linuxソフトウェア	DPG-0226	Ver.2.40-15	ダウンロード
Linux	Interface Backup Tool (VUY-0800)	VUY-0800	Ver.1.00-01	有償ソフト
ドキュメント				
マニュアル				
	ユーザーズマニュアル(日本語)	M14-JG019L8XAM04	Ver.1.1 2020/03/17	PDF
	ユーザーズマニュアル(日本語)	M14-JG019L8XAM04	Ver.1.1 2020/03/17	PDF
	クイックスタートガイド	MNE-0012	Ver.1.1 2018/09/11	PDF
	ハードウェア仕様ドキュメント			PDF

※ ダウンロードのトラブルの解決方法

下の赤い囲みの「ユーザーズマニュアル」には、お使いのコンピュータで使用可能なシステム監視機能と、ハードウェア仕様、コネクタのピンアサインメントなどの情報が記載されています。

上の赤い囲みの「システム監視Linuxソフトウェア」の「DPG-0226」とある箇所をクリックすると、システム監視ソフトウェアの情報ページに飛びます。

(この例では、型式が「DPG-0226」でしたが、製品によっては「DPG-0200」の場合もあります)

型式	DPG-0226
バージョン	2.40-15
価格	無料ダウンロード CD-ROMなどの媒体による提供は¥3,000 (税込み ¥3,300)
オンラインヘルプ	こちらから 英語版はこちら
機能対応表	こちらから 英語版はこちら
ダウンロード	ここ
概要	

上の赤い囲みの「オンラインヘルプ」横の「こちらから」をクリックすると、システム監視ソフトウェアの「オンラインヘルプ」が参照できます。

ここには、システム監視の各機能の詳細な説明が載っています。

説明中に出てくる関数名は、異なるプログラミング言語(C言語)向けですが、機能の動作や仕様は同じです。

5.2 機能対応表

機能対応表(func_table.pdf)は、システム監視ソフトウェア(DPG-0226)に付属する、システム監視ソフトウェアの各関数群や機能などが、どの弊社コンピュータ製品で利用できるか示した表です。

AJANの本システム監視コマンドは、システム監視ソフトウェアを使用しており、この機能一覧表と付き合わせる事で、どのコマンドが、どの弊社コンピュータで利用可能か知る事ができます。

AJAN システム監視コマンドの解説の使用関数項に、どのシステム監視ソフトウェアの関数を使用しているかが記載されています。

ここで記載されている関数名と、機能対応表の関数名、使用する弊社コンピュータ製品を照らし合わせる事で、使用可否を確認できます。

		RESET	周期満了時に CPU リセットを起こします。	
	BIOS 設定でウォッチドッグタイマを有効にしている場合は、その設定となります			
使用関数	本コマンドは、システム監視ライブラリ(DPG-0226/0200)の、CmSetWDTConfig 関数を使用しています。			
使用例	『2.1 WDT』を参照してください。			

機能対応表(func_table.pdf)は、以下の方法で閲覧いただけます。

- システム監視ソフトウェアが製品にプリインストールされている場合
 - DPG-0226 : 「/usr/src/」にある「DPG0226」フォルダ内にfunc_table.pdfが格納されています。
- 弊社Web siteからダウンロードする場合

以下のページからオンラインで閲覧頂けます。また、ダウンロードした製品内にもfunc_table.pdfが含まれています。

 - DPG-0226 : 弊社Web(<http://www.interface.co.jp/>)にて「DPG-0226」で検索を行うと、製品ページが表示されますので、下の赤い囲みの「機能対応表」横の「こちらから」をクリックすると、ドキュメントが参照できます。

Interface® Vision & Freedom 株式会社 インタフェース ユーザ1

新着情報 製品情報 サポート

ホーム > 製品情報 > ソフトウェア > DPG-0226

DPG-0226 システム監視Linuxソフトウェア

型式	DPG-0226
バージョン	1.90-10
価格	無料ダウンロード CD-ROMなどの媒体による提供は¥3,000 (税込み ¥3,300)
オンラインヘルプ	こちらから 英語版はこちら
機能対応表	こちらから 英語版はこちら
ダウンロード	ダウンロード ※ 製品のシリアル番号入力が必要です

概要

システム監視ソフトウェアは、Linux上のアプリケーションから、弊社産

第6章 索引

B		R	
BATPOWEROFFTIME.....	49	RAIDSTATUS.....	62
BATSTATUS.....	47	RASCONTROLSTATUS.....	56
BUZZERCONTROL.....	53	RASERRORSTATUS.....	55
C		RASNOTIFY.....	37
CM OFF.....	44	RAS出力.....	10
CM ON.....	44	RAS通知.....	10
CM STOP.....	44	ROCONFIG.....	36
CMBUZZER.....	53	ROOUTPUT.....	35
CMEVENT.....	45	ROSTATUS.....	36
CMEVENTCONFIG.....	42	T	
CMLLED.....	52	TEMPGET.....	28
CMPSMGET.....	65	TEMPLIMIT.....	29
CMTRANSCONFIG.....	63	TEMPLOWERLIMIT.....	31
CMVERGET.....	66	TEMPUPPERLIMIT.....	30
D		U	
DIGITALFILTER.....	34	UPCONFIG.....	41
E		W	
EXCONTROL.....	61	WDT.....	7
F		WDTCLEAR.....	27
FANAUTOSPEED.....	58	WDTCONFIG.....	26
FANAUTOTHRESHOLD.....	57	WDTSTART.....	27
FANROTATEGET.....	57	WDTSTOP.....	27
FAN監視.....	18	え	
G		エラー/アラームLED.....	14
GSENSEGET.....	59	す	
GSENSELIMIT.....	60	ストレージ監視.....	19
H		スロットイン電源モジュール監視	20
HIMOSWITCHCONFIG.....	54	て	
HIMOSWITCHSTATUS.....	54	デジタル入出力.....	11
I		ひ	
INPUTUP.....	40	ひもスイッチ.....	17
L		ふ	
LEDCONTROL.....	50	ブザー制御.....	16
LEDPATTERN.....	51	漢字	
LED制御.....	14	温度監視.....	8
O		概要.....	4, 70
ON CM CALL.....	44	割り込みイベント.....	12
OUTPUTUP.....	39	機能対応表.....	71
P		自己診断.....	20
POWERGET.....	32	瞬低対策用電源監視.....	13
POWERLIMIT.....	33		

振動・衝撃監視.....	19	汎用LED	15
遷移出力.....	21, 23	用語説明	5
電源電圧監視	9		

第7章 重要な情報

保証の内容と制限

弊社は本ドキュメントに含まれるソースプログラムの実行が中断しないこと、またはその実行に誤りが無いことを保証していません。

本製品の品質や使用に起因する、性能に起因するいかなるリスクも使用者が負うものとします。

弊社はドキュメント内の情報の正確さに万全を期しています。万一、誤記または誤植などがあつた場合、弊社は予告無く改訂する場合があります。ドキュメントまたはドキュメント内の情報に起因するいかなる損害に対しても弊社は責任を負いません。

ドキュメント内の図や表は説明のためであり、ユーザ個別の応用事例により変化する場合があります。

著作権、知的所有権

弊社は本製品に含まれるおよび本製品に対する権利や知的所有権を保持しています。

本製品はコンピュータ ソフトウェア、映像/音声(例えば図、文章、写真など)を含んでいます。

医療機器/器具への適用における注意

弊社の製品は人命に関わるような状況下で使用される機器に用いられる事を目的として設計、製造された物では有りません。

弊社の製品は人体の検査などに使用するに適する信頼性を確保する事を意図された部品や検査機器と共に設計された物では有りません。

医療機器、治療器具などの本製品の適用により、製品の故障、ユーザ、設計者の過失などにより、損傷/損害を引き起こす場合があります。

複製の禁止

弊社の許可なく、本ドキュメントの全て、または一部に関わらず、複製、改変などを行うことはできません。

責任の制限

弊社は、弊社または再販売者の予見の有無にかかわらず発生したいかなる特別損害、偶発的損害、間接的な損害、重大な損害について、責任を負いません。

本製品(ハードウェア、ソフトウェア)のシステム組み込み、使用、ならびに本製品から得られる結果に関する一切のリスクについては、本製品の使用者に帰属するものとします。

本製品に含まれる不都合、あるいは本製品の供給(納期遅延)、性能もしくは使用に起因する付帯的損害もしくは間接的損害に対して、弊社に全面的に責がある場合でも、弊社はその製品に対する改良(有償サービスの利用)、代品交換までとし、製品の予防交換並びに、代金減額等、金銭面での賠償の責任は負わないものとします。

本製品は、日本国内仕様です。

商標/登録商標

本書に掲載されている会社名、製品名は、それぞれ各社の商標または登録商標です。

改訂履歴

Ver.	年 月	改 訂 内 容
0.90	2019年10月	新規作成
0.91	2021年10月	最新情報に更新
0.92	2022年1月	誤記修正

このマニュアルは、製品の改良その他により将来予告なく改訂しますので、予めご了承ください。