

LW SERIES

Windows API・USB デバイスドライバ

Ver. 3.10



＜ソフトウェア使用許諾契約＞

1. 権利の許諾

当社はお客様に対して、本使用許諾契約に同意いただいてダウンロード可能となるソフトウェア及びその関連資料(以下「本ソフトウェア」といいます)に関し、以下の権利を許諾します。

(a) お客様は、本ソフトウェアに対応する当社製品を利用する目的で本ソフトウェアを使用することができます。

(b) お客様は、本ソフトウェアを複製し、1台以上のコンピュータ上で使用することができます。

2. 追加許諾条項

本ソフトウェアを定められた目的に従って使用した結果、作成された各種のファイルは、お客様の著作物となります。

3. 著作権

本ソフトウェア及びその複製物の著作権は当社又は当社が認めた者が有するものであり、日本国著作権法及び国際条約によって保護されています。本使用許諾契約に基づき、お客様が本ソフトウェアを複製する場合は、ダウンロードされた本ソフトウェアに付されていたものと同一の著作権表示がなされることを要します。

4. 禁止事項

本ソフトウェアがソースコードで提供される場合、お客様は、ソースコードを改変したものを当社製として第三者に配布することはできません。

5. 無保証

当社は、本ソフトウェアがお客様特定の目的のために適切であること、もしくは有用であること、又は本ソフトウェアに瑕疵がないこと、その他本ソフトウェアに関していかなる保証もいたしません。

6. 免責

当社は、いかなる場合においても、本ソフトウェアの使用又は使用不能から生ずるいかなる損害(事業利益の損害、事業の中断、事業情報の損失、又はその他金銭的損害)に関して、一切責任を負いません。

7. 契約の解除

お客様が本使用許諾契約に違反した場合、当社は本使用許諾契約を解除することができます。その場合、お客様は本ソフトウェアを一切使用しないものとします。

＜サポート＞

本ソフトウェアの不具合等のお問い合わせは、当社サービスまでお願いいたします。

＜保証範囲＞

本ソフトウェアのダウンロード・インストールはお客様の責任においておこなっていただきます。また本ソフトウェアは、予告せず改良、変更することがあります。

＜著作権者＞

各ソフトウェアの著作権は、当社に帰属します。

＜各社商標＞

TEXIO は当社の産業用電子機器における製品ブランドです。また、本説明書に記載されている会社名および商品名は、それぞれの国と地域における各社および各団体の商標または登録商標です。

目 次

ソフトウェア使用許諾契約

第 1 章 概 要	1
第 2 章 動作環境	1
第 3 章 API のインストール方法	1
第 4 章 USB デバイスドライバのインストール	2
4-1. インストールについて	2
4-2. 再インストールについて	2
第 5 章 関数一覧表	3
第 6 章 デバイス制御関数	4
6-1. LW_DeviceOpen	4
6-2. LW_DeviceClose	4
6-3. LW_SetTimeOut	5
第 7 章 INPUT 制御関数	5
7-1. LW_SetMainInput	5
7-2. LW_GetMainInput	6
7-3. LW_SetInputSelect	6
7-4. LW_GetInputSelect	7
第 8 章 プリセット制御関数	7
8-1. LW_SetPresetNumber	7
8-2. LW_GetPresetNumber	8
8-3. LW_SavePreset	8
第 9 章 モニター制御関数	9
9-1. LW_GetMonitor	9
9-2. LW_SetMonitorSelect	9
9-3. LW_GetMonitorSelect	10
9-4. LW_SetDisplaySelect	10
9-5. LW_GetDisplaySelect	11
第 10 章 放電負荷制御関数	12
10-1. LW_SetLoadMode	12
10-2. LW_GetLoadMode	13
10-3. LW_SetLoadValue	14
10-4. LW_GetLoadValue	15
10-5. LW_SetSiemensValue	16
10-6. LW_GetSiemensValue	17
10-7. LW_SetCurrentLimit	18
10-8. LW_GetCurrentLimit	19
第 11 章 メモリー制御関数	20
11-1. LW_SaveLoadValue	20
11-2. LW_SaveCurrentLimit	21

11-3. LW_SaveMemoryData.....	21
第 12 章 スイッチング制御関数	22
12-1. LW_SetSwSelect.....	22
12-2. LW_GetSwSelect	22
12-3. LW_SetSwFreq.....	23
12-4. LW_GetSwFreq	23
12-5. LW_SetSwDuty	24
12-6. LW_GetSwDuty.....	24
12-7. LW_SetSwTime	25
12-8. LW_GetSwTime.....	25
12-9. LW_SetSwMethod.....	26
12-10. LW_GetSwMethod	26
第 13 章 デレイ制御関数	27
13-1. LW_SetDelayFunction	27
13-2. LW_GetDelayFunction.....	27
13-3. LW_SetDelayTime.....	28
13-4. LW_GetDelayTime	28
第 14 章 トラッキング制御関数	29
14-1. LW_SetTrackFunction	29
14-2. LW_GetTrackFunction.....	29
14-3. LW_SetTrackDirection.....	30
14-4. LW_GetTrackDirection	30
14-5. LW_SetTrackMode	31
14-6. LW_GetTrackMode.....	31
14-7. LW_SetTrackValue.....	32
14-8. LW_GetTrackValue	32
第 15 章 アラーム制御関数	33
15-1. LW_ResetAlarm	33
15-2. LW_GetAlarm	34
15-3. LW_GetLimit.....	35
第 16 章 その他の関数	35
16-1. LW_SetOperationTimeOut	35
16-2. LW_GetOperationTimeOut.....	36
16-3. LW_SetLocal.....	36
16-4. LW_SetLocalLockOut.....	37
16-5. LW_GetLocalLockOut.....	37
16-6. LW_GetIdNumber.....	37
16-7. LW_GetChExist.....	38
第 17 章 利用上の注意について	39
第 18 章 制御用通信コマンド	40

第 1 章 概要

本ソフトウェアは LW シリーズの WindowsAPI を提供します。パーソナルコンピュータ(以下、PC)により、GP-IB、USB で制御するためのアプリケーション作成に利用できます。

PC を使用して電源をコントロールするためには、Windows が動作するための PC と専用のインタフェイスユニット(IF-50 シリーズ)が必要です。

第 2 章 動作環境

マイクロソフト社の Windows XP SP3(32bit)、VISTA(32bit) 、Windows7 以降(32bit / 64bit)が動作する PC。

対応言語として、Visual Basic6.0、Visual Basic for Applications、Visual C++6.0(C,C++)、VisualBasic.2005、VisualC++.2005、VisualC#.2005 以後を想定しています。また、Windows7 以降(64bit)で使用できる言語は VisualBasic2008、Visual C++2008、Visual C#2008 以後を想定しています。

GP-IB については、National Instruments 社製の GP-IB ドライバ、GP-IB カード、NI488.2M Software Ver2.8.1 以降についてのみ対応しております。他社の GP-IB ドライバでは API は動作しません。

USB については、当社の USB デバイスドライバに対応しております。本ドライバは USB のプラグアンドプレイに完全に適合していません。電源本体の電源を途中で ON・OFF したり、追加・削除した場合の動作は保証できませんのでご注意ください。また、USB のインタフェイスは、GP-IB と比較して手軽なだけに電氣的に弱く、ノイズや振動のある環境に適していません、十分なノイズ対策と検証を行った上でご使用ください

電源を複数のプログラム、プロセス、スレッドから同時にコントロールする場合は通信が重ならないように十分検証をおこなった上でご使用ください。

第 3 章 API のインストール方法

以下の手順によって、API をインストールしてください。

- PC へのログインは必ず管理者(administrator)としてください。
- 適当なフォルダにダウンロードしたファイルを展開し、API フォルダにある Setup32.exe(32bit 用)／Setup64.exe(64bit 用)を右クリックで管理者として実行し、インストールしてください。
(注意) Window7(64bit)で VisualBasicC6.0,VBA(32bit),VC6 を使用する場合は、32bit の API を使用します。
- USB を使用する場合は続けて Driver フォルダにある Setup**.exe を実行し、デバイスドライバをインストールしてください。API の利用方法はサンプルプログラムが用意しておりますのでこちらを参照してください。
- OS によってはインストール時に設定または続行の許可のメッセージが出ますが、そのまま続行してください。
- インストール中にフォルダの指定が出る場合がありますが、インストール用のフォルダは作成されません、ファイルは全てシステムフォルダにコピーされます。
- .netFramework4 ライブラリと VC++ランタイムライブラリのインストールが自動で行われます。システムと競合してエラーが発生しインストールできない場合はライブラリを一旦削除してからインストールするか TEXIO_API**.msi を実行してください。
- 再インストールは環境に応じて TEXIO_API32.msi または、TEXIO_API64.msi を実行して下さい。

第4章 USB デバイスドライバのインストール

4-1. インストールについて

- ・ 本体の電源スイッチを OFF にしたままで USB ケーブルを動作している PC に接続してください。
- ・ PC へのログインは必ず管理者(administrator)としてください。
- ・ 32ビット・64ビットの環境に合わせた Driver フォルダにある Setup32.exe(32bit 用)／Setup64.exe(64bit 用)を右クリックで管理者として実行しインストールしてください。途中で マイクロソフトのロゴテストに合格していない とメッセージが出るがありますが、そのまま続行してください。
- ・ インストールが完了してから、本体の電源スイッチを ON して PC が認識するのを待ちます。接続している機器の台数や状況により時間がかかる場合がありますので注意してください。
- ・ 認識された場合、新しいハードウェアの検出ウィザードが始まります。自動でインストールを選択し、メッセージにしたがってデバイスドライバをインストールします。途中で マイクロソフトのロゴテストに合格していない とのメッセージが出るがありますが、そのまま続行してください。
- ・ OS によってはインストール時に設定または続行の許可のメッセージが出ますが、そのまま続行してください。
- ・ PC によってはセキュリティの関係で本器が自動認識されないことがあります。この場合はコントロールパネルのデバイスマネージャーで"ほかのデバイス"にある本器を右クリックで選択し、ドライバの更新を行ないます。ドライバファイルは手動で同梱の inf ファイルを指定してください。

4-2. 再インストールについて

- ・ 本ドライバを使用している全てのデバイスとの通信が停止している事を確認してください。
- ・ 「コントロールパネル>プログラムの追加と削除」からドライバを削除してください。
(Windows ドライバパッケージ (WinUSB) USB Driver の当社発行のもの)
- ・ ドライバを Setup**.exe で再度インストールしてください。
- ・ デバイスマネージャーで、接続している本体が正常に認識・動作していることを確認してください。

第 5 章 関数一覧表

関 数	内 容
int LW_DeviceOpen(int Class, int BoardAdr, int PcAdr, int SysAdr)	デバイスハンドルのオープン
int LW_DeviceClose(int DeviceID)	デバイスハンドルのクローズ
int LW_SetTimeOut(int DeviceID, int Time)	タイムアウト時間の設定
int LW_SetMainInput(int DeviceID, int Flag)	メインインプットの設定
int LW_GetMainInput(int DeviceID)	メインインプットの状態読出し
int LW_SetInputSelect(int DeviceID, int Ch, int Flag)	インプットセレクトの設定
int LW_GetInputSelect(int DeviceID, int Ch)	インプットセレクトの状態読出し
int LW_SetPresetNumber(int DeviceID, int Preset)	プリセット指定の設定
int LW_GetPresetNumber(int DeviceID)	プリセット指定の状態読出し
int LW_SavePreset(int DeviceID, int Src, int Dest)	プリセットデータのコピー指示
int LW_GetMonitor(int DeviceID, int Ch, struct MONITOR_DATA *Monitor)	モニタデータの読出し
int LW_SetMonitorSelect(int DeviceID, int Flag)	モニタ表示の設定
int LW_GetMonitorSelect(int DeviceID)	モニタ表示の状態読出し
int LW_SetDisplaySelect(int DeviceID, int Ch)	表示 CH の設定
int LW_GetDisplaySelect(int DeviceID)	表示 CH の状態読出し
int LW_SetLoadMode(int DeviceID, int Preset, int Ch, int Mode, int Ref)	負荷モードの設定
int LW_GetLoadMode(int DeviceID, int Preset, int Ch, struct MODE_DATA *ModeData)	負荷モードの状態読出し
int LW_SetLoadValue(int DeviceID, int Preset, int Ch, double Data)	負荷設定値の設定
int LW_GetLoadValue(int DeviceID, int Preset, int Ch, double *Data)	負荷設定値の読出し
int LW_SetSiemensValue(int DeviceID, int Preset, int Ch, int Data)	負荷設定値の設定 (シーメンス値)
int LW_GetSiemensValue(int DeviceID, int Preset, int Ch, int *Data)	負荷設定値の読出し (シーメンス値)
int LW_SetCurrentLimit(int DeviceID, int Preset, int Ch, double Data)	電流制限値の設定
int LW_GetCurrentLimit(int DeviceID, int Preset, int Ch, double *Data)	電流制限値の読出し
int LW_SaveLoadValue(int DeviceID, int Preset, int Ch)	負荷設定値の記憶指示
int LW_SaveCurrentLimit(int DeviceID, int Preset, int Ch)	電流制限値の記憶指示
int LW_SaveMemoryData(int DeviceID, int object)	設定情報の記憶指示
int LW_SetSwSelect(int DeviceID, int Ch, int Flag)	スイッチング動作指定
int LW_GetSwSelect(int DeviceID, int Ch)	スイッチング動作状態読出し
int LW_SetSwFreq(int DeviceID, int Freq)	スイッチング周波数設定
int LW_GetSwFreq(int DeviceID, int *Freq)	スイッチング周波数読出し
int LW_SetSwDuty(int DeviceID, int Duty)	デューティ値設定
int LW_GetSwDuty(int DeviceID, int *Duty)	デューティ値読出し
int LW_SetSwTime(int DeviceID, double Ta, double Tb)	スイッチング時間設定
int LW_GetSwTime(int DeviceID, struct TIME_DATA *Time)	スイッチング時間読出し
int LW_SetSwMethod(int DeviceID, int Flag)	スイッチング状態設定
int LW_GetSwMethod(int DeviceID)	スイッチング状態読出し
int LW_SetDelayFunction(int DeviceID, int Flag)	ディレイ動作設定
int LW_GetDelayFunction(int DeviceID)	ディレイ動作状態読出し
int LW_SetDelayTime(int DeviceID, int Ch, double Time)	ディレイ時間設定
int LW_GetDelayTime(int DeviceID, int Ch, double *Time)	ディレイ時間読出し
int LW_SetTrackFunction(int DeviceID, int Flag)	トラッキング動作設定
int LW_GetTrackFunction(int DeviceID)	トラッキング動作情報読出し
int LW_SetTrackDirection(int DeviceID, int ChA, int ChB, int ChC, int ChD)	トラッキング方向設定
int LW_GetTrackDirection(int DeviceID, struct TRACK_DATA *Track)	トラッキング方向状態読出し
int LW_SetTrackMode(int DeviceID, int Mode)	トラッキングモード設定
int LW_GetTrackMode(int DeviceID)	トラッキングモード読出し
int LW_SetTrackValue(int DeviceID, double Data)	トラッキング変位置量設定
int LW_GetTrackValue(int DeviceID, double *Data)	トラッキング変位置量読出し
int LW_ResetAlarm(int DeviceID)	アラームクリア
int LW_GetAlarm(int DeviceID, struct ALARM_DATA *Alarm)	アラーム状態読出し
int LW_GetLimit(int DeviceID, struct LIMIT_DATA *Limit)	リミット情報読出し
int LW_SetOperationTimeOut(int DeviceID, int Time)	操作タイムアウト時間の設定
int LW_GetOperationTimeOut(int DeviceID)	操作タイムアウト時間の読出し
int LW_SetLocal(int DeviceID)	ローカル状態への移行
int LW_SetLocalLockOut(int DeviceID, int Flag)	ローカルロックアウトの設定
int LW_GetLocalLockOut(int DeviceID)	ローカルロックアウトの状態読出し
int LW_GetIdNumber(int DeviceID)	本体 ID の読出し
int LW_GetChExist(int DeviceID, struct CH_DATA *ChData)	チャンネル装備状態の読出し

本定義では int:32bit で記述されています。

VC++より API を利用する場合は DLL インポート用のサンプルを利用します。サンプルでは関数名が“LW_”をとった名前で登録してあります。

VisualBasic6.0 より API を利用する場合は関数定義が IF_50.bas にありますので参照してください。また VisualBasic2005 以後の場合は IF_50.vb に関数定義があります。

第 6 章 デバイス制御関数

6-1. LW_DeviceOpen

機 能: 指定されたデバイスについて、オープン処理を行う。

書 式: int LW_DeviceOpen(int Class, int BoardAdr, int PcAdr, int SysAdr)

パラメータ:

- ・ Class
デバイス種別の指定。
GPIB、または USB を指定します。
 - 1 GPIB 指定
 - 2 - USB 指定
- ・ BoardAdrs
ボードアドレスの指定。
GPIB のボードアドレスを指定します。
ISB の場合は、無視されます。
 - 0~3 - GPIB ボードアドレス
- ・ PcAdrs
PC アドレスの指定。
PC アドレスを設定します。
 - 0~31 - PC アドレス
- ・ SysAdrs
SYS アドレスの指定。
SYS アドレスを設定します。
 - 1~32 - SYS アドレス

戻り値:

指定されたデバイスのデバイス ID が戻ります。

エラーが発生した場合は、負数が戻ります。

- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

なし

6-2. LW_DeviceClose

機 能: 指定されたデバイスについて、クローズ処理を行う。

書 式: int LW_DeviceClose(int DeviceID)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
クローズ対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー

注意事項:

なし

6-3. LW_SetTimeOut

機 能: 指定されたデバイスについて、通信タイムアウト設定を行う。

書 式: int LW_SetTimeOut(int DeviceID, int Time)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
設定対象となるデバイスのデバイス ID を指定します。
- ・ Time
タイムアウト時間指定(単位一秒)。0~65535。
0 秒指定で、タイムアウト検出を禁止します。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 対象デバイスが GP-IB の場合、設定値は以下の値となります。
1 秒、3 秒、10 秒、30 秒、100 秒、300 秒、1000 秒
設定値と時間については GP-IB のマニュアルを参照してください。
- ・ ローカルバスの機器を指定した場合はマスターの LW のタイムアウトが変更されます。
またローカルバスを利用する場合はタイムアウトを2秒以上に設定する必要があります。

第 7 章 INPUT 制御関数

7-1. LW_SetMainInput

機 能: 指定されたデバイスについて、MAIN_INPUT ON/OFF 切替え処理を行う。

書 式: int LW_SetMainInput(int DeviceID, int Flag)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Flag
ON/OFF 切替指定。
0 - OFF 指定
1 - ON 指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

なし

7-2. LW_GetMainInput

機能: 指定されたデバイスについて、MAIN_INPUT 状態の取得を行う。

書式: int LW_GetMainInput(int DeviceID)

パラメータ:

- DeviceId
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、MAIN_INPUT の状態が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - OFF
- 1 - ON
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

7-3. LW_SetInputSelect

機能: 指定されたデバイスについて、INPUT_SELECT ON/OFF 切替え処理を行う。

書式: int LW_SetInputSelect(int DeviceID, int Ch, int Flag)

パラメータ:

- DeviceId
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- Ch
切替え対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- Flag
ON/OFF 切替指定。
 - 0 - OFF 指定
 - 1 - ON 指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- 実装されていない CH を指定した場合、関数の実行は正常終了しますが、切替動作は無視されます。
また 1 チャンネル入力モデルでは切替えは無視されます。

7-4. LW_GetInputSelect

機 能: 指定されたデバイスについて INPUT_SELECT 状態の取得を行う。

書 式: int LW_GetInputSelect(int DeviceID, int Ch)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。
- ・ Ch
状態取得対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定

戻り値:

関数が成功した場合、対象 CH の INPUT_SELECT 状態が戻ります。エラーが発生した場合は、負数が戻ります。

- 0 - OFF
- 1 - ON
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー
- 4 - 受信エラー

注意事項:

- ・ 実装されていない CH を指定した場合はエラーとなります。

第 8 章 プリセット制御関数

8-1. LW_SetPresetNumber

機 能: 指定されたデバイスについて、プリセット番号切替え処理を行う。

書 式: int LW_SetPresetNumber(int DeviceID, int Preset)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Preset
切替え後のプリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

なし

8-2. LW_GetPresetNumber

機能: 指定されたデバイスについて、プリセット番号の取得を行う。

書式: int LW_GetPresetNumber(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、プリセット番号が戻ります。

エラーが発生した場合は、負数が戻ります。

- 1 - プリセット番号 1
- 2 - プリセット番号 2
- 3 - プリセット番号 3
- 4 - プリセット番号 4
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

8-3. LW_SavePreset

機能: 指定されたデバイスについて、プリセットデータの転送と保存を行う。

書式: int LW_SavePreset(int DeviceID, int Src, int Dest)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
取得対象となるデバイスのデバイス ID を指定します。
- ・ Src
転送元のプリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Dest
転送先のプリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 転送元、転送先に同一プリセットを指定した場合、プリセットデータの保存のみが行われます。
- ・ 転送された設定は直ちに、不揮発性メモリに保存されます。

第 9 章 モニター制御関数

9-1. LW_GetMonitor

機 能: 指定されたデバイスについて、モニターデータの取得を行う。

書 式: int LW_GetMonitor(int DeviceID, int Ch, struct MONITOR_DATA *Monitor)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
取得対象となるデバイスのデバイス ID を指定します。
- ・ Ch
モニターデータ取得対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Monitor
取得結果格納用モニターデータ構造体のポインタ。
struct MONITOR_DATA{
 double Current; モニター電流値(単位-A)
 double Voltage; モニター電圧値(単位-V)
 double Wattage; モニター電力値(単位-W)
};

戻り値:

関数が成功した場合、1 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー
- 4 - 受信エラー

注意事項:

- ・ 実装されていない CH を指定した場合はエラーとなります。

9-2. LW_SetMonitorSelect

機 能: 指定されたデバイスについて、モニター表示指定切替え処理を行う。

書 式: int LW_SetMonitorSelect(int DeviceID, int Flag)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Flag
切替え後のモニター表示指定。
 - 0 - 電流/電圧表示
 - 1 - 電流/電力表示

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

なし

9-3. LW_GetMonitorSelect

機 能: 指定されたデバイスについて、モニター表示指定の取得を行う。

書 式: int LW_GetMonitorSelect(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、モニター表示指定が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 電流／電圧表示
- 1 - 電流／電力表示
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

9-4. LW_SetDisplaySelect

機 能: 指定されたデバイスについて、表示CH切替え処理を行う。

書 式: int LW_SetDisplaySelect(int DeviceID, int Ch)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Ch
表示 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 実装されていない CH を指定した場合、関数の実行は正常終了しますが、切替え動作は無視されます。

9-5. LW_GetDisplaySelect

機 能: 指定されたデバイスについて、表示CHの状態取得を行う。

書 式: int LW_GetDisplaySelect(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、表示CHの状態が戻ります。エラーが発生した場合は、負数が戻ります。

- 1 - CH-A 指定
- 2 - CH-B 指定
- 3 - CH-C 指定
- 4 - CH-D 指定
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

第 10 章 放電負荷制御関数

10-1. LW_SetLoadMode

機能: 指定されたデバイスについて、放電負荷モード設定処理を行う。

書式: int LW_SetLoadMode(int DeviceID, int Preset, int Ch, int Mode, int Ref)

パラメータ:

- DeviceID
デバイス ID の指定。
設定対象となるデバイスのデバイス ID を指定します。
- Preset
設定対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- Ch
設定対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- Mode
設定放電負荷モード指定。
 - 1 - CC モード、電流レンジ H を指定
 - 2 - CC モード、電流レンジ L を指定
 - 3 - CR モード、電流レンジ H を指定
 - 4 - CR モード、電流レンジ L を指定
 - 5 - CV モード、電流レンジ H を指定
 - 6 - CV モード、電流レンジ L を指定
 - 7 - CP モード、電流レンジ H、電圧レンジ L を指定
 - 8 - CP モード、電流レンジ H、電圧レンジ H を指定
 - 9 - CP モード、電流レンジ L、電圧レンジ L を指定
 - 10 - CP モード、電流レンジ L、電圧レンジ H を指定
 - 11 - ショートモードを指定
- Ref
コントロールリファレンス指定。
 - 0 - 内部電圧コントロールを指定
 - 1 - 外部電圧コントロールを指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- 実装されていない CH、放電負荷モード、コントロールリファレンスを指定した場合、関数は成功しますが、設定は無視されます。
- 「MAIN_INPUT」が ON 状態で、指定されたプリセットが現在運用中のプリセットと同一の場合、関数は成功しますが、設定は無視されます。
- 変更された設定は直ちに、不揮発性メモリに保存されます。

10-2. LW_GetLoadMode

機能: 指定されたデバイスについて、放電負荷モード設定情報の取得を行う。

書式: int LW_GetLoadMode(int DeviceID, int Preset, int Ch, struct MODE_DATA *ModeData)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。
- ・ Preset
対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Ch
対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ ModeData
取得結果格納用、放電負荷モード設定データ構造体のポインタ。

```
struct MODE_DATA{  
    int      Mode;  
    int      Reference;  
};
```

 - ◇ Mode 放電負荷モード設定。
 - 1 - CC モード、電流レンジ H
 - 2 - CC モード、電流レンジ L
 - 3 - CR モード、電流レンジ H
 - 4 - CR モード、電流レンジ L
 - 5 - CV モード、電流レンジ H
 - 6 - CV モード、電流レンジ L
 - 7 - CP モード、電流レンジ H、電圧レンジ L
 - 8 - CP モード、電流レンジ H、電圧レンジ H
 - 9 - CP モード、電流レンジ L、電圧レンジ L
 - 10 - CP モード、電流レンジ L、電圧レンジ H
 - 11 - ショートモード
 - ◇ Ref コントロールリファレンス設定。
 - 0 - 内部電圧コントロール
 - 1 - 外部電圧コントロール

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー
- 4 - 受信エラー

注意事項:

- ・ 実装されていない CH を指定した場合はエラーとなります。

10-3. LW_SetLoadValue

機能: 指定されたデバイスについて、放電負荷設定値の設定処理を行う。

書式: int LW_SetLoadValue(int DeviceID, int Preset, int Ch, double Data)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
設定対象となるデバイスのデバイス ID を指定します。
- ・ Preset
設定対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Ch
設定対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Data
放電負荷設定値。
0.0~9999.9999
電流は A 単位。
電圧は V 単位。
抵抗は Ω 単位。
電力は W 単位。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 実装されていない CH を指定した場合、関数は成功しますが、設定は無視されます。
- ・ 設定値が対象となる CH の放電モードで規定された範囲を超える場合は無視されます。
- ・ 対象 CH がショートモードの場合、無視されます。
- ・ 対象 CH が CR モードの場合、抵抗値 (Ω) での設定となります。その場合、実際に設定される値はコンダクタンス (S) 換算された値となりますので、本コマンドで指定した設定値と異なる場合があります。
- ・ 変更された設定は不揮発性メモリには保存されません。必要に応じて、保存操作を実施してください。

10-4. LW_GetLoadValue

機能: 指定されたデバイスについて、放電負荷設定値の取得を行う。

書式: int LW_GetLoadValue(int DeviceID, int Preset, int Ch, double *Data)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。
- ・ Preset
対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Ch
対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Data
放電負荷設定値格納用ポインタ。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー
- 4 - 受信エラー

注意事項:

- ・ 実装されていない CH を指定した場合はエラーとなります。

10-5. LW_SetSiemensValue

機能: 指定されたデバイスについて、放電負荷設定値(シーメンス値)の設定処理を行う。

書式: int LW_SetSiemensValue(int DeviceID, int Preset, int Ch, int Data)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
設定対象となるデバイスのデバイス ID を指定します。
- ・ Preset
設定対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Ch
設定対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Data
放電負荷設定値(シーメンス値)。
3~30000

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 実装されていない CH を指定した場合、関数は成功しますが、設定は無視されます。
- ・ CR モードでない CH を指定した場合、関数は成功しますが、設定は無視されます。
- ・ 設定値が規定された範囲を超える場合は無視されます。
- ・ 変更された設定は不揮発性メモリには保存されません。必要に応じて、保存操作を実施してください。

10-6. LW_GetSiemensValue

機 能: 指定されたデバイスについて、放電負荷設定値(シーメンス値)の取得を行う。

書 式: int LW_GetSiemensValue(int DeviceID, int Preset, int Ch, int *Data)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。
- ・ Preset
対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Ch
対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Data
放電負荷設定値格納用ポインタ。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー
- 4 - 受信エラー

注意事項:

- ・ 実装されていない CH を指定した場合はエラーとなります。
- ・ CR モードでない CH を指定した場合はエラーとなります。

10-7. LW_SetCurrentLimit

機 能: 指定されたデバイスについて、電流リミット設定値の設定処理を行う。

書 式: int LW_SetCurrentLimit(int DeviceID, int Preset, int Ch, double Data)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
設定対象となるデバイスのデバイス ID を指定します。
- ・ Preset
設定対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Ch
設定対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Data
電流リミット設定値。
0.0~99.999

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 実装されていない CH を指定した場合、関数は成功しますが、設定は無視されます。
- ・ 設定値が規定された範囲を超える場合は無視されます。
- ・ 変更された設定は不揮発性メモリには保存されません。必要に応じて、保存操作を実施してください。

10-8. LW_GetCurrentLimit

機 能: 指定されたデバイスについて、電流リミット設定値の取得を行う。

書 式: int LW_GetCurrentLimit(int DeviceID, int Preset, int Ch, double *Data)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。
- ・ Preset
対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Ch
対象 CH 指定
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Data
電流リミット設定値格納用ポインタ。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー
- 4 - 受信エラー

注意事項:

- ・ 実装されていない CH を指定した場合はエラーとなります。
- ・ 指定された CH がショートモードまたは CC モードの場合はエラーとなります。

第 11 章 メモリ制御関数

11-1. LW_SaveLoadValue

機 能: 指定されたデバイスについて、放電負荷設定値の保存処理を行う。

書 式: int LW_SaveLoadValue(int DeviceID, int Preset, int Ch)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
保存対象となるデバイスのデバイス ID を指定します。
- ・ Preset
保存対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Ch
保存対象 CH。
 - 1 - CH-A
 - 2 - CH-B
 - 3 - CH-C
 - 4 - CH-D

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 実装されていない CH 定した場合、関数は成功しますが、設定は無視されます。

11-2. LW_SaveCurrentLimit

機 能: 指定されたデバイスについて、電流リミット設定値の保存処理を行う。

書 式: int LW_SaveCurrentLimit(int DeviceID, int Preset, int Ch)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
保存対象となるデバイスのデバイス ID を指定します。
- ・ Preset
保存対象プリセット番号指定。
 - 1 - プリセット番号 1 指定
 - 2 - プリセット番号 2 指定
 - 3 - プリセット番号 3 指定
 - 4 - プリセット番号 4 指定
- ・ Ch
保存対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 実装されていない CH を指定した場合、関数は成功しますが、設定は無視されます。

11-3. LW_SaveMemoryData

機 能: 指定されたデバイスについて、各種設定データの保存処理を行う。

書 式: int LW_SaveMemoryData(int DeviceID, int object)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
保存対象となるデバイスのデバイス ID を指定します。
- ・ object
保存対象設定データの指定
 - 1 - SW 周波数
 - 2 - SW デューティ
 - 3 - SW 時間 ta,tb
 - 4 - デイレイ時間
 - 5 - トラッキング設定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

なし

第 12 章 スイッチング制御関数

12-1. LW_SetSwSelect

機 能: 指定されたデバイスについて、SW_SELECT ON/OFF 切替処理を行う。

書 式: int LW_SetSwSelect(int DeviceID, int Ch, int Flag)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Ch
切り替え対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Flag
ON/OFF 切替え指定。
 - 0 - OFF 指定
 - 1 - ON 指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 実装されていない CH を指定した場合、関数の実行は正常終了しますが、切替え動作は無視されます。
- ・ 指定された CH がスイッチング動作不可の場合、関数の実行は正常終了しますが、切替え動作は無視されます。

12-2. LW_GetSwSelect

機 能: 指定されたデバイスについて、SW_SELECT 状態の取得を行う。

書 式: int LW_GetSwSelect(int DeviceID, int Ch)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。
- ・ Ch
状態取得対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定

戻り値:

関数が成功した場合、対象 CH の SW_SELECT 状態が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - OFF
- 1 - ON
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー
- 4 - 受信エラー

注意事項:

- ・ 実装されていない CH を指定した場合はエラーとなります。

12-3. LW_SetSwFreq

機能: 指定されたデバイスについて、スイッチング周波数設定処理を行う。

書式: int LW_SetSwFreq(int DeviceID, int Freq)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Freq
スイッチング周波数設定値(単位-Hz)
1~500

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ スwitching設定モードが「周波数」の場合、設定値は直ちに実際の SW 波形に反映されますが、「時間」の場合、実際の波形には反映されません。
- ・ 当コマンドでは設定データの不揮発性メモリ書き込みは致しません。

12-4. LW_GetSwFreq

機能: 指定されたデバイスについて、スイッチング周波数設定値の取得を行う。

書式: int LW_GetSwFreq(int DeviceID, int *Freq)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。
- ・ Freq
スイッチング周波数設定値格納用ポインタ。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

12-5. LW_SetSwDuty

機 能: 指定されたデバイスについて、スイッチングデューティ設定処理を行う。

書 式: int LW_SetSwDuty(int DeviceID, int Duty)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Duty
スイッチングデューティ設定値(単位 -%)。
5~95

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ スwitching設定モードが「周波数」の場合、設定値は直ちに実際の SW 波形に反映されますが、「時間」の場合、実際の波形には反映されません。
- ・ 当コマンドでは設定データの不揮発性メモリ書き込みは致しません。

12-6. LW_GetSwDuty

機 能: 指定されたデバイスについて、スイッチングデューティ設定値の取得を行う。

書 式: int LW_GetSwDuty(int DeviceID, int *Duty)

パラメータ:

- ・ DeviceID
デバイ ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。
- ・ Duty
スイッチングデューティ設定値格納用ポインタ。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

12-7. LW_SetSwTime

機能: 指定されたデバイスについて、スイッチング時間設定処理を行う。

書式: int LW_SetSwTime(int DeviceID, double Ta, double Tb)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Ta
スイッチング時間設定値 ta(単位-ms)。
0.1~950.0
- ・ Tb
スイッチング時間設定値 tb(単位-ms)。
0.1~950.0

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 設定値 ta, tb から計算される周波数、デューティが規定範囲を超える設定は無視されます。
- ・ スwitching設定モードが「時間」の場合、設定値は直ちに実際の SW 波形に反映されますが、「周波数」の場合、実際の波形には反映されません。
- ・ 当コマンドでは設定データの不揮発性メモリ書き込みは致しません。

12-8. LW_GetSwTime

機能: 指定されたデバイスについて、スイッチング時間設定値の取得を行う。

書式: int LW_GetSwTime(int DeviceID, struct TIME_DATA *Time)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。
- ・ Time
取得結果格納用時間データ構造体のポインタ。
struct TIME_DATA{
double Ta; イッチング時間 ta(単位-ms)
double Tb; イッチング時間 tb(単位-ms)
};

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

12-9. LW_SetSwMethod

機能: 指定されたデバイスについて、スイッチング設定方法の切替え処理を行う。

書式: int LW_SetSwMethod(int DeviceID, int Flag)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Flag
スイッチング設定方法切替え指定。
 - 0 - 周波数&デューティ指定
 - 1 - 時間 ta&tb 指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 変更された設定は不揮発性メモリに保存されます。

12-10. LW_GetSwMethod

機能: 指定されたデバイスについて、スイッチング設定方法の状態の取得を行う。

書式: int LW_GetSwMethod(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、現在のスイッチング設定方法が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 周波数&デューティ
- 1 - 時間 ta&tb
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

第 13 章 デイレイ制御関数

13-1. LW_SetDelayFunction

機 能: 指定されたデバイスについて、デイレイ機能の ON/OFF 切替え処理を行う。

書 式: int LW_SetDelayFunction(int DeviceID, int Flag)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Flag
ON/OFF 切替え指定。
0 - OFF 指定
1 - ON 指定

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

なし

13-2. LW_GetDelayFunction

機 能: 指定されたデバイスについて、デイレイ機能の状態取得を行う。

書 式: int LW_GetDelayFunction(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、デイレイ機能の状態が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - OFF
- 1 - ON
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

13-3. LW_SetDelayTime

機 能: 指定されたデバイスについて、ディレイ時間設定処理を行う。

書 式: int LW_SetDelayTime(int DeviceID, int Ch, double Time)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
設定対象となるデバイスのデバイス ID を指定します。
- ・ Ch
設定対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Time
ディレイ時間設定値(単位一秒)。
0.0~10.00。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 実装されていない CH を指定した場合、関数は成功しますが、設定は無視されます。
- ・ 当コマンドでは設定データの不揮発性メモリ書き込みは致しません。

13-4. LW_GetDelayTime

機 能: 指定されたデバイスについて、ディレイ時間設定値の取得を行う。

書 式: int LW_GetDelayTime(int DeviceID, int Ch, double *Time)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。
- ・ Ch
取得対象 CH 指定。
 - 1 - CH-A 指定
 - 2 - CH-B 指定
 - 3 - CH-C 指定
 - 4 - CH-D 指定
- ・ Time
取得結果格納用のポインタ(単位一秒)。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー
- 4 - 受信エラー

注意事項:

- ・ 実装されていない CH を指定した場合はエラーとなります。

第 14 章 トラッキング制御関数

14-1. LW_SetTrackFunction

機 能: 指定されたデバイスについて、オンライントラッキング機能 ON/OFF の切替え処理を行う。

書 式: int LW_SetTrackfunction(int DeviceID, int Flag)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Flag
オンライントラッキング機能 ON/OFF 切替え指定。
 - 0 - OFF
 - 1 - ON

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 1CH 実装モデルでは無視されます。
- ・ 手動で動作中のトラッキング動作はキャンセルされます。
- ・ %トラッキングの場合、このコマンド受信時の設定が 100%となります。
- ・ トラッキング設定されている全ての CH がトラッキング条件を満たしていない場合は無視されます。

14-2. LW_GetTrackFunction

機 能: 指定されたデバイスについて、オンライントラッキング機能 ON/OFF 状態の取得を行う。

書 式: int LW_GetTrackFunction(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、現在のオンライントラッキング状態が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - OFF
- 1 - ON
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

14-3. LW_SetTrackDirection

機 能: 指定されたデバイスについて、トラッキング方向の設定処理を行う。

書 式: int LW_SetTrackDirection(int DeviceID, int ChA, int ChB, int ChC, int ChD)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
設定対象となるデバイスのデバイス ID を指定します。
- ・ ChA~ChD
各 CH のトラッキング方向設定値。
 - 0 - トラッキング無し
 - 1 - 「+」トラッキング
 - 2 - 「-」トラッキング

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 1CH 実装モデルでは無視されます。
- ・ 手動でのトラッキング中は無視されます。
- ・ 未実装の CH 分については無視されます。
- ・ 当コマンドでは設定データの不揮発性メモリ書き込みは致しません。

14-4. LW_GetTrackDirection

機 能: 指定されたデバイスについて、トラッキング方向設定値の取得を行う。

書 式: int LW_GetTrackDirection(int DeviceID, struct TRACK_DATA *Track)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。
- ・ Track
取得結果格納用データ構造体のポインタ。

```
struct TRACK_DATA{  
    int    ChA;  
    int    ChB;  
    int    ChC;  
    int    ChD;  
};
```

◇ ChA~ChD

- 各 CH のトラッキング方向設定値。
 - 0 - トラッキング無し
 - 1 - 「+」トラッキング
 - 2 - 「-」トラッキング

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

- ・ 実装されていない CH の設定は「0」(トラッキング無し)となります。

14-5. LW_SetTrackMode

機 能: 指定されたデバイスについて、オンライントラッキングモードの切替え処理を行う。

書 式: int LW_SetTrackMode(int DeviceID, int Mode)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Mode
オンライントラッキングモード切替え指定。
 - 0 - %トラッキングモード
 - 1 - 絶対値トラッキングモード

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 1CH 実装モデルでは無視されます。
- ・ 当コマンドでは設定データの不揮発性メモリ書き込みは致しません。

14-6. LW_GetTrackMode

機 能: 指定されたデバイスについて、オンライントラッキングモードの取得を行う。

書 式: int LW_GetTrackMode(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、現在のオンライントラッキングモードが戻ります。エラーが発生した場合は、負数が戻ります。

- 0 - %トラッキングモード
- 1 - 絶対値トラッキングモード
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

14-7. LW_SetTrackValue

機 能: 指定されたデバイスについて、オンライントラッキングモード中の設定値ロード処理を行う。

書 式: int LW_SetTrackValue(int DeviceID, double Data)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
設定対象となるデバイスのデバイス ID を指定します。
- ・ Data
設定値。
0.0~9999.9999
設定可能範囲は、CH の設定による。
%トラッキングモードの場合は相対値。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 1CH 実装モデルでは無視されます。
- ・ オンライントラッキング中以外は無視されます。
- ・ 設定値が対象となる CH で規定された範囲を超える場合はそれぞれの最大値、最小値に設定されます。
- ・ 当コマンドでは設定データの不揮発性メモリ書き込みは致しません。

14-8. LW_GetTrackValue

機 能: 指定されたデバイスについて、オンライントラッキング設定値(%トラッキング)の取得を行う。

書 式: int LW_GetTrackValue (int DeviceID, double *Data)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。
- ・ Data
取得結果格納用のポインタ。
%トラッキングの現在値。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

- ・ 絶対値トラッキングの場合、「999.9」が戻ります。
- ・ オンライントラッキング中でない場合、「999.9」が戻ります。

第 15 章 アラーム制御関数

15-1. LW_ResetAlarm

機 能: 指定されたデバイスについて、アラームのリセットを行う。

書 式: int LW_ResetAlarm(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
制御対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ アラーム要因が解除されていない場合は、リセット出来ません。
- ・ 複数のアラームが同時に発生している場合、優先順位の高いアラームのみをリセットします。

15-2. LW_GetAlarm

機能: 指定されたデバイスについて、アラームデータの取得を行う。

書式: int LW_GetAlarm(int DeviceID, struct ALARM_DATA *Alarm)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。

- ・ Alarm
取得結果格納用データ構造体のポインタ。

```
struct ALARM_DATA{  
    int      Eta;  
    int      Oha;  
    int      Ova[4];  
    int      Oca[4];  
    int      Ora[4];  
};
```

※ データの内容は全て共通

- 0 - 正常状態
- 1 - アラーム状態

- ◇ Eta
外部アラーム。
- ◇ Oha
オーバーヒートアラーム。
- ◇ Ova[0]~[3]
オーバーボルテージアラーム。
CH-A~CH-D
- ◇ Oca[0]~[3]
オーバーカレントアラーム。
CH-A~CH-D
- ◇ Ora[0]~[3]
オーバーレンジアラーム。
CH-A~CH-D

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

- ・ 未実装の CH については、常に正常となります。

15-3. LW_GetLimit

機能: 指定されたデバイスについて、リミットデータの取得を行う。

書式: int LW_GetLimit(int DeviceID, struct LIMIT_DATA *Limit)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
状態取得対象となるデバイスのデバイス ID を指定します。
- ・ Limit
取得結果格納用データ構造体のポインタ。
struct LIMIT_DATA{
 int Current[4];
 int Powe[4];
};
※ データの内容は全て共通
0 - 正常状態
1 - リミット状態
- ◇ Oca[0]~[3]
電流リミット。
CH-A~CH-D
- ◇ Ora[0]~[3]
電力リミット。
CH-A~CH-D

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

- ・ 未実装の CH については、常に正常となります。

第 16 章 その他の関数

16-1. LW_SetOperationTimeOut

機能: 指定されたデバイスについて、操作タイムアウト時間の設定値ロードを行う。

書式: int LW_SetOperationTimeOut(int DeviceID, int Time)

パラメータ:

- ・ DeviceId
デバイス ID の指定。
設定対象となるデバイスのデバイス ID を指定します。
- ・ Time
操作タイムアウト時間設定値(単位一分)。
0~10
※ 「0」設定でタイムアウト無し

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

- ・ 変更された設定は不揮発性メモリに保存されます。

16-2. LW_GetOperationTimeOut

機 能: 指定されたデバイスについて、操作タイムアウト設定値の取得を行う。

書 式: int LW_GetOperationTimeOut(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
設定値取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、現在の操作タイムアウト設定値が戻ります。

0~10 — 操作タイムアウト設定値(単位一分)

※「0」で操作タイムアウト無し

エラーが発生した場合は、負数が戻ります。

- 1 — デバイスエラー
- 2 — タイムアウトエラー
- 4 — 受信エラー

注意事項:

なし

16-3. LW_SetLocal

機 能: 指定されたデバイスについて、ローカル状態への切替を行う。

書 式: int LW_SetLocal(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 — 正常終了
- 1 — デバイスエラー
- 2 — タイムアウトエラー
- 3 — パラメータエラー

注意事項:

なし

16-4. LW_SetLocalLockOut

機 能: 指定されたデバイスについて、ローカルロックアウト機能の切替え処理を行う。

書 式: int LW_SetLocalLockOut(int DeviceID, int Flag)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
切替え対象となるデバイスのデバイス ID を指定します。
- ・ Flag
ローカルロックアウト切替え指定。
 - 0 - ロック解除可能
 - 1 - ロック解除不可

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 3 - パラメータエラー

注意事項:

なし

16-5. LW_GetLocalLockOut

機 能: 指定されたデバイスについて、ローカルロックアウト設定の取得を行う。

書 式: int LW_GetLocalLockOut(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
設定値取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、現在のローカルロックアウト設定が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 解除可能
- 1 - 解除不可
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

16-6. LW_GetIdNumber

機 能: 指定されたデバイスについて、ID 番号の取得を行う。

書 式: int LW_GetIdNumber(int DeviceID)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
設定値取得対象となるデバイスのデバイス ID を指定します。

戻り値:

関数が成功した場合、LW の ID 番号が戻ります。

エラーが発生した場合は、負数が戻ります。

- 1~ - ID 番号
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

- ・ 「ID 番号」は LW 本体に設定されているハードウェア構成を識別する番号であり、「デバイス ID」とは異なります。番号については LW 本体取扱説明書を参照してください。

16-7. LW_GetChExist

機 能: 指定されたデバイスについて、CH 実装情報の取得を行う。

書 式: int LW_GetChExist(int DeviceID, struct CH_DATA *ChData)

パラメータ:

- ・ DeviceID
デバイス ID の指定。
情報取得対象となるデバイスのデバイス ID を指定します。

- ・ ChData
取得結果格納用データ構造体のポインタ。

```
struct CH_DATA{  
    int      Ch_A;  
    int      Ch_B;  
    int      Ch_C;  
    int      Ch_D;  
};
```

- ◇ Ch_A~Ch_D
CH の実装情報。
0 - 未実装
1 - 実装

戻り値:

関数が成功した場合、0 が戻ります。

エラーが発生した場合は、負数が戻ります。

- 0 - 正常終了
- 1 - デバイスエラー
- 2 - タイムアウトエラー
- 4 - 受信エラー

注意事項:

なし

第 17 章 利用上の注意について

- ・複数のローカルバスマスターに重複した PC アドレスを設定しないでください。また、同一ローカルバスに接続されているスレーブ機に、重複したシステムアドレスを設定しないでください。正常に動作しません。
- ・ローカルバスは、バスマスター機がコマンドを解釈してからスレーブ機にコマンドを転送します、このため、実際のコマンド実行まで遅延があるため注意が必要です。また、短い時間で設定変更を繰り返す行くと、本体の動作速度が低下したり、コマンドを受付けない場合がありますので、コマンド送信間隔の設定には十分注意し、タイムアウトの設定を長めにおこなってください。
- ・ローカルバスマスターは、電源 ON 時に接続されているスレーブを確認しますので、スレーブと同時に電源が立ち上がるようにしてください。また、スレーブの構成やアドレス設定が変わる場合は、必ずローカルバスマスターの電源を OFF にしてください。
- ・ローカルバス接続を行なう場合は、PC 及び OS の高速化・最適化やローカル機器の台数により、連続してコマンドを送信すると正常にコマンドを受付けない場合があります。コマンド間にウェイトを 100ms 程度入れると安定することがあります。PC の CPU、チップセット、クロックや OS のドライバ類が変更になった場合は、十分ご注意ください。
- ・PC の CPU クロックが高速な場合に連続して設定や読み取りを行うと、本体の動作速度が遅くなる場合があります。また、同じ設定のオン・オフを繰り返して行くと、処理が終わる前に次の設定が行われるため、設定が省略されて見えることがあります。特に内部のステータスやモニタデータの読み出しについては、数百 ms の読取周期がありますので注意が必要です。
- ・本 API は、サスペンドなどの PC の省電力モードに対応しておりません、省電力モード以後の本機の動作は不定となりますので設定には十分注意してください。
- ・本 API 内部で double 型の変数を利用している値については、2 進数演算による演算誤差を含む場合があります。この場合は、最小分解能の半分の値を加算して有効桁数以後を切り捨てると正しい値を得ることができます。詳しくは利用する開発環境での数値の取り扱いを確認してください。
- ・電源を複数のプログラム、プロセス、スレッドから同時にコントロールする場合は、通信が重ならないように十分検証をおこなった上でご使用ください。
- ・アプリケーションのフォルダなどに、API の "IF_50.dll" がある場合は、動作がおかしくなる場合があります。フォルダから DLL を削除して API を正しくインストールしてください。
- ・WindowsUpdate によるインストールが実行中の場合は API または USB デバイスドライバのインストールができないことがあります、WindowsUpdate の処理がすべて終了してから再度インストールをおこなってください。
- ・通信を行う場合に、PC および機器のグランド電位が異なったり、接地されていない場合は、動作が不安定になります。必ず 3 芯の電源ケーブルを利用し、筐体が接地した状態でご利用ください。USB ハブを利用する場合も、できるだけ筐体グランドが同じ電位になるように設置してください。
- ・モーターやインバーター、溶接機などノイズが発生している環境では、通信が不安定になります。PC、被制御機器、通信経路、商用電源がノイズの影響を受けないように設置し、十分に動作確認をした上でご利用ください。
- ・OS や USB の仕様変更に伴い接続断やノイズなどの状態異常時の各関数の戻り値や動作が正しくない場合があります。

第 18 章 制御用通信コマンド

API の関数を利用せずに GP-IB から直接制御を行う場合のコマンド一覧は以下のとおりです。
 またナショナルインスツルメンツ社製以外の GP-IB ボードを利用する場合はこちらを利用します。
 なおパラメータなどの詳細は LW 本体取扱説明書を参照してください。

項目	コマンド	パラメータ
被制御機の設定	SV	システムアドレス
プリセット	PRESET	1/2/3/4
放電モード設定	LMODE	プリセット、CH、モード、外部制御
メインインプット設定	MINPUT	0/1
インプットセレクト設定	INPSEL	CH、0/1
放電量設定	VALUE	プリセット、CH、放電モード設定値
CR モードの STEP 値設定	SVALUE	プリセット、CH、CR モード STEP 値
電流制限値の設定	CLIM	プリセット、CH、電流制限値
スイッチングセレクト指定	SWSEL	CH、0/1
スイッチング周波数設定	SWFREQ	周波数
スイッチング DUTY 設定	SWDUTY	デューティ比
スイッチング時間設定	SWTIME	Ta 時間、Tb 時間
スイッチング方法の切替	SWSET	0/1
ディレイ動作指定	DELAY	0/1
ディレイ時間設定	DLYTIM	CH、ディレイ時間
タイムアウト設定	TOUT	タイムアウト時間
表示チャンネル選択	DISPSEL	CH
表示項目選択	MONSEL	0/1
ローカルロックアウト設定	LOCKOUT	0/1
トラッキング設定	TRSET	0/1,0/1,0/1,0/1
トラッキングモードの設定	TRMODE	0/1
トラッキング動作指定	TRACK	0/1
トラッキング変位量指定	TRVAL	増減量
プリセット記憶	SETPRE	コピー元、コピー先 (クエリ無)
設定値記憶	MEMWRT	保存項目 (クエリ無)
放電量記憶	VALWRT	プリセット、CH (クエリ無)
電流制限記憶	CLMWRT	プリセット、CH (クエリ無)
アラームクリア	ALMRST	なし (クエリ無)
ローカル指定	LOCAL	なし (クエリ無)
スレーブ検出	SLV?	なし (設定無)
機器 ID	ID?	なし (設定無)
製品情報読出し	* IDN?	なし (設定無)
動作状態読出し	MONDATA?	CH (設定無)
アラーム読出し	ALARM?	なし (設定無)
リミット状態読出し	LIMIT?	なし (設定無)

クエリ要求時はパラメータが“?”となります



株式会社 テクシオ・テクノロジー

〒222-0033 横浜市港北区新横浜 2-18-13 藤和不動産新横浜ビル 7F
<http://www.texio.co.jp/>

アフターサービスに関しては下記サービスセンターへ

サービスセンター 〒222-0033 横浜市港北区新横浜 2-18-13 藤和不動産新横浜ビル 7F
TEL.045-620-2786