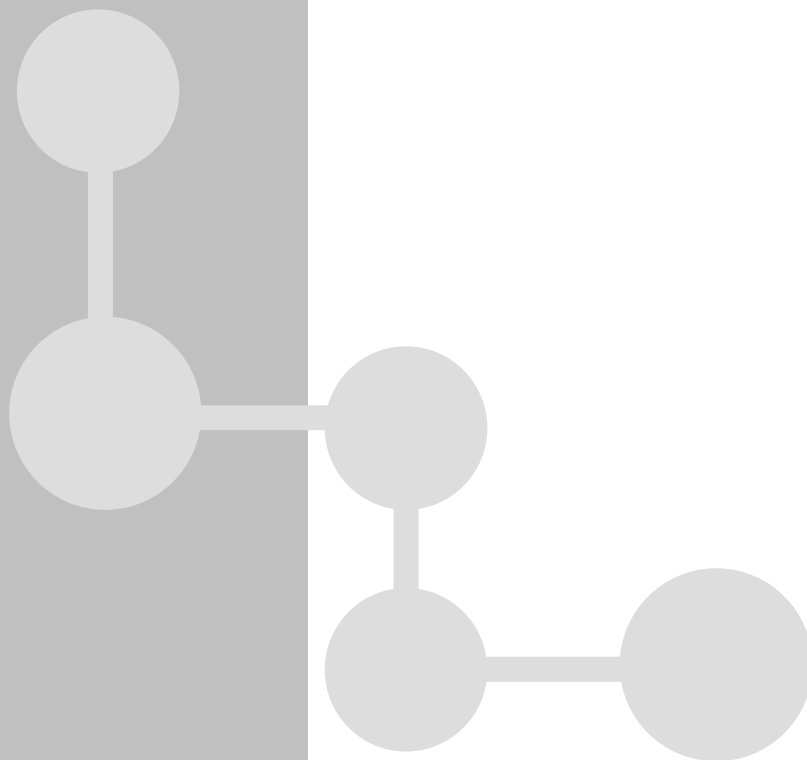


CASSIOPEIA

DT-5100 シリーズ

ライブラリマニュアル for C++
【システムライブラリ編】



変更履歴

No	Revision	更新日	項	改訂内容
1	1.00	03/1/20	初版	初版発行
2	2.00	03/07/16	2	5項目関数追加
3	〃	〃	5	パラメータ説明修正 dwkey *500(msec) → 5sec
4	〃	〃	6	パラメータ説明修正 デフォルト時間 5000msc → 5sec
5	〃	〃	8	説明文修正 PBT_APMRESUMESUSPEND → PBT_APMSUSPEND
6	〃	〃	15	関数書式変更 パラメータにDWORD dwTimeを追加
7	〃	〃	33	CLBSetCPUMode関数のパラメータ修正 → DWORD dwMode
8	〃	〃	34	CLBGetCPUMode関数のパラメータ修正 → DWORD dwMode
9	〃	〃	35	関数名変更 CLB180Rotate → CLBSet180Rotate
10	〃	〃	36	CLBGet180Rotate関数を追加
11	〃	〃	37	CLBGetDeviceIDCode関数を追加
12	〃	〃	38	CLBGetUserIDCode関数を追加
13	〃	〃	39	CLBPrepareLED関数を追加
14	〃	〃	40	CLBUpdateLED関数を追加
15	〃	〃	41	CLBGetBackupLife、CLBSetBackupLife関数を追加
16	3.00	05/03/15		表紙、目次を追加しました。
17				
18				
19				
20				
21				
22				
23				
24				
25				

目次

1	システムライブラリ	1
1.1	機能.....	1
1.2	動作環境.....	1
1.3	開発環境.....	1
1.4	提供ファイル.....	1
1.5	3-5.使用方法.....	1
1.6	ライブラリー一覧.....	2
1.7	関数リファレンス	2
	CLBSetBootableButtons	3
	CLBGetBootableButtons	4
	CLBSetOffMaskTime	5
	CLBGetOffMaskTime	6
	CLBPowerOff	7
	CLBDisablePowerOff	8
	CLBEnablePowerOff	9
	CLBGetPowerOff	10
	CLBAPODisable	11
	CLBAPOEnable	12
	CLBAPOGet	13
	CLBSoftReset	14
	CLBSetIndicator	15
	CLBGetIndicator	16
	CLBCheckCharger	17
	CLBCardDetectDisable	18
	CLBCardDetectEnable	19
	CLBCardDetectGet	20
	CLBSetFnButtons	21
	CLBGetFnButtons.....	22
	CLBSetInputChangeButtons	23
	CLBGetInputChangeButtons	24
	CLBSetKeyLock.....	25
	CLBGetKeyLock	26
	CLBPlayBuzzer.....	27
	CLBStopBuzzer.....	28
	CLBSetBuzzerVolume	29
	CLBGetBuzzerVolume.....	30
	CLBSetBuzzerMute	31
	CLBGetBuzzerMute	32
	CLBSetCPUMode.....	33
	CLBGetCPUMode	34
	CLBSet180Rotate	35
	CLBGet180Rotate.....	36
	CLBGetDeviceIDCode.....	37
	CLBGetUserIDCode.....	38
	CLBPrepareLED	39
	CLBUpdateLED	40
	CLBGetBackupLife	41
	CLBSetBackupLife.....	42

1 システムライブラリ

1.1 機能

システムライブラリは、キー制御／電源制御等、本機のシステム動作に対する機能補助／拡張を実現するものです。

1.2 動作環境

- 機種 DT-5100
- OS Microsoft WindowsCE .NET

1.3 開発環境

- Microsoft eMbedded C++ Version4.0 + ServicePack 1 + DT-5100ExportSDK

1.4 提供ファイル

- Clbsys.lib インポートライブラリ
- Clbsys.h ヘッダファイル

1.5 使用方法

- ・開発環境において、プログラムソース内にヘッダファイル「Clbsys.h」をインクルードし、インポートライブラリ「Clbsys.lib」を使用するライブラリとして指定してください。

1.6 ライブラリー一覧

NO.	関数名	機能
01	CLBSetBootableButtons	Wakeup 要因による電源 ON 許可/禁止
02	CLBGetBootableButtons	Wakeup 要因による電源許可/禁止の取得
03	CLBSetOffMaskTime	電源 ON 後指定時間 OFF を禁止する
04	CLBGetOffMaskTime	電源 ON 後 OFF を禁止する時間の取得
05	CLBPowerOff	カードのアクセス終了を待って電源 OFF
06	CLBDisablePowerOff	電源ボタンでの OFF を禁止する
07	CLBEnablePowerOff	電源ボタンでの OFF を許可する
08	CLBGetPowerOff	電源ボタンの Enable/Disable を取得する
09	CLBAPODisable	APO を禁止する
10	CLBAPOEnable	APO を許可する
11	CLBAPOGet	APO の禁止/許可状態を取得する
12	CLBSoftReset	ソフト RESET
13	CLBSetIndicator	LED の点灯/消灯
14	CLBGetIndicator	LED の点灯状態取得
15	CLBCheckCharger	IO ボックスとの接続状態を取得する
16	CLBCardDetectDisable	カード検出端子をディセーブルしてカード電源切状態にする
17	CLBCardDetectEnable	カード検出端子をイネーブルにしてカード電源入状態にする
18	CLBCardDetectGet	カード検出端子の状態を取得する
19	CLBSetFnButtons	Fn キー動作の許可/禁止
20	CLBGetFnButtons	Fn キー動作の許可/禁止の取得
21	CLBSetInputChangeButtons	入力切替キー動作の入力設定
22	CLBGetInputChangeButtons	入力切替キー動作の入力の取得
23	CLBSetKeyLock	キー Lock の許可/禁止
24	CLBGetKeyLock	キー Lock の許可/禁止の取得
25	CLBPlayBuzzer	ブザーを鳴らす
26	CLBStopBuzzer	ブザーを止める
27	CLBSetBuzzerVolume	ブザー音量設定
28	CLBGetBuzzerVolume	ブザー音量取得
29	CLBSetBuzzerMute	ブザー全音量、個別ミュート設定
30	CLBGetBuzzerMute	ブザー全音量、個別ミュート取得
31	CLBSetCPUMode	CPU 周波数制御の設定
32	CLBGetCPUMode	CPU 周波数制御の取得
33	CLBSet180Rotate	表示画像を 180 度回転させる
34	CLBGet180Rotate	表示画像の回転状態の取得
35	CLBGetDeviceIDCode	EEPROM に書き込まれている Device ID の取得
36	CLBGetUserIDCode	EEPROM に書き込まれている User ID の取得
37	CLBPrepareLED	LED 点灯準備関数
38	CLBUpdateLED	LED 点灯関数
39	CLBGetBackupLife	VDET1 からの RAM バックアップ時間を取得する
40	CLBSetBackupLife	VDET1 からの RAM バックアップ時間を設定する

1.7 関数リファレンス

次頁より、関数リファレンスを説明します。

なお、エミュレータ機能に関する追記は、「※エミュレータ」部分に記述されています。

CLBSetBootableButtons

機能 Wakeup 要因による電源 ON の有効／無効設定を行います。

本体 OFF 状態で Wakeup 要因が発生することにより、本体電源を ON するようになっています。
本ファンクションにより、この電源 ON 動作を制御することができます。

書式 `BOOL CLBSetBootableButtons(DWORD dwBootableButtons)`

パラメータ `DWORD dwBootableButtons`

<code>CLB_BUTTON_LEFTTRIGGER</code>	左トリガーキーによる電源 ON 有効
<code>CLB_BUTTON_RIGHTTRIGGER</code>	右トリガーキーによる電源 ON 有効
<code>CLB_BUTTON_INBOX</code>	IOBOX 載せによる電源 ON 有効 (デフォルト)
<code>CLB_BUTTON_ARRIVAL</code>	着信信号による電源 ON 有効
<code>CLB_BUTTON_NONE</code>	Wakeup 要因による電源 ON 無効

戻り値 `TRUE` 正常終了
`FALSE` 内部エラー

インクルード `#include <Clbsys.h>`

備考: `CLB_BUTTON_NONE` 以外は、各値の論理和を設定できます。

CLBGetBootableButtons

機能 Wakeup 要因による電源 ON の有効／無効状態を読み出します。

書式 **BOOL CLBGetBootableButtons(DWORD *pdwBootableButtons)**

パラメータ **DWORD *pdwBootableButtons**

指定された領域に、設定状態が返ります。

CLB_BUTTON_LEFTTRIGGER	左トリガーキーによる電源 ON は有効
CLB_BUTTON_RIGHTTRIGGER	右トリガーキーによる電源 ON は有効
CLB_BUTTON_INBOX	IOBOX 載せによる電源 ON は有効
CLB_BUTTON_ARRIVAL	着信信号による電源 ON は有効
CLB_BUTTON_NONE	Wakeup 要因による電源 ON は無効

戻り値 **TRUE** 正常終了

インクルード **#include <Clbsys.h>**

備考: **CLB_BUTTON_NONE** 以外の設定状態では、各値の論理和を返す場合があります。

CLBSetOffMaskTime

機能 電源 ON 後、指定された時間電源 OFF を禁止するための設定を行います。

書式 `BOOL CLBSetOffMaskTime(DWORD dwKey, DWORD dwCard)`

パラメータ `DWORD dwKey` 電源 ON 後の電源キーによる OFF 禁止時間 (sec)
デフォルト時間は、5 sec です
`DWORD dwCard` 常に "0" を指定して下さい

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

CLBGetOffMaskTime

機能 電源 ON 後、電源 OFF を禁止する時間を取得します。

書式 `BOOL CLBGetOffMaskTime(DWORD *pdwKey, DWORD *pdwCard)`

パラメータ `DWORD *pdwKey` 電源 ON 後の電源キーによる OFF 禁止時間 (sec)
デフォルト時間は、5 sec です
`DWORD *pdwCard` "0" を返します

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

CLBPowerOff

機能 本体電源を OFF します。

書式 VOID CLBPowerOff()

パラメータ 無し

戻り値 無し

インクルード #include <Clbsys.h>

CLBDisablePowerOff

機能 電源スイッチによる本体電源 OFF を禁止します。

この設定が有効の場合、電源スイッチを押された時に、WM_POWERBROADCAST の PBT_APMSUSPEND メッセージを発行します。

ユーザーアプリでこれを取得してオフ処理後ソフト OFF して下さい。

本設定は、電源 ON 時に解除されますので、必要に応じて電源 ON 後設定してください。
電池蓋開けや緊急電源 OFF は、本設定をした後でもすぐに OFF します。

書式 BOOL CLBDisablePowerOff()

パラメータ 無し

戻り値 TRUE 正常終了

インクルード #include <Clbsys.h>

CLBEnablePowerOff

機能 電源スイッチによる本体電源 OFF を許可します。

書式 `BOOL CLBEnablePowerOff()`

パラメータ 無し

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

CLBGetPowerOff

機能 電源スイッチによる本体電源 OFF の許可／禁止状態を取得します。

書式 `BOOL CLBGetPowerOff()`

パラメータ 無し

戻り値 **TRUE** 電源 OFF 禁止
FALSE 電源 OFF 許可 デフォルト値

インクルード `#include <Clbsys.h>`

CLBAPODisable

機能 オートパワーオフ (APO) を無効にします。

このファンクションを実行すると、コントロールパネルの設定に関わらず APO しなくなります。

書式 `BOOL CLBAPODisable()`

パラメータ 無し

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

CLBAPOEnable

機能 オートパワーオフ(APO)を有効にします。

このファンクションを実行すると、コントロールパネルの設定通りに APO するようになります。

書式 `BOOL CLBAPOEnable()`

パラメータ 無し

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

備考 本関数を実行しても、他のプロセスやスレッドが“CLBAPODisable”関数を実行している時は、APO は有効になりません。APO を有効にする為には、“CLBAPODisable”関数を実行した回数分、本関数を実行する必要があります。

CLBAPOGet

機能 オートパワーオフ(APO)の有効/無効状態を取得します。

書式 `BOOL CLBAPOGet()`

パラメタ 無し

戻り値 **TRUE** APO 無効
FALSE APO 有効 デフォルト値

インクルード `#include <Clbsys.h>`

CLBSoftReset

機能 システムをリセットします。

書式 void CLBSoftReset()

パラメータ 無し

戻り値 無し

インクルード #include <Clbsys.h>

注意 本ファンクションを実行すると、ただちにシステムがリセットされるため、書き込み中のファイルやデータが失われる可能性があります。オープン中のファイルやデバイスは全てクローズしてから実行してください。

CLBSetIndicator

機能 LED の点灯／消灯を行います。

書式 **BOOL CLBSetIndicator(DWORD dwLedMode, DWORD dwTime,
DWORD dwOnTime, DWORD dwOffTime)**

パラメータ **DWORD dwLedMode**

CLB_LED_OFF	LED を消灯します。(デフォルト値)
CLB_LED_GREEN	LED の緑を点灯します。
CLB_LED_RED	LED の赤を点灯します。
CLB_LED_ORANGE	LED のオレンジを点灯します。

DWORD dwTime LED 連続点灯時間。(1/16 秒単位)
値は(dwOnTime+dwOffTime)×点滅回数を設定

DWORD dwOnTime LED 点灯時間。(1/16 秒単位) 0~255

DWORD dwOffTime LED 消灯時間。(1/16 秒単位) 0~255

戻り値 **TRUE** 正常終了

FALSE 内部エラーによる異常終了

インクルード **#include <Clbsys.h>**

使用例 LED 橙を素早く 1 回点滅する場合の設定

(ハードの仕様により以下の引数 (dwTime,dwOnTime,dwOffTime) が最低の値となります。)

```
CLBSetIndicator (CLB_LED_GREEN | CLB_LED_RED, 4, 1, 3);
```

LED 緑を 1 秒 ON、1 秒 OFF を 3 回点滅する場合の設定

```
CLBSetIndicator (CLB_LED_GREEN, 96, 16, 16);
```

LED 赤を 2 秒点灯する場合の設定

```
CLBSetIndicator (CLB_LED_RED, 35, 32, 3);
```

LED 赤を連続点灯する場合の設定

(連続点灯の場合は dwOnTime,dwOffTime に 0 使用可)

```
CLBSetIndicator (CLB_LED_RED, 255, 255, 0);
```

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBGetIndicator

機能 LED の点灯／消灯状態を取得します。

書式 `DWORD CLBGetIndicator()`

パラメータ 無し

戻り値	CLB_LED_OFF	LED は消灯状態です。(デフォルト値)
	CLB_LED_GREEN	LED は緑を点灯しています。
	CLB_LED_RED	LED は赤を点灯しています。
	CLB_LED_ORANGE	LED はオレンジを点灯しています。

インクルード `#include <Clbsys.h>`

CLBCheckCharger

機能 本体と IO ボックスとの接続状態を取得します。また、接続状態を監視する時間の設定も同時に行います。

書式 `int CLBCheckCharger(DWORD time_out)`

パラメータ **DWORD time_out** 接続状態監視時間
0 : 接続状態を取得するのみ。
1~3600000 : 接続状態監視時間(msec)
INFINITE : 監視時間無限大 (タイムアウト無し)

戻り値 **0** 接続検出
1 タイムアウト発生
-1 内部エラー

インクルード `#include <Clbsys.h>`

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBCardDetectDisable

機能 擬似カード挿抜機能として、カードデテクト端子をディセーブルしてカード電源を切状態にします。

書式 **BOOL** CLBCardDetectDisable(**DWORD socket**)

パラメータ **DWORD socket** ソケット名を指定する

0 : PCMCIA (※DT-870 のみ)

1 : CF (※DT-5100 は無線 LAN モジュールに電源供給を OFF)

戻り値 **TRUE** 正常終了

インクルード **#include <Clbsys.h>**

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBCardDetectEnable

機能 擬似カード挿抜機能として、カードデテクト端子をイネーブルしてカード電源を入状態にします。

書式 **BOOL** CLBCardDetectEnable(**DWORD socket**)

パラメータ **DWORD socket** ソケット名を指定する

0 : PCMCIA (※DT-870 のみ)

1 : CF (※DT-5100 は無線 LAN モジュールに電源供給を ON)

戻り値 **TRUE** 正常終了

インクルード **#include <Clbsys.h>**

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBCardDetectGet

機能 擬似カード挿抜機能として、カードデテクト端子の状態を取得します。

書式 **BOOL** CLBCardDetectGet(**DWORD socket**)

パラメータ **DWORD socket** ソケット名を指定する

0 : PCMCIA (※DT-870 のみ)

1 : CF (※DT-5100 は無線 LAN の電源供給状態)

戻り値 **TRUE** カードデテクト端子イェ-ブル状態

FALSE カードデテクト端子ディセ-ブル状態

インクルード **#include <Clbsys.h>**

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBSetFnButtons

機能 Fn キー動作の許可／禁止設定を行います。

書式 `BOOL CLBSetFnButtons(BOOL bFnButtons)`

パラメータ `BOOL bFnButtons`
`TRUE` Fn キー有効 (デフォルト)
`FALSE` Fn キー無効

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

CLBGetFnButtons

機能 Fn キー動作の許可／禁止状態を読み出します。

書式 **BOOL** CLBGetFnButtons()

パラメータ 無し

戻り値 **TRUE** Fn キー有効
FALSE Fn キー無効

インクルード **#include <Clbsys.h>**

CLBSetInputChangeButtons

機能 入力切替キー動作の入力設定を行います。

書式 **BOOL** CLBSetInputChangeButtons(**DWORD dwInputChangeButtons**)

パラメータ	DWORD dwInputChangeButtons	
	CLB_BUTTON_NORMAL	通常動作 (デフォルト) 切替可能
	CLB_BUTTON_LOCK_NUM	数字固定
	CLB_BUTTON_LOCK_HIRA	ひらがな固定
	CLB_BUTTON_LOCK_KANA	カタカナ固定 切替不可
	CLB_BUTTON_LOCK_ALPHA	英大文字固定
	CLB_BUTTON_LOCK_ALPHAS	英小文字固定

戻り値 **TRUE** 正常終了

インクルード **#include <Clbsys.h>**

CLBGetInputChangeButtons

機能 入力切替キー動作の入力状態を読み出します。

書式 `DWORD CLBGetInputChangeButtons()`

パラメータ 無し

戻り値	<code>CLB_BUTTON_NORMAL</code>	通常動作 (デフォルト)
	<code>CLB_BUTTON_LOCK_NUM</code>	数字固定
	<code>CLB_BUTTON_LOCK_HIRA</code>	ひらがな固定
	<code>CLB_BUTTON_LOCK_KANA</code>	カタカナ固定
	<code>CLB_BUTTON_LOCK_ALPHA</code>	英大文字固定
	<code>CLB_BUTTON_LOCK_ALPHAS</code>	英小文字固定

インクルード `#include <Clbsys.h>`

CLBSetKeyLock

機能 キーロックの許可／禁止設定を行います。

トリガーキー、電源キー以外のキー押下の許可／禁止設定を行います。

書式 `BOOL CLBSetKeyLock(BOOL bKeyLock)`

パラメータ `BOOL bKeyLock`

TRUE ロック状態(トリガーキー、電源キー以外は無効)

FALSE 非ロック状態(デフォルト)

戻り値 **TRUE** 正常終了

インクルード `#include <Clbsys.h>`

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBGetKeyLock

機能 キーロックの許可／禁止状態を読み出します。

トリガーキー、電源キー以外のキー押下の許可／禁止状態を読み出します。

書式 `BOOL CLBGetKeyLock()`

パラメータ 無し

戻り値 **TRUE** ロック状態(トリガーキー、電源キー以外は無効)

FALSE 非ロック状態

インクルード `#include <Clbsys.h>`

CLBPlayBuzzer

機能 ブザーを鳴らします。

書式 `BOOL CLBPlayBuzzer(DWORD dwType, DWORD dwHelz, DWORD dwTime)`

パラメータ `DWORD dwType` 属性

ブザー音の属性は以下の6種類です。デフォルト値を示します。

	周波数(Hz)	時間(ms)	個別ミュート	属性
タップ音	2200	25	ON/OFF	B_TAP
キークリック音	2600	50	ON/OFF	B_CLICK
アラーム音	2800	150	ON/OFF	B_ALARM
警告音	3000	100	ON/OFF	B_WARNING
読取完了音	3100	75	ON/OFF	B_SCANEND
ユーザー指定音	-	-	ON/OFF	B_USERDEF

`DWORD dwHelz` ユーザ指定音時のみ有効

B_USERDEF 以外の場合は、`CLB_BUZZER_DEFAULT` を使用します。

`DWORD dwTime` ユーザ指定音時のみ有効

B_USERDEF 以外の場合は、`CLB_BUZZER_DEFAULT` を使用します。

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

備考 全音量ミュートが設定されている場合、または鳴らしたい属性の個別ミュートが ON の場合はブザーは鳴りません。

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBStopBuzzer

機能 ブザーを止めます。

書式 `BOOL CLBStopBuzzer()`

パラメータ 無し

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

CLBSetBuzzerVolume

機能 ブザー音量設定を行います。

書式 `BOOL CLBSetBuzzerVolume(DWORD dwType, DWORD dwBuzzerVolume)`

パラメータ `DWORD dwType` 属性

ブザー音の属性は以下の 6 種類です。

	属性
タップ音	<code>B_TAP</code>
キークリック音	<code>B_CLICK</code>
アラーム音	<code>B_ALARM</code>
警告音	<code>B_WARNING</code>
読取完了音	<code>B_SCANEND</code>
通常音	<code>B_USERDEF</code>

`DWORD dwBuzzerVolume`

`CLB_BUZZERVOLUME_MIN` 音量 小

`CLB_BUZZERVOLUME_MID` 音量 中 (デフォルト)

`CLB_BUZZERVOLUME_MAX` 音量 大

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBGetBuzzerVolume

機能 ブザー音量を読み出します。

書式 `DWORD CLBGetBuzzerVolume(DWORD dwType)`

パラメータ `DWORD dwType` 属性

ブザー音の属性は以下の 6 種類です。

	属性
タップ音	<code>B_TAP</code>
キークリック音	<code>B_CLICK</code>
アラーム音	<code>B_ALARM</code>
警告音	<code>B_WARNING</code>
読取完了音	<code>B_SCANEND</code>
通常音	<code>B_USERDEF</code>

戻り値 `CLB_BUZZERVOLUME_MIN` 音量 小

`CLB_BUZZERVOLUME_MID` 音量 中 (デフォルト)

`CLB_BUZZERVOLUME_MAX` 音量 大

インクルード `#include <Clbsys.h>`

CLBSetBuzzerMute

機能 ブザーの全音量、個別ミュートを設定します。

書式 **BOOL CLBSetBuzzerMute(DWORD dwType, BOOL bMute)**

パラメータ **DWORD dwType** 属性

ブザー音の属性は以下の7種類です。

	属性
タップ音	B_TAP
キークリック音	B_CLICK
アラーム音	B_ALARM
警告音	B_WARNING
読取完了音	B_SCANEND
通常音	B_USERDEF
全音量	B_ALL

BOOL bMute

TRUE ミュート ON (ブザーは鳴りません)

FALSE ミュート OFF (ブザーは鳴ります) デフォルト

戻り値 **TRUE** 正常終了

インクルード **#include <Clbsys.h>**

備考 全音量ミュートが設定されている場合、または鳴らしたい属性の個別ミュートが ON の場合はブザーは鳴りません。

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBGetBuzzerMute

機能 ブザーの全音量、個別ミュートを取得します。

書式 `BOOL CLBGetBuzzerMute(DWORD dwType)`

パラメータ `DWORD dwType` 属性

ブザー音の属性は以下の7種類です。

	属性
タップ音	<code>B_TAP</code>
キークリック音	<code>B_CLICK</code>
アラーム音	<code>B_ALARM</code>
警告音	<code>B_WARNING</code>
読取完了音	<code>B_SCANEND</code>
通常音	<code>B_USERDEF</code>
全音量	<code>B_ALL</code>

戻り値 `TRUE` ミュート ON (ブザーは鳴りません)

`FALSE` ミュート OFF (ブザーは鳴ります) デフォルト

インクルード `#include <Clbsys.h>`

備考 全音量ミュートが設定されている場合、または鳴らしたい属性の個別ミュートが ON の場合はブザーは鳴りません。

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBSetCPUMode

機能 CPU 周波数制御の設定を行います。

書式 **BOOL** CLBSetCPUMode(**DWORD** dwMode)

パラメータ	DWORD dwMode	CPU 周波数
	CLB_CPUMODE_LOW	CPU 周波数:200MHz
	CLB_CPUMODE_HIGH	CPU 周波数:400MHz

戻り値 **TRUE** 正常終了

インクルード **#include <Clbsys.h>**

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBGetCPUMode

機能 CPU 周波数制御を読み出します。

書式 `BOOL CLBGetCPUMode(DWORD *pdwMode)`

パラメータ	<code>DWORD *pdwMode</code>	CPU 周波数
	<code>CLB_CPUMODE_LOW</code>	CPU 周波数:200MHz
	<code>CLB_CPUMODE_HIGH</code>	CPU 周波数:400MHz

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBSet180Rotate

機能 表示画像を180度回転させます。

書式 **BOOL CLBSet180Rotate(BOOL bRotate)**

パラメータ **BOOL bRotate**
TRUE 180度回転(反転)
FALSE 0度回転(通常)

戻り値 **TRUE** 正常終了

インクルード **#include <Clbsys.h>**

※エミュレータ 現状、エミュレータ機能としては表示画像を180度回転しません。

CLBGet180Rotate

機能 表示画像の回転状態を読み出します。

書式 **BOOL** CLBGet180Rotate ()

パラメータ 無し

戻り値 **TRUE** 180度回転 (反転)

FALSE 0度回転 (通常)

インクルード **#include <Clbsys.h>**

※エミュレータ 現状、エミュレータ機能としては表示画像を180度回転しません。

CLBGetDeviceIDCode

機能 EEPROM に書き込まれている DeviceID を読み出します。

書式 `BOOL CLBGetDeviceIDCode (TCHAR *pdwDevID)`

パラメータ `TCHAR *pdwDevID[32]`

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

CLBGetUserIDCode

機能 EEPROM に書き込まれている UserID を読み出します。

書式 `BOOL CLBGetUserIDCode (DWORD *pdwUserID)`

パラメータ `DWORD *pdwUserID`

戻り値 `TRUE` 正常終了

インクルード `#include <Clbsys.h>`

CLBPrepareLED

機能LEDの点灯準備を行います。

CLBSetIndicator 関数ではハードの仕様により LED 点灯まで 130ms かかってしまいます。

この関数と CLBUpdateLED を使用することにより、実行後すぐに LED 点灯が出来ます。

書式 **BOOL CLBPrepareLED(DWORD dwLedMode, DWORD dwTime,
DWORD dwOnTime, DWORD dwOffTime)**

パラメータ **DWORD dwLedMode**
CLB_LED_GREEN LED の緑の点灯準備をします。
CLB_LED_RED LED の赤の点灯準備をします。
DWORD dwTime LED 連続点灯時間。(1/16 秒単位)
値は(dwOnTime+dwOffTime)×点滅回数を設定
DWORD dwOnTime LED 点灯時間。(1/16 秒単位) 0~255
DWORD dwOffTime LED 消灯時間。(1/16 秒単位) 0~255

戻り値 **TRUE** 正常終了
FALSE 内部エラーによる異常終了

インクルード **#include <Clbsys.h>**

使用例 LED 橙を素早く 1 回点滅する場合の点灯準備設定
(ハードの仕様により以下の引数 (dwTime,dwOnTime,dwOffTime) が最低の値となります。)
CLBPrepareLED(CLB_LED_GREEN | CLB_LED_RED, 4, 1, 3);
LED 緑を 1 秒 ON、1 秒 OFF を 3 回点滅する場合の点灯準備設定
CLBPrepareLED(CLB_LED_GREEN, 96, 16, 16);
LED 赤を 2 秒点灯する場合の点灯準備設定
CLBPrepareLED(CLB_LED_RED, 35, 32, 3);
LED 赤を連続点灯する場合の点灯準備設定
(連続点灯の場合は dwOnTime,dwOffTime に 0 使用可)
CLBPrepareLED(CLB_LED_RED, 255, 255, 0);

注意事項

LED 点灯中に CLBPrepareLED 関数を設定しますと、LED の設定が変更されてしまいますので、CLBPrepareLED 関数が実行された場合は強制的に LED は OFF します。

※エミュレータ 現状、エミュレータ機能としては表示画像を 180 度回転しません。

CLBUpdateLED

機能 現在の LED 設定値で LED の点灯を行います。

CLBSetIndicator 関数ではハードの仕様により LED 点灯まで 130ms かかってしまいます。
この関数と CLBPrepareLED を使用することにより、実行後すぐに LED 点灯が出来ます。

書式 DWORD CLBUpdateLED(DWORD dwLedMode)

パラメータ DWORD dwLedMode

CLB_LED_OFF	LED を消灯します。
CLB_LED_GREEN	現在の LED 設定値で LED の緑を点灯します。
CLB_LED_RED	現在の LED 設定値で LED の赤を点灯します。

戻り値 **TRUE** 正常終了
FALSE 内部エラーによる異常終了

インクルード #include <Clbsys.h>

使用方法 LED を点灯させる前に CLBPrepareLED 関数を実行し、LED 点灯準備設定を行います。
その後 CLBUpdateLED 関数を実行し LED を CLBPrepareLED 関数にて設定した条件で点灯させます。
(CLBUpdateLED 関数を実行した場合、CLBPrepareLED 関数が実行されてから 130ms 以上経過していない場合は実行されてから 130ms 待ってから LED 点灯処理を行います。)

注意事項 CLBPrepareLED 関数を行わないで CLBUpdateLED 関数を実行した場合は、現在の設定値に基づいて LED を点灯させる為、予期せぬ LED 点灯になりますのでご注意ください。

※エミュレータ エミュレータ機能として実機動作とは異なります。

CLBGetBackupLife

機能 VDET1 発生後のRAMバックアップ時間の設定値を取得します。

CLB_NORMAL_LIFE に設定されていますと、VDET1 発生後電源を OFF した場合、RAM は、約 1.5 日バックアップされます。また、CLB_LONG_LIFE に設定されていますと、同条件で、RAM は、約 3 日バックアップされます。

本関数は、OS ビルドが、4139 以降で動作します。

書式 DWORD CLBGetBackupLife (void)

戻り値	CLB_LONG_LIFE	RAM バックアップ約 3 日
	CLB_NORMAL_LIFE	RAM バックアップ約 1.5 日
	CLB_NOT_SUPPORT	OS バージョンが 4139 でない場合

インクルード `#include <Clbsys.h>`

CLBSetBackupLife

機能 VDET1 発生後のRAMバックアップ時間を設定します。

CLB_NORMAL_LIFE に設定すると、VDET1 発生後電源を OFF した場合、RAM は、約 1.5 日バックアップされます。また、CLB_LONG_LIFE に設定すると、同条件で、RAM は、約 3 日バックアップされます。本関数は、OS ビルドが、4139 以降で動作します。

書式 **DWORD CLBSetBackupLife(DWORD dwSetBackupLife)**

パラメータ **DWORD dwSetBackupLife**

CLB_LONG_LIFE RAM バックアップ約 3 日

CLB_NORMAL_LIFE RAM バックアップ約 1.5 日

戻り値 **CLB_SET_SUCCESS** 正常終了

CLB_SET_ERROR OS バージョンが 4139 でない場合

インクルード **#include <Clbsys.h>**

1.8 レジストリ

システムライブラリでは、以下のレジストリを使用します。これらの値は提供する API 関数により自動的に更新されますので、ユーザが直接編集する必要はありません。

LocalMachine\Drivers\BuiltIn\pmc			
WakeUp要因による電源ONの設定をします。			
キー名	型	値	説明
LeftTrgKeyWakeUpDisable	DWORD	0	左トリガキーによるWake-upを許可
		1	左トリガキーによるWake-upを禁止
RightTrgKeyWakeUpDisable	DWORD	0	右トリガキーによるWake-upを許可
		1	右トリガキーによるWake-upを禁止
IoBoxWakeUpDisable	DWORD	0	IoBox 載せによる Wake-up を許可
		1	IoBox 載せによる Wake-up を禁止

LocalMachine\Drivers\BuiltIn\pmc			
電源 ON 後指定時間 OFF を禁止する時間を設定します。			
キー名	型	値	説明
PowerOffKeyReadyTime	DWORD	非設定(初期値)	電源 ON 後一定時間 OFF キーを無効にする時間を秒単位で設定

LocalMachine\Drivers\BuiltIn\pmc			
電源ボタンでの OFF の禁止/許可を設定します。			
キー名	型	値	説明
PowerOffKeyDisable	DWORD	0	電源キー有効
		1	電源キー無効

LocalMachine\Drivers\BuiltIn\BUZ			
ブザーの個別設定を設定します。			
キー名	型	値	説明
OemTapFreq	DWORD	2200	タップ周波数 [Hz]
OemTapTime	DWORD	25	タップ音長 [ms]
OemTapVol	DWORD	2	タップ音量 [1-3]
OemTapMute	DWORD	0	タップミュート [0:OFF、1:ON]
OemClickFreq	DWORD	2600	クリック周波数 [Hz]
OemClickTime	DWORD	50	クリック音長 [ms]
OemClickVol	DWORD	2	クリック音量 [1-3]
OemClickMute	DWORD	0	クリックミュート [0:OFF、1:ON]
OemAlarmFreq	DWORD	2800	アラーム周波数 [Hz]
OemAlarmTime	DWORD	150	アラーム音長 [ms]
OemAlarmVol	DWORD	2	アラーム音量 [1-3]
OemAlarmMute	DWORD	0	アラームミュート [0:OFF、1:ON]
OemWarningFreq	DWORD	3000	警告音周波数 [Hz]
OemWarningTime	DWORD	100	警告音音長 [ms]
OemWarningVol	DWORD	2	警告音音量 [1-3]
OemWarningMute	DWORD	0	警告音ミュート [0:OFF、1:ON]
OemScanendFreq	DWORD	3100	スキャナ読取完了音周波数 [Hz]
OemScanendTime	DWORD	75	スキャナ読取完了音音長 [ms]
OemScanendVol	DWORD	2	スキャナ読取完了音音量 [1-3]
OemScanendMute	DWORD	0	スキャナ読取完了音ミュート [0:OFF、1:ON]
OemUserdefFreq	DWORD	XXXX	ユーザー定義音周波数 [Hz]
OemUserdefTime	DWORD	XXXX	ユーザー定義音音長 [ms]
OemUserdefVol	DWORD	2	ユーザー定義音音量 [1-3]
OemUserdefMute	DWORD	0	ユーザー定義音ミュート [0:OFF、1:ON]

DT-5100

ライブラリマニュアル for C++

【システムライブラリ編】

Ver3.00

発行元：カシオ計算機株式会社

〒162-8543

東京都渋谷区本町 1-6-2

システムソリューション営業統轄部

TEL:03-5334-4638