

Smart-USB Sigma 製品ファミリ

WinUSB.sys 汎用デバイスドライバ対応

SUSlv.dll API 仕様

(LabVIEW2008 以降に対応)

有限会社プライムシステムズ

2019.04.23 第 6.1 版

SUSlv_Usb_Open

最初に見つかった 1 台の USB デバイスをオープンします

```
BOOL SUSlv_Usb_Open (  
    UINT64      *dFoundHandle,  
    UINT64      *dWinUSBHandle  
);
```

【パラメータ】

*dFoundHandle	オープンした USB デバイスのデバイスハンドルを格納するポインタを指定します。
*dWinUSBHandle	オープンした USB デバイスの WinUSB ハンドルを格納するポインタを指定します。

【戻り値】

USB デバイスを Open できた場合 TRUE を返します。
それ以外は FALSE を返します。

【解説】

対応する USB デバイスが接続されていない場合や、
すでにオープンされている場合は失敗します。

SUSlv_Usb_Close

USB デバイスをクローズします。

```
BOOL SUSlv_Usb_Close (  
    UINT64      dFoundHandle,  
    UINT64      dWinUSBHandle  
);
```

【パラメータ】

dFoundHandle	オープンしたときに取得した USB デバイスのデバイスハンドルを指定します。
dWinUSBHandle	クローズする USB デバイスの WinUSB ハンドルを指定します。

【戻り値】

Close が成功した場合 TRUE を返します。
それ以外は FALSE を返します。

【解説】

オープンしている USB デバイスが抜かれた時にはそのタイミングで必ずクローズを行って下さい。

SUSlv_Get_Device_Handle

ホスト PC に接続しているすべての USB デバイス (ボード) の台数 (個数) とハンドルを取得します。SUSlv_Find_Device 関数は、複数台のボード上のすべての FPGA をコンフィグした状態で使用する関数ですが、この関数では FPGA が未コンフィグの状態の場合に使用します。

```
BOOL SUSlv_Get_Device_Handle (  
    DWORD          *num,  
    UINT64         *dFoundHandle  
);
```

【パラメータ】

*num	指定された GUID を持つ USB デバイスの台数を格納するポインタを指定します。
*dFoundHandle	指定された GUID を持つ USB デバイスハンドルを格納するポインタを指定します。

【戻り値】

情報取得が成功した場合 TRUE を返します。
それ以外は FALSE を返します。

【使用上の注意事項】

1 台のホスト PC に接続する複数台の USB デバイスは、すべて同じ FPGA を搭載したボードにしてください。制御が簡単になります。

SUSlv_Usb_Open_Handle

SUSlv_Get_Device_HandleやSUSlv_Find_Device関数により取得したUSBデバイスハンドルを指定して、USBデバイスをオープン処理します。

※SUSlv_Find_Device()または SUSlv_Get_Device_Handle()と SUSlv_Usb_Open_Handle()を組み合わせて使用してください。

```
BOOL SUSlv_Usb_Open_Handle (  
    UINT64          dFoundHandle,  
    UINT64          *dWinUSBHandle  
);
```

【パラメータ】

dFoundHandle オープンする USB デバイスのデバイスハンドルを指定します。

*dWinUSBHandle 取得した WinUSB ハンドルを格納するポインタを指定します。

【戻り値】

情報取得が成功した場合 TRUE を返します。

それ以外は FALSE を返します。

SUSlv_Find_Device

ホスト PC に接続しているすべての USB デバイスを順番にオープン処理後、FPGA 回路に実装した任意のハードウェア・レジスタを読み出し、指定した INDEX 値と一致する USB デバイスのデバイスハンドルを取得します。

この関数は、ホスト PC に複数台のボード（USB デバイス）を接続するときに使用します。

複数台のすべてのボードは、PC に接続する前に FPGA コンフィグが完了している必要があります（ボード上のコンフィグ ROM を利用して FPGA ブートする必要があります）。

```
BOOL SUSlv_Find_Device (  
    DWORD      adrs,  
    DWORD      id,  
    UINT64     *dFoundHandle  
);
```

【パラメータ】

adrs	USB デバイスを識別するためのレジスタ・アクセスを実施する際に必要なレジスタアドレスを指定します。
id	検索する INDEX 値を指定します。
*dFoundHandle	見つかった USB デバイスのデバイスハンドルを格納するポインタを指定します。

【戻り値】

指定した INDEX 値が見つかった場合 TRUE を返します。

それ以外は FALSE を返します。

SUSlv_Reg_Read

任意のレジスタから値を読み込みます。

```
BOOL SUSlv_Reg_Read (  
    UINT64      dWinUSBHandle,  
    DWORD      adrs,  
    DWORD      size,  
    BYTE       *rd  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
adrs	対象となるレジスタ番号を指定します。 0~2,097,151 まで設定できます。
size	バイト幅を指定します。1 から 5 1 2 バイトまでの範囲です。
*rd	読み込んだ値を受け取るバッファへのポインタを指定します。

【戻り値】

レジスタ読み込みが成功した場合 TRUE を返します。
それ以外は FALSE を返します。

【注意】

vreq_Smart-USB_Sigma.pdf 資料の bRequest=A0h 部分を参照してください。

SUSlv_Reg_Write

任意のレジスタに値を書き込みます。

```
BOOL SUSlv_Reg_Write (  
    UINT64      dWinUSBHandle,  
    DWORD      adrs,  
    DWORD      size,  
    BYTE       *wd  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
adrs	対象となるレジスタ番号を指定します。 0~2,097,151 まで設定できます。
size	バイト幅を指定します。1 から 5 1 2 バイトまでの範囲です。
*wd	書き込むデータバッファへのポインタを指定します。

【戻り値】

レジスタ書き込みが成功した場合 TRUE を返します。
それ以外は FALSE を返します。

【注意】

vreq_Smart-USB_Sigma.pdf 資料の bRequest=A0h 部分を参照してください。

SUSlv_Data_Write

メモリに可変長のデータを書き込みます。

```
BOOL SUSlv_Data_Write (  
    UINT64      dWinUSBHandle,  
    DWORD      FileSize,  
    LPVOID      pBuffer  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
FileSize	書き込むバイト数を指定します。
pBuffer	書き込むデータが入ったバッファのポインタを指定します。

【戻り値】

データ書き込みが成功した場合 TRUE を返します。
それ以外は FALSE を返します。

【注意】

vreq_Smart-USB_Sigma.pdf 資料の bRequest=A1h 部分を参照してください。

SUSlv_Data_Read

メモリから可変長のデータを読みだします。

```
BOOL SUSlv_Data_Read (  
    UINT64      dWinUSBHandle,  
    DWORD      FileSize,  
    LPVOID      pBuffer  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
FileSize	読み出すバイト数を指定します。
pBuffer	読み出すデータが格納されるバッファのポインタを指定します。

【戻り値】

データ読み込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

【注意】

vreq_Smart-USB_Sigma.pdf 資料の bRequest=A1h 部分を参照してください。

SUSlv_Bulk_Write

メモリに可変長のデータを書き込みます。

このコマンドの前後に必ず SUSlv_Mem_Cmd 関数を実行して下さい。

SUSlv_Data_Write 関数は、SUSlv_Mem_Cmd (開始コマンド) + SUSlv_Bulk_Write (1 回) + SUSlv_Mem_Cmd (終了コマンド) で構成しています。

SUSlv_Mem_Cmd コマンドに挟まれた SUSlv_Bulk_Write 関数は何度でも繰り返し発行できます。

BOOL SUSlv_Bulk_Write (

```
    UINT64      dWinUSBHandle,  
    DWORD      WriteSize,  
    LPVOID      pBuffer  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
WriteSize	書き込むバイト数を指定します。
pBuffer	書き込むデータが入ったバッファのポインタを指定します。

【戻り値】

データ書き込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

【注意】

vreq_Smart-USB_Sigma.pdf 資料の bRequest=A1h 部分を参照してください。

SUSlv_Bulk_Read

メモリから可変長のデータを読みだします。

このコマンドの前後に必ず SUSlv_Mem_Cmd 関数を実行して下さい。

SUSlv_Data_Read 関数は [SUSlv_Mem_Cmd (開始コマンド) + SUSlv_Bulk_Read (1 回) + SUSlv_Mem_Cmd (終了コマンド)] で構成しています。

SUSlv_Mem_Cmd コマンドに挟まれた SUSlv_Bulk_Read 関数は何度でも繰り返し発行できます。

```
BOOL SUSlv_Bulk_Read (  
    UINT64      dWinUSBHandle,  
    DWORD      ReadSize,  
    LPVOID      pBuffer  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
ReadSize	読み出すバイト数を指定します。
pBuffer	読み出すデータが格納されるバッファのポインタを指定します。

【戻り値】

データ読み込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

【注意】

vreq_Smart-USB_Sigma.pdf 資料の bRequest=A1h 部分を参照してください。

SUSlv_Mem_Cmd

SUSlv_Bulk_Write および SUlv_Bulk_Read を使用する前後にこのコマンドを使用します。

これらの API 発行直前に SUSlv_Mem_Cmd（メモリアクセス開始表示用）を発行した上で、SUSlv_Bulk_Write または SUSlv_Bulk_Read を複数回実行し、最後に SUSlv_Mem_Cmd（メモリアクセス終了表示用）を発行します。

```

BOOL SUSlv_Mem_Cmd(
    UINT64      dWinUSBHandle,
    BYTE        cmd,
    DWORD       length
);

```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
cmd	SUSlv_DATA_TRANS_OUT_START (SUSlv_Bulk_Write 開始コマンド) SUSlv_DATA_TRANS_OUT_END (SUSlv_Bulk_Write 終了コマンド) SUSlv_DATA_TRANS_IN_START (SUSlv_Bulk_Read 開始コマンド) SUSlv_DATA_TRANS_IN_END (SUSlv_Bulk_read 終了コマンド)
length	データ転送長を指定します。

【戻り値】

データ読み込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

【注意】

vreq_Smart-USB_Sigma.pdf 資料の bRequest=A1h 部分を参照してください。

SUSlv_Get_BoardInfo

ボードの情報を取得します。

```

BOOL SUSlv_Get_BoardInfo (
    UINT64      dWinUSBHandle,
    DWORD       Sel,
    DWORD       *ID
);

```

【パラメータ】

dWinUSBHandle 動作を実行する USB デバイスの WinUSB ハンドルを指定します。

Sel

SUS_BOARDINFO_FIRMID	ファームウェアバージョン情報を選択します。
SUS_BOARDINFO_CHIPVER	チップリビジョン情報を選択します。
SUS_BOARDINFO_PID	プロダクト IC 情報を選択します。
SUS_BOARDINFO_SPEED	PC との USB 接続スピード情報を選択します。
SUS_BOARDINFO_API_BUILD	使用している API のビルド情報を選択します。
SUS_BOARDINFO_API_PATCH	使用している API のパッチ情報を選択します。
SUS_BOARDINFO_API_MINOR	使用している API のマイナー情報を選択します。
SUS_BOARDINFO_API_MAJOR	使用している API のメジャー情報を選択します。

ID Sel パラメータで設定した情報が格納されるポインタを指定します。

【戻り値】

情報取得が成功した場合 TRUE を返します。

それ以外は FALSE を返します。

【注意】

パラメータ Sel の詳細は、vreq_Smart-USB_Sigma.pdf 資料の bRequest=B3h 部分を参照してください。

SUSlv_Usb_Config

FPGA をコンフィグします。 (適用ボード : SX-USB4)

```
BOOL SUSlv_Usb_Config (  
    UINT64      dWinUSBHandle,  
    TCHAR      *PathName  
);
```

【パラメータ】

dWinUSBHandle 動作を実行する USB デバイスの WinUSB ハンドルを指定します。

PathName FPGA コンフィグファイルを指定します

【戻り値】

FPGA コンフィグが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

SUSlv_Usb_Deconfig

コンフィグ状態の FPGA を未コンフィグ状態（コンフィグする前の状態）にします。

```
BOOL SUSlv_Usb_Deconfig (  
    UINT64      dWinUSBHandle  
);
```

【パラメータ】

dWinUSBHandle 動作を実行する USB デバイスの WinUSB ハンドルを指定します。

【戻り値】

FPGA コンフィグが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

SUSlv_Usb_Config_KXCARD7

FPGA をコンフィグします。(適用ボード : KX-Card7、AX-Card7)

```
BOOL SUSlv_Usb_Config_KXCARD7 (  
    UINT64      dWinUSBHandle,  
    TCHAR      *PathName,  
    bool        Encrypt  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
PathName	FPGA コンフィグファイルを指定します
Encrypt	暗号化したコンフィグデータを指定する場合、“1”を指定します。

【戻り値】

FPGA コンフィグが成功した場合 TRUE を返します。
それ以外は FALSE を返します。

【注意】

KX-Card7 および AX-Card7 システム開発ボード以外での利用はできません。

SUSlv_Usb_Config_CXCARD5

FPGA をコンフィグします。(適用ボード : CX-Card5)

```
BOOL SUSlv_Usb_Config_CXCARD5 (  
    UINT64      dWinUSBHandle,  
    TCHAR      *PathName  
);
```

【パラメータ】

dWinUSBHandle 動作を実行する USB デバイスの WinUSB ハンドルを指定します。

PathName FPGA コンフィグファイルを指定します

【戻り値】

FPGA コンフィグが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

【注意】

CX-Card5 システム開発ボード以外での利用はできません。

SUSlv_I2c_Read

USB 制御 IC に接続する I2C ポートを制御します。

ベンダリクエストコード“C1”に対応します。“C4”には対応していません。

詳細は vreq_Smart-USB_Sigma.pdf を参照して下さい。

```
BOOL SUSlv_I2c_Read (  
    UINT64      dWinUSBHandle,  
    DWORD       DevAdrs,  
    DWORD       RegAdrs,  
    BYTE        *Rd  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
DevAdrs	スレーブデバイスのデバイスアドレスを指定します。
RegAdrs	スレーブデバイスのレジスタアドレスを指定します。
*Rd	読み出すデータのポインタを指定します。

【戻り値】

I2C のデータ読み込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

SUSlv_I2c_Write

USB 制御 IC に接続する I2C ポートを制御します。

ベンダリクエストコード“C0”に対応します。“C3”には対応していません。

詳細は vreq_Smart-USB_Sigma.pdf を参照して下さい。

```
BOOL SUSlv_I2c_Write (  
    UINT64      dWinUSBHandle,  
    DWORD       DevAdrs,  
    DWORD       RegAdrs,  
    BYTE        Wd  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
DevAdrs	スレーブデバイスのデバイスアドレスを指定します。
RegAdrs	スレーブデバイスのレジスタアドレスを指定します。
Wd	RegAdrs で指定したレジスタに書込むデータを指定します。

【戻り値】

I2C データ書き込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

SUSlv_I2c_ReadEx

USB 制御 IC に接続する I2C ポートを制御します。

ベンダリクエストコード“C5”に対応します。

詳細は vreq_Smart-USB_Sigma.pdf を参照して下さい。

```

BOOL SUSlv_I2c_ReadEx (
    UINT64      dWinUSBHandle,
    DWORD       port
    DWORD       DevAdrs,
    DWORD       RegAdrs,
    BYTE        *Rd
);

```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
port	ベンダリクエストコードの切替 (0:C1(EXT1)、1:C4(EXT2))
DevAdrs	スレーブデバイスのデバイスアドレスを指定します。
RegAdrs	スレーブデバイスのレジスタアドレスを指定します。
*Rd	読み出すデータのポインタを指定します。

【戻り値】

I2C のデータ読み込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

SUSlv_I2c_WriteEx

USB 制御 IC に接続する I2C ポートを制御します。
 ベンダリクエストコード“C0”と“C3”に対応します。
 詳細は vreq_Smart-USB_Sigma.pdf を参照して下さい。

```

BOOL SUSlv_I2c_WriteEx (
    UINT64      dWinUSBHandle,
    DWORD       port,
    DWORD       DevAdrs,
    DWORD       RegAdrs,
    BYTE        Wd
);

```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
port	ベンダリクエストコードの切替 (0:C1(EXT1)、1:C4(EXT2))
DevAdrs	スレーブデバイスのデバイスアドレスを指定します。
RegAdrs	スレーブデバイスのレジスタアドレスを指定します。
Wd	RegAdrs で指定したレジスタに書込むデータを指定します。

【戻り値】

I2C データ書き込みが成功した場合 TRUE を返します。
 それ以外は FALSE を返します。

SUSlv_I2c_ReadEx2

USB 制御 IC に接続する I2C ポートを制御します。ベンダリクエストコード“C6”に対応します。

詳細は vreq_Smart-USB_Sigma.pdf を参照して下さい。

```

BOOL SUSlv_I2c_ReadEx2 (
    UINT64      dWinUSBHandle,
    DWORD      port
    DWORD      cmd_length,
    DWORD      read_length,
    BYTE       *iic_cmd,
    BYTE       *rd
);

```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
port	ベンダリクエストコードの切替 (0:C1(EXT1)、1:C4(EXT2))
cmd_length	スレーブデバイスへのコマンド長を指定。(*1)
read_length	スレーブデバイスのバースト読み出し長を指定。(*2)
*iic_cmd	スレーブデバイスへのコマンドを設定。
*Rd	スレーブデバイスから読み出すデータのポインタを指定します。

【戻り値】

I2C のデータ読み込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

(*1) (*2)cmd_length と read_renigth の合算値には以下の制限があります。

SuperSpeed 接続時 最大 512 バイト、 HighSpeed 接続時 最大 64 バイト

SUSlv_I2c_WriteEx2

USB 制御 IC に接続する I2C ポートを制御します。

ベンダリクエストコード“C5”に対応します。

詳細は vreq_Smart-USB_Sigma.pdf を参照して下さい。

```

BOOL SUSlv_I2c_WriteEx2 (
    UINT64      dWinUSBHandle,
    DWORD       port,
    DWORD       cmd_length,
    DWORD       write_length,
    BYTE        *iic_cmd
);

```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
port	ベンダリクエストコードの切替 (0:C1(EXT1)、1:C4(EXT2))
cmd_length	スレーブデバイスへのコマンド長を設定します。(*1)
write_length	スレーブデバイスへの書き込みデータ長を指定します。
*iic_cmd	スレーブデバイスに書込むコマンドとデータのポインタを指定します。

【戻り値】

I2C データ書き込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

(*1) 単発のデータ読み出し (SUSlv_I2c_ReadEx2) で指定するコマンド長と一致させてください。

cmd_length と write_renigth の合算値には以下の制限があります。

SuperSpeed 接続時 最大 512 バイト、 HighSpeed 接続時 最大 64 バイト

SUSlv_Get_FPGA_Temp

ボード上の FPGA 温度情報を取得します。(適用ボード : KX-Card7, SX-USB4)

```
BOOL SUSlv_Get_FPGA_Temp (  
    UINT64      dWinUSBHandle,  
    WORD        Sel,  
    WORD        *Temp  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
Sel	温度情報を取得する FPGA を指定します。 常に「0」を指定します。
*Temp	FPGA の温度情報データが返されるポインタを指定します。

【戻り値】

情報取得が成功した場合 TRUE を返します。
それ以外は FALSE を返します。

SUSlv_Temp_Alert

ボード上の FPGA 温度警告 LED 点滅設定値を変更します。デフォルトでは 55℃。ボードリセットによりデフォルト値になります。(適用ボード：KX-Card7、SX-USB4)

```
BOOL SUSlv_Temp_Alert (  
    UINT64      dWinUSBHandle,  
    BOOL        dir,  
    BYTE        *wd,  
    BYTE        *rd  
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
dir	0:温度設定、 1:設定値取得
wd	温度データ設定 (摂氏)
*rd	設定されている温度データが返されるポインタを指定します。

【戻り値】

情報取得が成功した場合 TRUE を返します。
それ以外は FALSE を返します。

SUSlv_FX3Reset

ホスト PC からボード側の USB 動作を初期化することができます。

USB 制御 IC (FX3) のソフトウェアリセットです。FPGA のリセットではありません。

(SX-USB4 ボードを除く)

```
BOOL SUSlv_FX3Reset (  
    UINT64          dWinUSBHandle  
);
```

【パラメータ】

dWinUSBHandle 動作を実行する USB デバイスの WinUSB ハンドルを指定します。

【戻り値】

FX3 (USB 制御 IC) のリセットが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

SUP64_Data_Read

メモリから可変長のデータを読みだします。

Smart-USB Plus 製品シリーズとの互換性があります。

BOOL **SUP64_Data_Read** (

```

    UINT64    dWinUSBHandle,
    BYTE      range,
    DWORD     FileSize,
    LPVOID    pBuffer
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
range	ビット幅を指定します。 SUP64_RANGE_8BIT メモリのバス幅が 8 ビット SUP64_RANGE_16BIT メモリのバス幅が 16 ビット SUP64_RANGE_32BIT メモリのバス幅が 32 ビット SUP64_RANGE_64BIT メモリのバス幅が 64 ビット
FileSize	読み出すバイト数を指定します。512(dec)で割り切れるデータ量を指定してください。 割り切れない場合は、上位アプリケーション側でデータ量の調整を行ってください。
pBuffer	読み出すデータが格納されるバッファのポインタを指定します。

【戻り値】

データ読み込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

SUP64_Data_Write

メモリに可変長のデータを書き込みます。

Smart-USB Plus 製品シリーズとの互換性があります。

BOOL **SUP64_Data_Write** (

```

    UINT64    dWinUSBHandle,
    BYTE      range,
    DWORD     FileSize,
    UINT64    pBuffer
);
```

【パラメータ】

dWinUSBHandle	動作を実行する USB デバイスの WinUSB ハンドルを指定します。
range	ビット幅を指定します。 SUP64_RANGE_8BIT メモリのバス幅が 8 ビット SUP64_RANGE_16BIT メモリのバス幅が 16 ビット SUP64_RANGE_32BIT メモリのバス幅が 32 ビット SUP64_RANGE_64BIT メモリのバス幅が 64 ビット
FileSize	書き込むバイト数を指定します。 512(dec)で割り切れるデータ量を指定してください。 割り切れないデータ量の場合は、上位アプリケーション側でデータ量の調整を行ってください。
pBuffer	書き込むデータが入ったバッファのポインタを指定します。

【戻り値】

データ書き込みが成功した場合 TRUE を返します。

それ以外は FALSE を返します。

API 選択のポイント

