

Smart-USB Plus／Smart-IO2 製品ファミリ

## WinUSB.sys 汎用デバイスドライバ対応

### SUP64.dll API 仕様

有限会社プライムシステムズ

2014.01.14 第 5.1 版

2013.08.01 第 5.0 版

2013.01.18 第 4.0 版

2012.03.01 第 3.0 版

2011.12.28 第 2.0 版

2011.11.29 第 1.0 版

## SUP64\_OpenEx

---

最初に見つかった 1 台の USB デバイスをオープンします

※SUP64\_Find\_Device()または SUP64\_Get\_Device\_Handle()と SUP64\_Usb\_Open\_Handle()を組み合わせて使用してください。

```
BOOL SUP64_OpenEx(  
    GUID                guidDeviceInterface,  
    HANDLE              *FoundHandle,  
    PWINUSB_INTERFACE_HANDLE phWinUSBHandle,  
);
```

### 【パラメータ】

**guidDeviceInterface**

オープンするデバイスのインターフェース GUID を指定します。

**\*FoundHandle**

オープンするデバイスのハンドルを指定します。

**phWinUSBHandle**

取得した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】

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。

### 【参考】

PsiCeb2.dll での API 対応 → CebOpenDevice

## SUP64\_CloseEx

---

デバイスをクローズします。  
SUP64\_OpenEX 関数を使用してオープンしたときには、この関数を使用してください。

```
BOOL SUP64_Usb_Close (  
    HANDLE                FoundHandle,  
    WINUSB_INTERFACE_HANDLE hWinUSBHandle  
);
```

### 【パラメータ】

HANDLE	オープンしたときに取得したデバイスのハンドルを指定します。
hWinUSBHandle	クローズするデバイスの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。

### 【参考】

PsiCeb2.dll での API 対応 → CebCloseDevice

## SUP64\_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。

新規に利用する場合には、SUP64\_OpenEX 関数を使用してください。

※1 台のホスト PC に複数台のボードを接続している場合には、SUP64\_Find\_Device()、SUP64\_Usb\_Open\_Handle()、SUP64\_Get\_Device\_Handle()を使用してください。

```
BOOL SUP64_Usb_Open(  
    GUID                                guidDeviceInterface,  
    PWINUSB_INTERFACE_HANDLE          phWinUSBHandle  
);
```

### 【パラメータ】

guidDeviceInterface

オープンするデバイスのインターフェース GUID を指定します。

phWinUSBHandle

取得した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】

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。

### 【参考】

PsiCeb2.dll での API 対応 → CebOpenDevice

## SUP64\_Usb\_Close

---

デバイスをクローズします。

SUP64\_Usb\_Open 関数でオープンした場合、この関数ではクローズできないバグがあります。できるだけ、SUP64\_OpenEx、SUP64\_CloseEx 関数の組み合わせを利用してください。

```
BOOL SUP64_Usb_Close (  
    WINUSB_INTERFACE_HANDLE hWinUSBHandle  
);
```

### 【パラメータ】

hWinUSBHandle            クローズするデバイスの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。

### 【参考】

PsiCeb2.dll での API 対応 → CebCloseDevice

## SUP64\_Reg\_Read

任意のレジスタから値を読み込みます。

```

BOOL SUP64_Reg_Read (
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,
    DWORD                      adrs,
    DWORD                      range,
    UINT64                     *rd
);

```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスの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。

0~63 または 0~16,383 まで設定できます。(設定範囲はボードにより異なります)

**range**

ビット幅を指定します。指定する値は次のいずれかになります。

SUP64_RANGE_8BIT	レジスタ・ビット幅が 8 ビット
SUP64_RANGE_16BIT	レジスタ・ビット幅が 16 ビット
SUP64_RANGE_32BIT	レジスタ・ビット幅が 32 ビット
SUP64_RANGE_64BIT	レジスタ・ビット幅が 64 ビット

**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。

### 【参考】

PsiCeb2.dll での API 対応 → CebRegisterRead

### 【注意】

vreq\_Smart-USB\_Plus.pdf 資料の bRequest=BOH 部分を参照してください。

## SUP64\_Reg\_Write

任意のレジスタに値を書き込みます。

```

BOOL SUP64_Reg_Write (
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,
    DWORD                      adrs,
    DWORD                      range,
    UINT64                     wr
);

```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスの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。

0~63 または 0~16,383 まで設定できます。(設定範囲はボードにより異なります)

**range**

ビット幅を指定します。指定する値は次のいずれかになります。

SUP64_RANGE_8BIT	レジスタ・ビット幅が 8 ビット
SUP64_RANGE_16BIT	レジスタ・ビット幅が 16 ビット
SUP64_RANGE_32BIT	レジスタ・ビット幅が 32 ビット
SUP64_RANGE_64BIT	レジスタ・ビット幅が 64 ビット

**wd**

書き込むデータを指定します。

### 【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。

### 【参考】

PsiCeb2.dll での API 対応 → CebRegisterWrite

### 【注意】

vreq\_Smart-USB\_Plus.pdf 資料の bRequest=BOH 部分を参照してください。

## SUP64\_GPIF\_Reg\_Wait

---

SUP64\_GPIF\_Reg\_Write や SUP64\_GPIF\_Reg\_Read で行う制御で、制御線とデータ線のタイミングを制御します。

```
BOOL SUP64_GPIF__Wait (  
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,  
    WORD                        dir,  
    BYTE                        num  
);
```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスのハンドルを指定します。

**dir**

レジスタアクセスの RD/WR を指定します。

1: WR

0: RD

**num**

タイミング制御したいクロック数を指定します。0~255 まで。

### 【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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

Smart-IO2 では利用できません。



## SUP64\_Data\_Write

メモリに可変長のデータを書き込みます。

```

BOOL SUP64_Data_Write (
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,
    BYTE                        range,
    DWORD                       FileSize,
    LPVOID                      pBuffer
);

```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスの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。

### 【参考】

PsiCeb2.dll での API 対応 → CebDataWrite

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_Data\_Read

メモリから可変長のデータを読みだします。

```

BOOL SUP64_Data_Read (
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,
    BYTE                        range,
    DWORD                       FileSize,
    LPVOID                      pBuffer
);

```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスの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**

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。

### 【参考】

PsiCeb2.dll での API 対応 → CebDataRead

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_Get\_BoardInfo

---

ボードの情報を取得します。

```
BOOL SUP64_Get_BoardInfo (
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,
    DWORD                      Sel,
    DWORD                      *ID
);
```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスのハンドルを指定します。

**Sel**

SUP64\_BOARDINFO\_FIRMID   ファームウェアバージョン情報を選択します。

SUP64\_BOARDINFO\_CHIPVER   チップリビジョン情報を選択します。

SUP64\_BOARDINFO\_PID       プロダクト IC 情報を選択します。

SUP64\_BOARDINFO\_SPEED    PCとボードのUSB接続スピード情報を選択します。

**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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_Usb\_Config

FPGA をコンフィグします。

```

BOOL SUP64_Usb_Config (
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,
    TCHAR                      *PathName,
    bool                        Mode,
    bool                        Swap,
    bool                        Encrypt
);

```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスの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

**Mode**

以下 2 つの中から、アルテラまたはザイリンクスを指定します。

SUP64\_CONFIG\_MODE\_ALTERA                      アルテラの場合

SUP64\_CONFIG\_MODE\_XILINX                      ザイリンクスの場合

**Swap**

ザイリンクスデバイス (Spartan6) 時に指定します。

**Encrypt**

ザイリンクスデバイス (Spartan6) 時、暗号化したコンフィグファイルを用いるときに指定します。

### 【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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_GPIF\_Set\_Wave

メモリ転送時のモードを設定します。  
使用できるボードは以下の製品です。

CX-Card2 (020 系のみ) , SX-Card3 (0202 以降) , SX-USB3 (0202 以降), ASUP-M3, CX-Card4, SX-Card6

```
BOOL SUP64_GPIF_Set_Wave (
    WINUSB_INTERFACE_HANDLE  hWinUSBHandle,
    word                      dir,
    word                      sel,
);
```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスのハンドルを指定します。

**dir**

RD/WR 方向を指定します。 1:WR、 0:RD

SUP64\_GPIFWAVE\_DIR\_READ: RD  
SUP64\_GPIFWAVE\_DIR\_WRITE: WR

**sel**

転送モードを指定します。 1: Word 転送 0:Burst 転送 (デフォルト)

SUP64\_GPIFWAVE\_MODE\_BURST: Burst 転送(デフォルト)  
SUP64\_GPIFWAVE\_MODE\_WORD: Word 転送

### 【戻り値】

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

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_I2c\_Read

---

USB 制御 IC に接続する I2C ポートを制御します。

```
BOOL SUP64_I2c_Read (  
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,  
    DWORD                      DevAdrs,  
    DWORD                      RegAdrs,  
    BYTE                       *Rd  
);
```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスのハンドルを指定します。

**DevAdrs**

I2C バスに接続するスレーブデバイスのデバイスアドレスを指定します。

**RegAdrs**

I2C バスに接続する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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

## SUP64\_I2c\_Write

---

USB 制御 IC に接続する I2C ポートを制御します。

```
BOOL SUP64_I2c_Write (  
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,  
    DWORD                      DevAdrs,  
    DWORD                      RegAdrs,  
    BYTE                       Wd  
);
```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスのハンドルを指定します。

**DevAdrs**

I2C バスに接続するスレーブデバイスのデバイスアドレスを指定します。

**RegAdrs**

I2C バスに接続する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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

## SUP64\_Get\_FPGA\_Temp

---

ボード上の FPGA 温度情報を取得します。  
対応ボード：VX-USB2、System-SX2、SX-USB3

```
BOOL SUP64_Get_FPGA_Temp (  
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,  
    WORD                        Sel,  
    WORD                        *Temp  
);
```

### 【パラメータ】

**hWinUSBHandle**  
動作を実行するデバイスのハンドルを指定します。

**Sel**

**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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

Smart-IO2 では利用できません。



## SUP64\_Suspend

---

ボード上の FPGA のサスペンド状態を制御します。  
対応ボード： SX-Card6、SX-Card3

```
BOOL SUP64_Get_FPGA_Temp (  
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,  
    WORD                        mode  
);
```

### 【パラメータ】

**hWinUSBHandle**  
動作を実行するデバイスのハンドルを指定します。

**mode**  
0 : Suspend 状態を解除  
1 : Suspend 状態に設定

### 【戻り値】

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

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_GPIO\_Dir\_Read

---

USB 制御 IC の GPIO を制御します。各ポートの入力、出力を切り替えます。  
対応ボード：USB 制御 IC の GPIO が利用できるボード

```
BOOL SUP64_GPIO_Read (  
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,  
    WORD                        port,  
    WORD                        dir  
);
```

### 【パラメータ】

hWinUSBHandle

動作を実行するデバイスのハンドルを指定します。

port

SUP64\_GPIO\_PORTA : Port A

SUP64\_GPIO\_PORTC : Port C

SUP64\_GPIO\_PORTE : Port E

dir

SUP64\_GPIO\_DIRIN: 入力 (IN)

SUP64\_GPIO\_DIROUT: 出力 (OUT)

### 【戻り値】

処理が成功した場合 TRUE を返します。

それ以外は FALSE を返します。

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_GPIO\_Read

---

USB 制御 IC の GPIO を制御します。

対応ボード：USB 制御 IC の GPIO が利用できるボード

```
BOOL SUP64_GPIO_Read (  
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,  
    WORD                        port,  
    BYTE                        *rd  
);
```

### 【パラメータ】

hWinUSBHandle

動作を実行するデバイスのハンドルを指定します。

port

SUP64\_GPIO\_PORTA : Port A

SUP64\_GPIO\_PORTC : Port C

SUP64\_GPIO\_PORTE : Port E

rd

読み込むデータ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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_GPIO\_Write

---

USB 制御 IC の GPIO を制御します。

対応ボード：USB 制御 IC の GPIO が利用できるボード

```
BOOL SUP64_GPIO_Read (  
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,  
    WORD                        port,  
    BYTE                        Wd  
);
```

### 【パラメータ】

hWinUSBHandle

動作を実行するデバイスのハンドルを指定します。

port

SUP64\_GPIO\_PORTA : Port A

SUP64\_GPIO\_PORTC : Port C

SUP64\_GPIO\_PORTE : Port E

Wd

書き込むデータを指定します。

### 【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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_Reg\_WriteEx

任意のレジスタに値を書き込みます。

対応ボード: SX-USB2 (0204 以降), CX-Card2(0204 以降), VX-USB2(0201 以降), SX-USB3 (0204 以降), SX-Card3(0205 以降), ASUP-M3(0202 以降), CX-Card4(0202 以降), SX-Card6(0201)以降  
 ※ ( ) 内の数字はファームウェア版数を示します。

```

BOOL SUP64_Reg_WriteEX (
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,
    DWORD                       adrs,
    DWORD                       size,
    BYTE                        *wd
);
  
```

### 【パラメータ】

**hWinUSBHandle**  
動作を実行するデバイスのハンドルを指定します。

**adrs**  
0~1023(D) まで設定できます。

**size**  
1 バイトから 64 バイトまで指定できます。

**\*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】

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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

vreq\_Smart-USB\_Plus.pdf 資料の bRequest=C6H 部分を参照してください。

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_Reg\_ReadEx

任意のレジスタから値を読み出します。

対応ボード: SX-USB2 (0204 以降), CX-Card2(0204 以降), VX-USB2(0201 以降), SX-USB3 (0204 以降), SX-Card3(0205 以降), ASUP-M3(0202 以降), CX-Card4(0202 以降), SX-Card6(0201 以降)

※ ( ) 内の数字はファームウェア版数を示します。

```

BOOL SUP64_Reg_RaedEX (
    WINUSB_INTERFACE_HANDLE    hWinUSBHandle,
    DWORD                       adrs,
    DWORD                       size,
    BYTE                        *rd
);

```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスのハンドルを指定します。

**adrs**

0~1023(D) まで設定できます。

**size**

1 バイトから 64 バイトまで指定できます。

**\*rd**

読み出すデータ・バッファへの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。

### 【参考】

PsiCeb2.dll での API 対応 → なし

### 【注意】

vreq\_Smart-USB\_Plus.pdf 資料の bRequest=C6H 部分を参照してください。

### 【注意】

Smart-IO2 では利用できません。

## SUP64\_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 SUP64_Find_Device(  
    GUID                    guidDeviceInterface,  
    DWORD                  adrs,  
    DWORD                  id,  
    HANDLE                 *FoundHandle  
);
```

### 【パラメータ】

**guidDeviceInterface**

オープンするデバイスのインターフェース GUID を指定します。

**adrs**

ボード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。

**\*FoundHandle**

見つかった USB デバイスのハンドル・バッファへの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。

### 【参考】

PsiCeb2.dll での API 対応 → CebFindDevice

## SUP64\_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。SUP64\_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 SUP64_Get_Device_Handle(  
    GUID  
    DWORD  
    HANDLE  
);  
    guidDeviceInterface,  
    *num,  
    *FoundHandle
```

### 【パラメータ】

**guidDeviceInterface**

オープンするデバイスのインターフェース GUID を指定します。

**\*num**

指定された GUID を持つ USB デバイスの台数へのポインタを指定します。

**\*FoundHandle**

指定された GUID を持つ USB デバイス・ハンドルへの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ボードは、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。

### 【参考】

PsiCeb2.dll での API 対応 → CebGetDeviceID



---

## SUP64\_Usb\_Open\_Handle

---

SUP64\_Get\_Device\_Handle や SUP64\_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。

```
BOOL SUP64_Usb_Open_Handle(  
    HANDLE                               FoundHandle  
    PWINUSB_INTERFACE_HANDLE            phWinUSBHandle  
);
```

### 【パラメータ】

FoundHandle

オープンするデバイスのハンドルを指定します。

phWinUSBHandle

取得した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。

## SUP64\_Usb\_Reg\_Burst\_Write

**Smart-IO2 専用**のバーストレジスタライト関数です。  
最大 64 バイトを一括転送できます。

```
BOOL SUP64_Reg_Burst_Write(    WINUSB_INTERFACE_HANDLE hWinUSBHandle,  
                                DWORD adrs,  
                                DWORD length,  
                                BYTE *Wd  
                                );
```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスの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。0~7、18 (LED) のいずれかを指定します。  
アドレスと Smart-IO2 ボードのポートマップは、Smart-IO2 ボードのマニュアル第 11 章「レジスタマップ」を参照して下さい。

**length**

バーストライトするとバイト数を 1~64 の範囲で指定します。

**\*wd**

書き込むデータが保存されている BYTE 配列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。

## SUP64\_Reg\_WriteExIO2

**Smart-IO2 専用**の高速版レジスタライト関数です。

Smart-USB Plus 製品では同様の機能を SUP64\_Reg\_WriteEx()関数で提供しています。

※ SUP64\_Reg\_Wright 関数より、制御ソフトウェアからボード I/O ピンまでの制御時間を短縮します。設計値では約 1/2 です。

```
BOOL SUP64_Reg_WriteExIO2(    WINUSB_INTERFACE_HANDLE hWinUSBHandle,  
                             DWORD adrs,  
                             BYTE wd  
                             );
```

### 【パラメータ】

**hWinUSBHandle**

動作を実行するデバイスの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。0~7、18 (LED) のいずれかを指定します。アドレスと Smart-IO2 ボードのポートマップは、Smart-IO2 ボードのマニュアル第 11 章「レジスタマップ」を参照して下さい。

**wd**

書き込むバイトデータを指定します。

### 【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。

---

## 削除した API

---

PsiCeb2.sys デバイスドライバを利用して制御ソフトウェアをカスタマイズしているユーザ向けの情報です。

PsiCeb2.sys デバイスドライバを WinUSB.sys デバイスドライバに入れ替えた場合、以下の API は利用できなくなりますのでご注意ください。

### **CebSetStartRegister**

【PsiCeb2.dll での API 対応】 → SUP64\_Reg\_Write に置き換え

### **CebSetEndRegister**

【PsiCeb2.dll での API 対応】 → SUP64\_Reg\_Write に置き換え

### **CebCheckMyDevice**

### **CebInitLog**

### **CebGetLog**

### **CebGetTransferTime**

### **CebControlTransfer**

### **CebGetDeviceID** (SUP64\_Usb\_Open に包含)

### **CebSetLog**

### **CebLastTransferTime**

### **CebSetTimeOut**

### **CebSetSize**

### **CebInterruptTransfer**

## API 選択のポイント (USB Open 関連)

