

## AZ9360 SDK USB Device Driver 使用方法

### 1. はじめに

AZのUSBデバイスドライバは、簡易的なCOMクラスのデバイスとして使用ができます。

また、PCからCOMポートに見せるためには、付属のデバイスドライバをPCに登録する必要があります。

### 2. インストール

#### 2. 1 USBデバイスドライバのインストール

##### (1) ドライバファイルのインストール

添付のusbdev.zipを解凍すると以下のファイルが解凍されます。

```

¥itron ---+--- usbdevlib ---+--- usbreg.h
                                +--- usbdef.h
                                +--- usb_task.h
                                +--- usb_d.h
                                +--- usb_task.c
                                +--- USBDEV.a
  
```

“usbdevlib”フォルダ以下を開発環境のフォルダ“itron”の下にコピーしてください。

##### (2) az9360.cfgの修正。

現在、ご使用されているaz9360.cfgに以下の行を追加してください。

```

INCLUDE ("¥usb_d.h¥");
INCLUDE ("¥usb_task.h¥");

~

CRE_TSK( TASKID_USBD_EP1,      { TA_HLNG | TA_ACT, 2, UsbD_EpInOut, PRI_USBD_EP, STACK_USBD_EP,
NULL });
CRE_TSK( TASKID_USBD_EP2,      { TA_HLNG | TA_ACT, 3, UsbD_EpInOut, PRI_USBD_EP, STACK_USBD_EP,
NULL });
CRE_TSK( TASKID_USBD_WR, { TA_HLNG | TA_ACT , 0, USB_Write_Task , PRI_USBD_WR, STACK_USBD_WR,
NULL });
CRE_TSK( TASKID_USBD_RD, { TA_HLNG | TA_ACT , 0, USB_Read_Task, PRI_USBD_RD, STACK_USBD_RD, NULL });

~

CRE_SEM( SEMID_USBMPL , { TA_TFIFO, 1, 1 } );

~

CRE_FLG( EVF_USB_OPNCLS, { TA_TFIFO | TA_CLR , 0 } );

~

CRE_MPF( MPFID_USBD_SEND, {TA_TFIFO | TA_MFIFO, 256, 1024, NULL});
  
```

### (3) makefile の修正。

現在、ご使用されている makefile に以下の行を追加／編集してください。

```
#
# ソースファイルのディレクトリの定義
#
USBDDIR = ../../usbdevlib

~

UTASK_COBJS = $(UNAME).o CPU_IO.o PS_TELNET.o HTTPD_CGI.o GSP.o NS9XXX_NIC.o ¥
              :
              (省略)
              :
              usb_task.o

~

INCLUDES := -I. -I$(SRCDIR)/include $(INCLUDES) -I$(EXTINC) -I$(APLDIR) -I$(EFSDIR) -I$(AZSYSDIR)
-I$(USBDDIR)

~

UTASK_LIBS = $(SRCDIR)/../TCP/ysock.a$(SRCDIR)/../EFFSLIB/EFFSFAT.a ¥
             $(SRCDIR)/../usbdevlib/USBDEV.a

~

#
# ソースファイルのあるディレクトリに関する定義
#
vpath %.c $(KERNEL_DIR):$(STASK_DIR):$(UTASK_DIR):$(EXTDIR):$(APLDIR):$(EFSDIR):$(USBDDIR)
```

## 2. 2 PC側デバイスドライバのインストール

前項のUSBドライバをリンク後、AZ9360に image.bin を書き込み実行させ、USBケーブルでPCに接続します。

その際に、付属のPC用USBドライバを使用し、USBドライバをインストールします。

```
¥PC ドライバ --+- azusbdevD.sys
                +- azusbdevS.sys
                +- azusbdev.inf
```

### 3. 使用方法

#### 3. 1 データの送信 (CPU→USB)

CPUからUSB側へのデータを送信する場合は、usb\_dev\_write() を使用します。

```
■ ER usb_dev_write(S32 dmaIndex, void * buffer, U32 length)
```

■ 引数

```
U32 dmaIndex      : DMA Channel  
void *dataString  : 書込みバッファ  
int strLen        : 書込みバッファ長
```

■ 戻り

```
終了ステータス (E_OK : 正常終了)
```

※送信処理をタスク内で行う場合は、usb\_task.c にある、USB\_Write\_Task() で、他のタスクからメッセージを受信し、そのデータを送信することが可能です。

#### (1) 送信バッファ

データを送信する際に、上記 API で指定する書き込みバッファは、メモリプールからアロケートしたメモリを使用してください。

使用するプールの ID は、"MPFID\_USBD\_SEND" です。

メモリのアロケートは、以下の通り。

```
ER = get_mpf( MPFID_USBD_SEND , (VP)&pBuff );
```

尚、送信バッファは、送信完了後に返却されます。(usb\_task.c にある、writeCallBackFunc() が、送信完了時にコールされます。

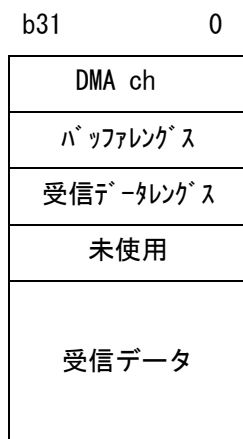
### 3. 2 データの受信 (USB→CPU)

USB側からCPUへデータを受信する場合は、usb\_task.cにある、USB\_Read\_Task()にて受信データを一旦受け取りますので、その後に必要な処理を行ってください。

USB\_Read\_Task()には、サンプルとして受信データをコンソールに出力するようになっています。

#### (1) 受信データ構造

USB\_Read\_Task()に通知されるバッファは以下のようにになっています。



### 3. 3 デバイス名、ベンダID、プロダクトIDなど

ベンダ名、デバイス名、ベンダIDとプロダクトIDなどは、評価用に暫定値になっていますので、製品時には変更してください。

IDは、usb\_task.cの72行目以降に定義してありますので、そこを変更してください。

```
const char MANUFACTURER[] = "YDK Technologies Company"; /* スtringディスクリプタ (ベンダ名) */
const char DEVICE_STRING[] = "USB Serial Driver "; /* スtringディスクリプタ (デバイス名) */
const char SERIAL_NUMBER[] = "00000001"; /* スtringディスクリプタ (シリアルNo.) */

const U8 VID[] = {0x01, 0xff}; /* ベンダーID (仮) 正式なIDを設定してください。*/
const U8 PID[] = {0x01, 0x00}; /* プロダクトID (仮) 正式なIDを設定してください。*/
```