Webブラウザなどの部品があらかじめ用意されている組み込みOS

Windows Embedded CE用 USBモデム・ドライバの製作（後編）

江島 午郎
清水 浩行

Windows Embedded CE（Windows CE）を使ったデバイス・ドライバ開発の解説の後編である．今回は，実際にUSB接続モデム用のデバイス・ドライバを作成し，動作させるまでを説明する．ドライバの作成にあたっては，CPUベンダが提供するBSP（Board Support Package）を入手しておく必要がある．（編集部）

今回使用するモデムは，USB接続タイプのオムロン製アナログ・モデム「ME5614U2」です．通信速度は56kbpsで，ATコマンドで制御するごく一般的なモデムです．対応OSはWindows Server 2003/XP/2000/Me/98SEとなっています．当然のことながら，Windows CE用のドライバは用意されていません．

以下では，このUSB接続モデムを動作させるためのWindows CEのデバイス・ドライバを製作していきます．

● ビルドのためのマクロが必要

Windows CEの開発環境では，ソース・コードをコンパイル，リンクして実行形式ファイルを作成することをビルドと呼び，その過程をビルド・プロセスと呼んでいます．デバイス・ドライバをビルドするためには，このビルド・プロセスにソース・コードを認識させ，その方法を指示する必要があります．そのために作成するのがsourcesファイルです．

リスト1は，USB CDC ACMクラス（USBコミュニケーション・デバイス・クラスのAbstract Controlモデル）のドライバをビルドするためのsourcesファイルです．sourcesファイルは，どのようにソース・コードをコンパイルしてリンクするかを指示するビルド・マクロを記述した，単純なテキスト・ファイルです．そのため，ソース・コードと同じようにテキスト・エディタなどで作成できます．

リスト1　sourcesの定義例

```
SYNCHRONIZE_DRAIN=1
TARGETNAME=cdcacm
TARGETTYPE=DYNLINK
RELEASETYPE=PLATFORM
DLLENTRY=DllEntry
PREPROCESSDEFFILE=1
TARGETDEFNAME=cdcacm
DEFFILE=$(TARGETDEFNAME).def
CDEFINES=$(CDEFINES) -DUSE_NEW_SERIAL_MODEL
WINCEOEM=1
TARGETLIBS = ¥
    $(_SYSGENSDKROOT)¥lib¥$(_CPUINDPATH)¥coredll.lib ¥
    $(_SYSGENOAKROOT)¥lib¥$(_CPUINDPATH)¥usbd.lib ¥
    $(_SYSGENOAKROOT)¥lib¥$(_CPUINDPATH)¥usbclient.lib
SOURCELIBS=¥
    $(_SYSGENOAKROOT)¥lib¥$(_CPUINDPATH)¥com_mdd2.lib
INCLUDES= ¥
    $(_PUBLICROOT)¥COMMON¥DDK¥INC; ¥
    $(_PUBLICROOT)¥COMMON¥SDK¥INC; ¥
    $(_PUBLICROOT)¥COMMON¥OAK¥INC; ¥
    $(_PUBLICROOT)¥COMMON¥OAK¥DRIVERS¥USB¥CLASS¥COMMON; ¥
    $(_PUBLICROOT)¥COMMON¥OAK¥DRIVERS¥SERIAL¥COM_MDD2;
SOURCES= usbdio.c cdcacm.c winceusb.c
```

ビルド・マクロの文法はいたってシンプルです．例えば，**リスト1**のTARGETNAMEのように，<マクロ名>=<マクロ値>と書くだけです．また，**リスト1**のTARGETLIBSのように<マクロ値>が長くなりすぎて1行に記述しきれない場合は，\（バック・スラッシュ，日本語環境の場合は¥になる）を書いていったん改行できます（ビルド・プロセスでは1行と認識される）．

使用可能なビルド・マクロの詳細な解説はWindows CEの開発環境のドキュメントに譲るとして，ここではそのいくつかを簡単に説明します．

TARGETNAMEは実行形式のファイル名を指示します．**リスト1**ではcdcacmと指示しているので，ビルドの結果，cdcacm.dllが作成されます．TARGETTYPEは実行形式の種類を指示します．ドライバの場合はDLL形式なので，DYNLINKと指示します．そのほかの形式として，LIBLARYとPROGRAMがあります．ちなみに，PROGRAMの場合は実行形式ファイルの拡張子が.EXEとなります．

TARGETLIBS，SOURCELIBSには，リンクするライブラリ・ファイルのパスとファイル名を指示します．INCLUDESには，インクルード・パスを複数指示できます．最後に，SOURCESにはビルドの対象となるソース・コード・ファイルを指示します．

sourcesファイルは一つのフォルダに一つしか置けません．複数の実行形式を作成する場合は別のフォルダを作り，同じようにsourcesファイルを定義します（各フォルダの

KEYWORD ── Windows Embedded CE，USBモデム，ME5614U2，Sizka PRO，Geode LX 800，BSP，デバイス・ドライバ，カタログ，Platform Builder，ICE

直下には一つのsourcesファイルが存在する）．この場合，複数のフォルダをビルド対象とすることをビルド・プロセスに認識させるためにdirsファイルが使われます．dirsファイルも，sourcesファイルと同じくテキスト・ファイルです．文法も簡単で，DIRS=<フォルダ名1> <フォルダ名2> … <フォルダ名n> とするだけです．dirsファイルはsourcesファイルが置かれたフォルダの1階層上のフォルダに置きます．

Windows CEでは，このようにsources/dirsで構成したフォルダのツリー構造をビルド・ツリーと呼んでいます．Windows CEのビルドは，すべてこの方法で行われています．

● ドライバの構造と主要関数を把握する

USB CDC ACMクラス・ドライバの構造は，本稿の前編の図3に示しました．一見して分かるように，シリアル・ポート・ドライバMDD（Model Device Driver）とシリアル・ポート・ドライバPDD（Platform-Dependent Driver）の二つから構成されています．MDDの部分は，前編で述べたようにMicrosoft社よりソース・コード付きで提供されています．つまり，今回製作したソース・コードは，前編の図3の灰色の部分を実装するためのものということになります．**表1**は，USB CDC ACMクラス・ドライバのソース・ファイルの一覧です．

このPDDの部分をさらに詳しく見てみると，三つの部分から構成されています．

一つ目はシリアル・ポートPDD関数の実装部で，HWInit，HWPostInit，HWDeinit，…のような一連のPDDエントリ関数がソース・ファイルcdcacm.c，cdcacm.hに書かれています．HWxxxといったエントリ関数名から想像できるように，単純なシリアル・ポート・ドライバの場合は本来，ここで8251や16550のようなSIO（Serial Input Output）チップを制御することになります．一方，USBシリアルの場合は，USBパケットに変換する仕組みになります．

二つ目はUSBデバイスのプラグ&プレイに関する部分で，USBInstallDriver，USBUninstallDriver，USBDeviceAttachなどのエントリ関数がソース・ファイルwinceusb.c，winceusb.hに書かれています．エントリは三つありますが，実質的に役目を果たしているのはUSBDeviceAttachだけです．

三つ目はUSB転送処理などにあたる通信処理部分で，ソース・ファイルusbdio.c，usbdio.hに書かれています．ここで具体的なUSB要求パケットを生成して，**表2**にあるようなIssueVendorTransfer，IssueBulkTransfer，IssueInterruptTransferを呼び出しています．これらの関数の先はUSBDI（USB DriverInterface）になります．誌面の関係で，すべての関数や処理経路について説明することは難しいので，一部について例を挙げて説明します．

例えば，上位のアプリケーション・プログラムまたはデバイス・ドライバが信号線DTR（Data Terminal Ready）を制御する場合を考えてみましょう．上位側はCreateFileによって開いたCOM*n*（*n*は1～9の番号）ポートに対してDTRの制御を行うべく，EscapeCommFunctionというWin32 APIにSET_DIRという指示を出します．Windowsの内部モジュールではこれを受けて，DeviceIoControl関数よりIOCTL_SERIAL_SET_DIRを発行します．

USB CDC ACMクラス・ドライバでは，これをCOM_IOControlエントリ関数で受け止めます．これはMDDの仕事です．MDDは，さらにそれをcdcacm.cのPDD関数

表1　ソース・ファイル一覧

ファイル名	説　明
cdcacm.c	HWInit，HwDeinitなどのシリアル・ポート・ドライバPDDインターフェース関数の実装
cdcacm.h	
winceusb.c	USBInstallDriver，USBDeviceAttachなどのUSBドライバのロード/アンロードにかかわる処理の実装
winceusb.h	
usbdio.c	CDC ACMクラスの仕様に従ったUSB転送処理などの実装
usbdio.h	
makefile	ビルドに必要なファイルで，内容は決まっている
sources	ビルド・プロセスに指示するためのビルド・マクロが記述されたテキスト・ファイル
cdcacm.def	ドライバのエントリ関数を公開（エクスポート）するための定義
cdcacm.bib	ドライバの実行形式cdcacm.dllをOSイメージに組み込む指示を定義したファイル
cdcacm.reg	ドライバのロードに必要なレジストリ設定が定義されているテキスト・ファイル
cdcacm.pbpxml	cdcacm.bibとcdcacm.regをビルド・プロセスに指示するための定義ファイル
cdcacm.pbcxml	Windows CEの開発環境であるPlatform Builderのカタログに登録するための定義ファイル

表2　USBクライアント・ドライバの転送関数

関　数	説　明
IssueVendorTransfer	ベンダ固有のコントロール転送を実行する
IssueBulkTransfer	バルク転送を実行する