
インターフェイスの街角 — WonderWitch

増井 俊之

ゲームマシンと携帯端末

この連載でも何回かとりあげましたが、最近は PalmPilot やザウルス、PocketStation などのように、ユーザー自身の手でプログラムの開発ができる携帯計算機が増えてきています。

段落を入れ替えました

そんななかで、ゲーム機メーカーのバンダイ¹が同社の携帯ゲーム「WonderSwan」のプログラム開発キット「WonderWitch」を発売しました(写真 1~2)。これはキュート²が開発したもので、同社の「Freya OS」を含む専用カートリッジとコンパイラ、接続ケーブルなどがセットになっています。Windows 98/2000 上で開発したプログラムを WonderWitch カートリッジにダウンロードすれば、市販のゲームと同じように WonderSwan を使うことができます。

WonderWitch には以下のような特徴があります。

- 実用的な割に価格が安い

WonderWitch は定価 16,800 円ですが、WonderSwan 本体は 3,000 円台で購入できます(定価は 4,800 円) 専用カートリッジは定価 3,980 円です。つまり、専用携帯端末を数千円で作れることになります。PalmPilot やザウルスは端末だけで数万円しますから、どうしても 1 台の機械をいろいろな場面で使いたくなります。しかし、数千円であれば特定の用途に絞った専用携帯端末として割り切って考えてもよいでしょう。たとえば、WonderSwan を電子ブックビューアに使ったり、自転車に付けてサイクル・コンピュータにしたり、

¹ <http://www.bandai.co.jp/>

² <http://www.qute.co.jp/>

写真 1 WonderSwan



写真 2 WonderWitch



あるいは目覚まし時計に使うこともできます。

- ゲーム機メーカーが積極的にサポートしている
任天堂のゲームボーイにもアングラの開発キットがあるそうですが、当然のことながらメーカーはサポートしていません。これに対し、WonderWitch はバンダイが正式にサポートする開発キットです。

- コマンドラインの開発環境

最近の PC では、エディタやコンパイラ、デバッガ、リソースエディタなどを 1 つの大きなプログラムで扱

う、いわゆる“統合開発環境”が増えてきています。組み込みシステムの開発でも、このような環境を利用することが多いようです。たしかに、ビジュアルな画面配置やデバッグには統合システムが便利な場面もあります。しかし、このようなシステムでは専用のエディタを使わねばならなかったり、Makefile を自由に書けなかったりと、柔軟性に欠けるにもかかわらず使い方が複雑で困ることがよくあります。

WonderWitch では、伝統的なプログラミング開発手法に即して、普通のエディタで編集したプログラムをコンパイル、リンクして転送するという方法でプログラム開発をおこないます。

仕様

WonderSwan と WonderWitch の仕様は以下のようになっています。

WonderSwan のハードウェア

- CPU : 80186 互換(クロック 3MHz)
- 2.49 インチ FSTN 反射型液晶(224 × 144 ドット、8 階調)
- シリアル通信ポート (9,600/38,400bps)
- 操作ボタン(12 個)
- キャラクタとスプライトによる描画
- 4 チャンネル音源
- 外形寸法 : 7.43 × 12.1 × 2.43cm(単 3 電池使用時)
- 重量 : 110g

WonderWitch 専用カートリッジ

カートリッジには 512KB フラッシュメモリ、64KB スタティック・メモリ、バックアップ電池が入っています。メモリには、WonderWitch の中核である Freya BIOS と Freya OS、アプリケーション、メモリ・ファイルシステムなどが収められています。

- Freya BIOS : 画面、キー、サウンド、シリアル通信、タイマー、システム/バンク制御など、ハードウェアに近いレベルの処理をおこないます。
- Freya OS : プロセスやメモリ上のファイルシステム、ライブラリなどを管理します。
- ライブラリ: プロセス管理、標準入出力、MML (Music

図 1 TransMagic



Macro Language) によるサウンド制御などのライブラリが用意されています。

- アプリケーション : ファイル操作、通信、プログラム起動などをおこなうための「MEG」というアプリケーションが用意されています。

開発環境

WonderWitch のプログラムは、Windows 98/2000 の DOS プロンプトで動く LSI C または Turbo C で開発します。これらは MS-DOS の初期のころによく使われた懐かしのコンパイラですが、8086/80186 が PC でほとんど使われなくなった現在では、LSI C-86 のスモールモデル(試食版) や Turbo C は無償で使えるようになっています。

これらは MS-DOS 用のプログラムですが、Cygwin 環境で動かせば、UNIX 上のプログラム開発とほとんど同じように作業を進めることができます。さらに、Linux の DOS エミュレータ dosemu などでも動くと思います。

LSI C または Turbo C で開発した WonderWitch プログラムは、形式を変更したあと、WonderWitch に付属の TransMagic という Windows プログラムでカートリッジに転送します(図 1)

シェル

WonderSwan に WonderWitch のカートリッジを挿して電源を入れると、“シェル”の MEG が動き、ファイル操作や通信、プログラム起動などの操作ができるようになります。

図 2 は、MEG のプログラム起動画面(ランチャ画面)

図 2 MEG のランチャ画面



図 3 MEG の通信画面



図 4 端末ソフトウェアからの通信

```

meg> speed
125 STARTING
38400
200 OK
meg> dir /
125 STARTING
total 4 files
      dr---      64 2000-04-01 00:00:00 @rootfs
kern      dr---      36 2000-04-01 00:00:00 @kernfs
rom0     drw-- 393216 2000-04-01 00:00:00 @hardfs
ram0     drw-- 65536 2000-04-01 00:00:00 @softfs
.
200 OK
meg> ls /ram0
125 STARTING
total 5 files
rever
luna
snake
junk
test
.
200 OK
meg> ls /rom0
125 STARTING
total 35 files
luna
rever
.....
test
string
.
200 OK
meg>

```

です。起動したいプログラムのところにカーソルを移動し、WonderSwan の A ボタンを押すとプログラムが起動します。

図 3 は MEG の通信画面です。TransMagic でファイル転送などをおこなうときは、このモードにしておく必要があります。

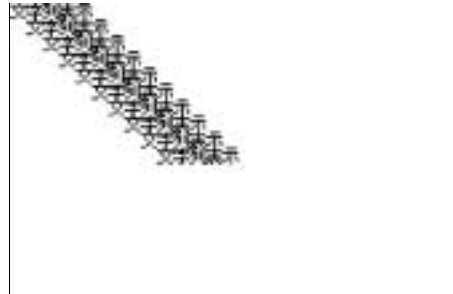
TransMagic と通信モードの MEG とのあいだのやりとりは、すべて RS232C ケーブルを介した無手順シリア

ル通信でおこなわれ、ファイル転送にはその上の XMODEM プロトコルが使われます。TransMagic 以外の端末ソフトウェアで WonderSwan に接続することも可能で、図 4 のように FTP と似た使い方ができます。TransMagic は Windows でしか動きませんが、XMODEM を利用すれば、それ以外のプラットフォームでも WonderWitch カートリッジとデータをやりとりすることができます。

リスト 1 文字列描画プログラム string.c

```
#include <sys/bios.h>
static char *str = "文字列表示";
main()
{
    int i;
    text_screen_init(); /* 画面初期化 */
    for(i=0;i<10;i++){
        text_put_string(i,i,str);
    }
    key_wait(); /* キー入力を待つ */
}
```

図 5 文字列描画結果



コンパイル/実行例

ごく簡単なプログラムの例をいくつか紹介します。

文字列の表示

リスト 1 は、WonderSwan 上に文字列を描くプログラムです。`Hello, World!`並みに、簡単にプログラムを書けるのが分かるのではないのでしょうか。

DOS プロンプトや Cygwin の bash を使い、以下のように string.c をコンパイル、リンクします(これは LSI C-86 試食版を使った例です)

```
% lcc86 -c -o string.obj string.c
% lcc86 -acrct0jpn2.obj -k-M -o string.bin
string.obj
```

このように生成した実行ファイル string.bin は、ランチャで使うアイコン名などの情報を付加してから XMODEM プロトコルで WonderWitch のカートリッジに送ります。情報は、下記のようなコンフィギュレーション・ファイルに記述しておきます。

```
% cat string.cf
name: string
info: 文字列
mode: 7
source: string.bin
output: string.fx
%
```

データ変換プログラム mkfent の引数にこのコンフィギュレーション・ファイルを指定して実行すると、転送ファイル string.fx が生成されます。

```
% mkfent string.cf
mkfent: string.fx 551 bytes (5 blocks)
%
```

リスト 2 キャラクタ描画プログラム

```
#include <sys/bios.h>

unsigned short bgpat[] = {
    0xf000, 0xf000, 0xf000, 0xf000,
    0x0f00, 0x0f00, 0x0f00, 0x0f00
};

void main()
{
    screen_fill_char(SCREEN1, 0, 0, 32, 32, 0);
    font_set_colordata(0, 1, bgpat);
    screen_set_scroll(SCREEN1, 0, 0);
    palette_set_color(0, 0x7530);
    lcd_set_color(0x6420, 0xfca8);
    key_wait();
}
```

図 6 キャラクタ描画結果



このプログラムを WonderSwan に転送して実行すると、図 5 のような表示が得られます。

キャラクタとスクリーン

WonderSwan の画面は、任意の点や線を描けるビットマップ画面ではなく、8 × 8ドットのキャラクタ・フォントを 32 × 32 文字並べたものです。画面には、この 256 × 256 の領域のうちの 244 × 144 の領域が表示されます。

図 7 効果音プログラム

```
#include <sys/bios.h>
#include <il_sound.h>

SoundIL soundIL;
extern BYTE *_heap;

void main()
{
    if(open_sound_il(&soundIL) != E_FS_SUCCESS){
        return;
    }
    sounddrv_init();
    sound_open();
    parse_mml(_heap, "CDEFG", 0);
    bgm_play(_heap, PLAY_LOOP);

    key_wait();

    sound_close();
    sounddrv_release();
}
```

リスト 2 は、市松模様を WonderSwan に描画するプログラムです。フォントを定義したあと、その文字でスクリーンを埋めつくすことにより市松模様が画面に表示されます(図 6)。

サウンド

さきほど紹介したように、WonderWitch では MML を使ってサウンドを鳴らすことができます。図 7 は“ドレミファソ”の音列を繰り返し生成し続けるプログラムです。

UNIX での開発

残念ながら、WonderWitch には Windows 用の開発環境しか用意されていません。しかし、前述のように XMODEM でプログラムの転送ができるので、コンパイラさえ用意すれば UNIX 上でも開発することができます。Intel アーキテクチャ用の Linux であれば、dosemu という DOS エミュレータが使えます。私自身は試していませんが、この上で LSI C などのコンパイラを動かせば WonderWitch での開発も可能でしょう。その他の UNIX でも、DOS エミュレータが用意されていれば開発に利用できます。

WonderWitch の長短

WonderWitch の利点

WonderWitch の最大の利点は、あまり多くのものを必要としないことです。UNIX 上で PalmPilot などのプログラムを開発するには、GCC などの“重厚な”システムが必要で、不慣れな人にはそれだけでもかなり敷居が高いのではないのでしょうか。これに対し、WonderWitch で使われるコンパイラやサポートシステムは非力な Windows マシン上で動き、コンパイラや転送システムをすべて含めても必要なディスクは数 MB 程度に収まります。価格も WonderSwan 本体と開発キットを全部合わせて 2 万円以下ですから、携帯端末のプログラミングを手軽に体験できます。

PalmPilot などにはファイルシステムがなく、プログラミング・スタイルも UNIX などとかなり異なるため、いろいろと勉強してからでないプログラミングにとりかかれません。一方、WonderWitch では画面にテキストを表示する程度のプログラムなら数行で書けますし、ファイルシステムや標準入出力が用意されているので、文字端末のためのプログラミングができる人なら簡単にプログラムが作れます。かつての UNIX 用の端末ゲームなどは、すぐに移植できるでしょう。

WonderWitch の限界

値段が安く手軽に開発できる反面、いろいろな点でやや“貧乏”な感じがあるのは否めません。

- CPU

3MHz の 80186 は、クロックが 1GHz の PC も登場しようという現在ではかなり見劣りがします。最近は携帯電話でも数 MB のメモリをもっているのに、メモリが全部で 512KB(ユーザー使用領域は 384KB)というのも悲しい感じですが、セグメントサイズが 64KB という制限もあります。

- コンパイラ

デフォルトでは、ポインタは 2 バイトで表現されるので注意が必要です。グローバル変数はデータセグメント (DS) に置かれ、ローカル変数はスタックセグメント (SS) に置かれますが、2 バイトのポインタではこれ

らを区別することができないので、以下のようなプログラムの場合には問題が生じます。

```
int i = 10, j;  
char *p;  
p = &i;  
j = *p;
```

変数 *i* は SS に置かれますが、ポインタ *p* は 2 バイトで表現されるため、3 行目の代入によって *p* には *i* の SS 内のオフセットが格納されます。ところが、4 行目の `*p` では *p* は DS 内のオフセットとして扱われてしまうので、DS と SS の位置が異なる場合には、まったく関係のない値が *j* に代入されてしまいます。

`char` の代わりに “char far” としてポインタをすべて 4 バイトで扱えばこのような問題は発生しませんが、速度やメモリをケチるとこのような問題が起こります。

●開発環境

最近の組込みシステム開発の世界では、コンパイラやエミュレータ、デバッガ、ICE などをすべて含む統合開発システムが増えているようです。しかし、WonderWitch にはエミュレータもデバッガもありません。printf などを使って地道にデバッグする必要があります。

●OS

ファイルシステムはあるものの階層はなく、ファイルは 100 個程度しか作れません。ユーザープロセスは 2 個しか動かず、TCP/IP のような通信ライブラリはありません。

●周辺機器

WonderSwan にはボタンとシリアル端子が付いているだけなので、入力装置として複雑な使い方はできません。画面はキャラクタとスプライトにもとづいているため、任意の点や直線の描画はかなり面倒です。シリアルポートは、9,600bps と 38,400bps しかサポートしていません。したがって、4,800bps が一般的な GPS 受信装置や、2,400bps の MindStorms の IR タワーなどは接続できません。

とはいえ、携帯端末にこのような “貧乏” はつきものですし、そのような環境のなかでいろいろと工夫を重ねていくのが味わい深い楽しみ方かもしれません。

アイデア

WonderWitch を何に使うかはアイデアの見せ所でしょう。ハンダイのサイトには、自分だけのゲームを作ろうとか PIM を作ってみようなどという話がかかれてます。もともとがゲーム機なので、ゲームを自作するのが “王道” かもしれませんが、そういう話ならば Palm などでもかまわないでしょう。むしろ、安さや軽さを活かした応用のほうが魅力的だと思います。

PalmPilot では、たくさんのメモリに大量のアプリケーションを入れて使っている人が多いようです。その場合、起動するアプリケーションを選ぶのに苦労することがあります。一方、WonderSwan 程度の値段であれば、アプリケーションごとに 1 台の端末を割り当てるという使い方も十分に考えられます。たとえば、目覚まし時計用に 1 台、キッチンタイマーに 1 台、電子ブック用に 1 台、小遣い管理にもう 1 台…… というふうに、非力なマシンでもたくさん使えば、汎用の PIM でなんでもこなすよりも気分は “富豪” になるでしょう。1 台の PC であらゆる仕事をしようとしたり、1 台の携帯端末ですべての情報を管理しようなどと考えず、仕事ごとに端末を 1 台使うという発想でいけばおもしろいアイデアが浮かびそうです。

おわりに

WonderWitch は発売されて日が浅いにもかかわらず、一部ではかなりの盛り上がりを見せ、多くの Web サイトが開設されています。今後、その利点を活かしたあっと驚くような応用が出てきてほしいものです。

(ますい・としゆき ソニー CSL)

[参考 URL]

- [1] ハンダイのサポートページ
<http://www.swan.channel.or.jp/wonderwitch/>
- [2] ユーザーによるポータルサイト
<http://www.WonderWitch.com/>
- [3] ウィッチの森
http://members.tripod.co.jp/Pochi_/
- [4] WonderWitch 同盟
http://hp.vector.co.jp/authors/VA014524/witch_allies.html