

電子取説表示(M2d)ライブラリ プログラマーガイド

2009-08-06

任天堂株式会社発行

このドキュメントの内容は、機密情報であるため、
厳重な取り扱い、管理を行ってください。

目次

1	はじめに	4
2	開発環境	4
2.1	必要な開発環境	4
2.2	ファイルの構成	4
2.2.1	ライブラリファイル	4
2.2.2	ヘッダファイル	4
2.2.3	電子取説バイナリファイル	4
2.2.4	ライブラリ用リソースファイル	4
3	ライブラリの組み込み	5
3.1	OSの初期化	5
3.2	電子取説表示前に必要となる前処理	5
3.2.1	タッチパネル	5
3.2.2	サウンド	5
3.3	M2dライブラリの初期化	6
3.4	電子取説の表示	7
3.5	終了処理	7
3.6	電子取説表示終了後に必要となる復帰処理	7
3.7	複数言語の対応	8
3.7.1	表示される言語の決定	8
3.7.2	言語を明示的に指定する場合	8
4	注意事項	9
4.1	アーカイブ識別名	9
4.2	IRQ割り込みハンドラ	9
4.3	自動電源OFF	9

コード

コード 3-1	OSの初期化	5
コード 3-2	電子取説表示前の前処理	5
コード 3-3	ファイルパスによるM2dライブラリの初期化	6
コード 3-4	メモリ上のデータによるM2dライブラリの初期化	7
コード 3-5	電子取説の表示	7
コード 3-6	終了処理	7
コード 3-7	電子取説表示後の復帰処理	8
コード 3-8	言語を明示的に指定する場合	9

改訂履歷

改訂日	改訂内容
2009-08-06	必要な開発環境の説明を修正。
2009-05-20	サウンドの必要なチャンネル数について追記。
2009-02-23	TWL-SDK, TWL-System のバージョン番号の変更。
2009-02-09	誤字を修正。
2008-12-22	<ul style="list-style-type: none"> 電子取説バイナリファイルに複数の言語の電子取説が入っている場合の言語の決定方法についての説明を追加。 注意事項を追加。(アーカイブの識別名と割り込みハンドラ、自動電源 OFF について) ライブラリ用リソースファイルの説明からリージョンに関する説明を削除。
2008-11-19	初期化関数の引数の変更による説明の修正。(フォントの配列)
2008-11-06	必要となる前処理に関して追記。
2008-10-29	必要となる復帰処理と注意事項に関して追記。
2008-10-15	説明を追加。
2008-10-06	初版。

1 はじめに

M2d ライブラリは電子取説の表示を行うライブラリです。本ドキュメントでは、アプリケーションに M2d ライブラリを組み込むために必要なことについて説明しています。

個々の関数の詳細な説明については、関数リファレンスを参照してください。

2 開発環境

2.1 必要な開発環境

M2d ライブラリでは次の開発環境が必要となります。バージョンについては、[docs/ReleaseNotes.html](https://docs.tenacious.jp/ManualTools/ReleaseNotes.html) の「ソフトウェア要件」を参照してください。

- TWL-SDK
- TWL-System

2.2 ファイルの構成

2.2.1 ライブラリファイル

ライブラリファイルは lib/ARM9-TS ディレクトリの下にあります。

- libntmvm2d.*a

2.2.2 ヘッダファイル

必要な関数用のヘッダファイルのインクルードは、

```
#include <ntmv/m2d.h>
```

で行えます。

2.2.3 電子取説バイナリファイル

電子取説バイナリファイルは、TWL-ManualEditor にてプロジェクトをバイナリ出力することで生成できます。詳しくは、TWL-ManualEditor のドキュメントを参照してください。

2.2.4 ライブラリ用リソースファイル

M2d ライブラリは、ユーザーインターフェースの表示等に使用するリソースファイルを必要とします。このファイルは、data ディレクトリの下にあります。

- m2dres_narc.blz

3 ライブラリの組み込み

M2d ライブラリをアプリケーションに組み込む手順を示します。

3.1 OSの初期化

必要に応じて OS の初期化を行います。m2d ライブラリではファイル関数を利用するため、FS_Init0を呼び出して置いてください。

コード 3-1 OS の初期化

```
void
NitroMain
{
    OS_Init();

    // vblank 割り込みを有効
    OS_SetIrqFunction(OS_IE_V_BLANK, VBlankIntr);
    OS_EnableIrqMask(OS_IE_V_BLANK);
    GX_VBlankIntr(TRUE);

    // ARM7との通信FIFO割り込み許可
    OS_EnableIrqMask(OS_IE_SPFIFO_RECV);

    OS_EnableIrq();

    FS_Init(FS_DMA_NOT_USE); // ファイルシステム初期化
    ...
}
```

3.2 電子取説表示前に必要となる前処理

3.2.1 タッチパネル

M2d ライブラリでは内部でタッチパネルのオートサンプリング機能を使用しています。そのため、アプリケーション側でタッチパネルのオートサンプリング機能を使用している場合は、M2d ライブラリを使用する前にタッチパネルのオートサンプリングを停止してください。

コード 3-2 電子取説表示前の前処理

```
// タッチパネルのオートサンプリングを停止
TP_RequestAutoSamplingStop();
```

3.2.2 サウンド

M2d ライブラリでは内部で TWL-System の NITRO コンポーザ(Snd)を使用しています。呼び出し側で NITRO コンポーザを使用されている場合は、電子取説の表示に入る前にサウンド再生、キャプチャは停止しておいてください。また、マスターボリュームは最大(デフォルト)にしておいてください。M2d ライブラリ側では、マスターボリュームの変更を行いません。

また、以下の関数を使ってチャンネルを確保している場合、電子取説内のサウンドが鳴らないことがあります。

電子取説の表示に入る前に、チャンネルの解放を行ってください。電子取説で必要なチャンネル数は 4 です。空ける

チャンネル番号に指定はありません。

- SND_LockChannel
- NNS_SndLockChannel
- NNS_SndArcStrmAllocChannel
- NNS_SndStrmAllocChannel
- NNS_SndWaveOutAllocChannel

3.3 M2dライブラリの初期化

M2d ライブラリの初期化を行います。初期化を行うには、TWL-System の NNSFndAllocator と、電子取説バイナリデータ、M2d 用リソースデータ、および TWL-System の NNSG2dFont 構造体のポインタ配列を初期化関数に渡します。電子取説バイナリデータと M2d 用リソースデータは ROM ファイルへのパス文字列で指定する方法と、あらかじめメモリ上に展開されたデータのアドレスを指定する方法の 2 種類があり、それぞれ初期化関数は NTMV_M2dInitForFilePath と NTMV_M2dInitForMem となっています。次にファイルパスで指定する関数 NTMV_M2dInitForFilePath の使い方を示します。

コード 3-3 ファイルパスによる M2d ライブラリの初期化

```
NNSFndAllocator    gAllocator;           // メモリアロケータ

...
// 電子取説を表示するために使用する本体内蔵フォントデータが設定された
// NNSG2dFont構造体へのポインタ配列。
NNSG2dFont* fonts[3];
...

// M2dライブラリが使用する情報を格納する構造体
NTMVM2dManualViewerInfo manViewInfo;

// 初期化します。
NTMV_M2dInitForFilePath(
    &manViewInfo,
    &gAllocator,
    "/data/manpages_narc.blz", // 電子取説バイナリファイル
    "/data/m2dres_narc.blz",  // リソースファイル
    fonts);
```

次にメモリ上に展開されたデータを指定する関数 NTMV_M2dInitForMem の使い方を示します。電子取説バイナリファイルおよび M2d ライブラリ用リソースファイルは、TWL-SDK のツール compBLZ にて圧縮されており、NTMV_M2dInitForMem に渡すメモリ上のデータは、TWL-SDK の関数 MI_SecureUncompressBLZ であらかじめ展開しておく必要があります。さらに、M2d ライブラリ用リソースファイルは 32 バイト境界に配置する必要があります。

コード 3-4 メモリ上のデータによる M2d ライブラリの初期化

```
// 初期化します。

FSFile manFile;
FSFile resFile;

// 電子取説バイナリファイルの読み込み
FS_InitFile(&manFile);
FS_OpenFile(&manFile, "/data/manpages_narc.blz");
...
//      compBLZで圧縮されたデータを展開します。
MI_SecureUncompressBLZ(manData, manCompBytes, manExtractBytes);
...
// M2dライブラリ用リソースファイルの読み込み
FS_InitFile(&resFile);
FS_OpenFile(&resFile, "/data/m2dres_narc.blz");
...
//      compBLZで圧縮されたデータを展開します。
//      resDataが指すメモリは32バイト境界に配置されている必要があります。
MI_SecureUncompressBLZ(resData, resCompBytes, resExtractBytes);
...

NTMV_M2dInitForMem(
    &manViewInfo,
    &gAllocator,
    manData, // 電子取説バイナリデータ
    resData, // リソースデータ
    fonts);
```

3.4 電子取説の表示

電子取説の表示を行うには、NTMV_M2dRun を呼び出します。一旦この関数を呼び出すと、ユーザが「説明書を閉じる」を押して電子取説の閲覧を終了するまで制御が返りません。

コード 3-5 電子取説の表示

```
// マニュアルビューアを実行します。
NTMV_M2dRun(&manViewInfo);
```

3.5 終了処理

M2d ライブラリの終了処理を行います。

コード 3-6 終了処理

```
// マニュアルビューアの終了処理を行います。
NTMV_M2dFinalize(&manViewInfo, &gAllocator);
```

3.6 電子取説表示終了後に必要となる復帰処理

M2d ライブラリでは内部で以下の TWL-SDK の API を呼び出して設定を変更しています。そのため、電子取説の表示が終了した後は適宜再設定を行ってください。

- グラフィックス
- タッチパネル

また、サウンドに関して M2d ライブラリ内部で TWL-System の NITRO コンポーザ(Snd)を使用しています。そのため NITRO コンポーザ(Snd)を使用されている場合は、NNS_SndArcSetCurrent(),NNS_SndArcPlayerSetup() を呼び出して再設定してください。

コード 3-7 電子取説表示後の復帰処理

```
// グラフィックスハードウェアの再設定
GX_Init();

...
// タッチパネルの再設定
TP_RequestAutoSamplingStart(vcount, frequency, samplingBufs, bufSize);

...
// サウンドの再設定
NNS_SndArcSetCurrent(gameSoundArc);
NNS_SndArcPlayerSetup(gameSoundHeap);
```

3.7 複数言語の対応

電子取説バイナリファイルには複数の言語を含むことができます。その場合に表示する電子取説の言語がどのように決定されるかを説明します。また、明示的に表示する言語を指定する方法についても説明します。

3.7.1 表示される言語の決定

本体のオーナー情報に設定されている言語の電子取説が含まれている場合はその言語で表示します。無い場合は TWL-ManualEditor にてプロジェクトをバイナリ出力する際に出力対象にチェックしたものの中で最初に選択されている言語が表示されます。

例えば、バイナリ出力時に英語とフランス語が選択されていた場合、オーナー情報に設定されている言語によって次のようになります。

- 英語の場合は英語。
- フランス語の場合はフランス語。
- スペイン語の場合は英語(スペイン語が含まれていないので、先頭に来る言語に決定されます)。

3.7.2 言語を明示的に指定する場合

TWL-ManualEditor では、オランダ語など本体のオーナー情報で定義されていない言語の電子取説も作成することができます。電子取説バイナリファイルに複数の言語の電子取説が含まれていて、その中にオーナー情報で定義されていない言語の電子取説が含まれる場合は、オーナー情報の言語の情報だけでは表示する電子取説の言語を決定することができません。そのような場合は、アプリケーション側で言語を適切に選択していただき、M2d 初期化関数に表示する言語を指定するようにしてください。M2d ライブラリの初期化関数で関数名の末尾が Ex で終わるものは電子取説で表示する言語を明示的に指定することができます。

コード 3-8 言語を明示的に指定する場合

```
// 言語の選択
u16 lang;
switch ( selectLang )
{
case LANG_EN: lang = 'en'; break; // 英語
case LANG_NL: lang = 'nl'; break; // オランダ語
...
}

// 初期化します。
NTMV_M2dInitForFilePathEx(
    &manViewInfo,
    &gAllocator,
    "/data/manpages_narc.blz", // 電子取説バイナリファイル
    "/data/m2dres_narc.blz",  // リソースファイル
    fonts,                    // フォント
    lang);                    // 表示する言語
```

4 注意事項

4.1 アーカイブ識別名

電子取説表示ライブラリでは、電子取説バイナリファイルとライブラリ用リソースファイルをファイルシステムにマウントしています。その際、識別名は以下のものを使用しています。

- 電子取説バイナリファイル ... M2M
- ライブラリ用リソースファイル ... M2U

4.2 IRQ割り込みハンドラ

電子取説表示ライブラリでは IRQ の割り込みの設定を変更していません。そのため、TWL-SDK の OS_SetIrqFunction 関数で割り込みハンドラを登録している場合は、電子取説の表示中(NTMV_M2dRun 関数の呼び出し中)においても割り込みが発生するたびに割り込みハンドラが呼び出される点に注意してください。

4.3 自動電源OFF

TWL-SDK のパワー制御関数 PM_SetAutoExit にて FALSE が設定されている場合は、自動電源 OFF 処理に対応するために電子取説表示ライブラリの初期化関数内で一時的に TRUE に設定しています。そして終了処理関数で元の設定に戻しています。

PM_SetAutoExit が TRUE の状態（既定の状態）の場合は、電子取説表示ライブラリ側では何も行いません。

Microsoft、Windows は米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

その他、記載されている会社名、製品名等は、各社の登録商標または商標です。

© 2007–2009 Nintendo

任天堂株式会社の許諾を得ることなく、本書に記載されている内容の一部あるいは全部を無断で複製・複写・転写・頒布・貸与することを禁じます。