

KLCD_KOR_SD : 78K0R マイコン用拡張ドライバ・キット説明書

Copyright 2010 てきーらサンドム

(1) 概要

- ・ SD カード , 音声録音 / 再生用のドライバ・キットです。
- ・ KLCD_78K0R (78K0R マイコン用ドライバ・キット) に追加して使用します。
- ・ **個人の研究用として公開しています。(7)の注意事項を熟読の上で使用願います。**

(2) インストール/アンインストール

- ・ KLCD_78K0R と同じフォルダに解凍してください。
- ・ アンインストールする場合は , 解凍したファイルを削除してください。

(3) ファイル/フォルダ構成

KLCD_KOR_SD.pdf	本書
KLCD_KOR_SD.prw	開発環境 (PM+) 起動用のファイル (78K0R/KE3 ~ KG3 用)
KLCD_KOR_SD.pri	同上プロジェクト・ファイル

driver_SD	
— KLCD_KOR_SD.lib	ドライバ・ライブラリ (78K0R/KE3 ~ KG3 用) (*1)
— BTIMER.c	定周期処理 (*2)
— driver_list.h	ドライバ組み込み用ヘッダ (*3)
— SD.*	SD カード・ドライバおよびヘッダ
— CODEC.*	音声コーデック制御ドライバおよびヘッダ
— ADPCM.*	ADPCM 音声圧縮/伸長ドライバおよびヘッダ
— main_9-1A.c	SD カード・アクセス用サンプル・プログラム
— main_9-2A.c	音声録音/再生用サンプル・プログラム
other	
— project.bak	プロジェクト・ファイルの控え

なお , KLCD_78K0R (R1.40 以上) の下記フォルダも必要です。

driver

補足 : KLCD_78K0R との違い

- *1 : fopen/fclose 関数が SD カード対応になっています。
- *2 : 同じです。このフォルダの driver_list.h を有効にするために入れてあります。
- *3 : SD.h をインクルードしています。

(4) SD カード・ドライバ(ヘッダ: SD.h)

SD カードの読み書きを C 言語標準関数 (getc, putc など) で行うためのドライバです。簡易ファイル・システムのためパソコンとのデータ交換は直接出来ません。データ交換が必要な場合はシリアル通信によりアップロード/ダウンロードして下さい。

2GBまでのノーマルSDカードに対応しています。買うなら2GBのSDカードを推奨します。

準備

次の関数を定期的呼び出すようにしてください。本キットの例では BTIMER.c で呼び出しています。

```
int SD_check(void);
```

戻り値が0の場合は、なるべく連続して呼び出してください。戻り値が EOF の場合は、次の 10ms 周期で呼び出して下さい。

このドライバは、SD カードの挿入を検知すると、SD カード・アクセスのための初期化を自動的に行います。初期化中に fopen を実行すると NULL が返ってきます。初期化が終わっているかどうかは、次の関数(実体はマクロ)で確認できます。

```
_Bool SD_status(void);
```

初期化が終了していれば '真', 終わってなければ '偽'です。

ファイル・オープン(書き込み時)

fopen 関数でファイルを開きます。1度に1ファイルしか開けません。

```
FILE *fopen("SD:nnn", "wb");
```

ここで、nnn はファイルを書き始めるクラスタ番号です。本ファイル・システムは、1クラスタ 32KB で管理しています。2GB の SD カードの場合はクラスタ番号は 0 ~ 65535 の範囲になります。オープンできなかった場合は、NULL が返ってきます。

注意: カードの最大容量を判定していないので、最大値は自分で管理してください。

書き始めのクラスタの先頭 512 バイトはファイル・サイズなどの管理データを記録しているため、先頭クラスタは書き込めるデータが 512 バイト減っています。後続のクラスタは 32KB まるまる書き込めます。

なお、管理データはクローズ時 (fclose 実行時) に書き込むため、クローズしなかった場合は、書き込まれたデータは失われます(メディア上には書き込まれていますが、管理データが無いと読み出し用のオープンでエラーになります)。

fopen を使わずに直接ドライバ関数を呼び出してオープンすることも出来ます。

```
FILE *SD_WR_open(unsigned int c);
```

引数 c にクラスタ番号を格納します。オープンできなかった場合は、NULL が返ってきます。

ファイル・オープン（読み出し時）

fopen 関数でファイルを開きます。1 度に 1 ファイルしか開けません（読み出しと書き込みで 1 ファイルづつは可）。

```
FILE *fopen("SD:nnn", "rb");
```

ここで、nnn はファイルを読み始めるクラスタ番号です。オープンできなかった場合（管理データが書き込まれていなかった場合）は、NULL が返ってきます。

注意：カードの最大容量を判定していないので、最大値は自分で管理してください。

正常にオープンできた場合は、次の関数でファイル・サイズを知ることが出来ます。

```
unsigned long SD_RD_size(void);
```

fopen を使わずに直接ドライバ関数を呼び出してオープンすることも出来ますが、戻り値が SD_OPEN_RETRY だった場合は再呼び出しが必要です。次のように呼び出してください。

```
while( (fp = SD_RD_open(c)) == SD_OPEN_RETRY);
```

ただし c にはクラスタ番号を格納します。

オープンできなかった場合は、NULL が返ってきます。

データの書き込み

putc 関数、fputs 関数で書き込みが可能です。本ライブラリの fputs 関数は、バッファが満杯の場合は、自動的に空くまで待ちます。putc 関数はバッファが満杯の場合は、EOF を返すので、プログラム側で待つ処理を記述する必要があります。

データの読み出し

getc 関数、fgets 関数で読み出しが可能です。本ライブラリの fgets 関数は、バッファに 1 バイトでもデータがあると、`¥n` または指定バイト数を読み取るまで待ち状態になります。**従って、`¥n` で終わらないデータが書き込まれたファイルを fgets 関数で読まないでください。**

なお、getc、fgets でエラーが返ってきてもファイル終了と判定しないでください。たんにストリームへの格納が遅れている可能性があります。ファイル終了判定は次のようにしてください（SD_RD_check を必ず先に書いてください）。

```
if (SD_RD_check() == EOF && (c = getc(fp)) == EOF) ファイル終了処理;
```

```
if (SD_RD_check() == EOF && fgets(s, n, fp) == NULL) ファイル終了処理;
```

ファイル・クローズ

次の関数でクローズします。

```
int fclose (FILE* fp);
```

fclose を使わずに直接ドライバ関数を呼び出してクローズすることも出来ますが、戻り値が SD_CLOSE_RETRY だった場合は再呼び出しが必要です。次のように呼び出して下さい。

```
while (SD_WR_close() == SD_CLOSE_RETRY);  
while (SD_RD_close() == SD_CLOSE_RETRY);
```

サンプル・プログラム (main_9-1A.c) の使い方

パソコンのターミナル・ソフトウェアからファイル操作 (ファイル一覧取得, データの読み書き) が出来ます。コマンドは次のとおりです。

- dir または dir c1, c2
2GB 全領域またはクラスタ番号 c1 ~ c2 の範囲のファイルの一覧を表示します。
途中で止めたい場合は, キー基板上的 BS キーを押して下さい。

- CD:nnn
クラスタ番号 nnn にファイル先頭が存在する場合はファイル・サイズを表示します。

- RD:nnn
テキスト・ファイルを読み出します。全て読み終わると EOF と表示します。
途中で止めたい場合は, キー基板上的 BS キーを押して下さい。

- rD:nnn
バイナリ・ファイルを読み出します。1 バイトを 16 進数 2 桁で表示します。
途中で止めたい場合は, キー基板上的 BS キーを押して下さい。

- WD:nnn
テキスト・ファイルを書き込みます。最後の行で EOF と入力すると終了します。

- wD:nnn
バイナリ・ファイルを書き込みます。1 バイトを 16 進数 2 桁で入力します。
改行しても改行コードは入力されません。最後の行で EOF と入力すると終了します。

(5) 音声録音 / 再生ドライバ (ヘッダ: CODEC.h, ADPCM.h)

音声入出力を行うコーデック制御部 (CODEC.c) と音声データの圧縮 / 伸長部 (ADPCM.asm) から構成しています。圧縮 / 伸長に時間がかかるため、次のように直接ストリームへ書き込み / 読み出しを行っています。

- ・録音の場合は、コーデックから受け取ったデータを ADPCM 圧縮してストリーム (SD カードまたは UART のファイル・ストリーム) に直接書き出しています。
- ・再生の場合は、ストリームから直接読み出したデータを ADPCM 伸長してコーデックへ出力しています。

準備

CODEC_open 関数を使ってコーデックの初期化を行います。次のように書きます。

```
while (CODEC_open() == CODEC_RETRY);
```

初期化完了まで 0.5 秒以上かかります。

録音開始

次の関数で録音を開始します。

```
int CODEC_record(FILE* fp);
```

fp は、SD カードの書き込みストリームあるいは UART の送信ストリームへのポインタです。あらかじめオープンしておく必要があります。UART の場合は動作未確認ですが、最低 57.6kbps 以上の速度が必要と思います。開始に失敗すると EOF を返してきます。

再生開始

次の関数で再生を開始します。

```
int CODEC_play(FILE* fp);
```

fp は、SD カードの読みだしストリームあるいは UART の受信ストリームへのポインタです。あらかじめオープンしておく必要があります。UART の場合は動作未確認ですが、最低 57.6kbps 以上の速度が必要と思います。開始に失敗すると EOF を返してきます。

録音 / 再生の停止

次の関数で停止します。

```
int CODEC_stop(void);
```

なお、SD カードからの再生の場合は、ファイルの終端に達すると自動的に停止します。停止したかどうかは次の関数で調べられます。

```
_Bool ADPCM_status(void);
```

戻り値が真なら終了です。

コーデックの使用終了

次の関数でコーデックの使用を終了します。

```
int CODEC_close(void);
```

これにより、録音/再生だけでなくライン入力のミキシング・アンプも停止します。

方式概要

音声は 8kHz, 16 ビット・サンプリングで入出力しています。

このデータを ADPCM 圧縮で 1/4 にしています。従って 1 秒間当りのデータは 4000 バイトになります。2GB の SD カードであれば 150 時間近くの録音が可能です。

ADPCM 圧縮と ADPCM 伸長は同じ変数領域を使うため、録音と再生を同時に行うことは出来ません。

また、R1.40 では、録音ボリュームは固定です。ALC (自動レベル制御) 機能は使用していません。

サンプル・プログラム (main_9-2A.c) の使い方

パソコンのターミナル・ソフトウェアから SD カードへの録音/再生の操作が出来ます。

次のコマンドは、SD カードのサンプル・プログラムと同じなので、(4) を参照してください。

```
dir, CD:nnn, rD:nnn, wD:nnn
```

• PD:nnn

クラスタ番号 nnn の録音データを再生します。ファイル終端で自動的に停止します。

途中で止めたい場合は、キー基板上の BS キーを押して下さい。

音量を下げたい場合は、キー基板上の ' - ' キーを押して下さい。

音量を上げたい場合は、キー基板上の ' + ' キーを押して下さい。

• AD:nnn

録音データをクラスタ番号 nnn から書き込みます。

録音を止めたい場合は、キー基板上の BS キーを押して下さい。

キーを押さない場合も、1 時間で自動的に停止します。

(6) SD カード, オーディオ・コーデックの解説

仕様や使い方の説明については本書では省略します。

(7) 仕様, 制限事項, 注意事項, 等

本ドライバの使用許諾条件。

下記 ~ に納得・同意できる方だけに使用許諾します。

SD カード・ドライバに関する注意

SD カードの仕様書を手に入れるにはライセンスが必要のようです。本版は Web で見つけた断片的な情報やプログラムを参考にして作っています。とりあえず動いていますが、これが正しい制御方法かどうかは分かりません。

なお MMC と互換性のある SPI モードを使用しています。

個人の研究用で使う分にはライセンスを取得しなくてもとやかく言われなと思います。が、問題が起きないことを保証できません。心配な人は SD アソシエーションからライセンスを取得してください。

参考サイト (2010 年 12 月時点):

SD アソシエーション: <http://www.sdcard.org/jp/home>

SD/MMC 関連記事: http://bluefish.orz.hm/sdoc/psoc_mmc.html#MMCモード

SD/MMC 関連記事: <http://naruken.cweb.tk/labo/mmc/index.html>

HMI デモソフト: <http://www2.renesas.com/micro/ja/designsupports/board/index.html>

ADPCM 圧縮 / 伸長ドライバに関する注意

ルネサス エレクトロニクスの Web で公開されている HMI デモソフトを逆アセンブル解析して、アルゴリズムを読み取って組み直したものです。本ソース・コード自体はオリジナルですが、アルゴリズムについてはルネサス エレクトロニクスがなんらかの権利を保有している可能性があります。

ルネサス エレクトロニクスでは 78K0R 用 ADPCM ライブラリ (名称 ADPCM-SP) を無償提供しているようなので、ルネサス製マイコンで使う限りにおいてはとやかく言われなんでしょうが、問題が起きないことを保証できません。心配な人は正規の無償版を取得してください。ただし正規版の API は本ドライバとは異なります。また正規版は本ドライバより処理がかなり遅い可能性があります (FAQ や API からの推測)。正規版を本キットに組み込むと速度不足で動かない可能性があります。

参考サイト (2010 年 12 月時点):

ADPCM-SP: http://www2.renesas.com/micro/ja/development/asia/softwarelibrary/adpcm_sp/all.html

HMI デモソフト: <http://www2.renesas.com/micro/ja/designsupports/board/index.html>

driver_SD フォルダ内のファイルの開示・頒布の制限

ADPCM.asm, SD.c の 2 つのファイルは開示や頒布を禁止します。

driver_SD 内のその他のファイルやオブジェクトの扱いについては、78K0R マイコン用ドライバ・キット（ベクター掲載の KLCD_78K0R）の driver フォルダ内ファイルと同じ扱いとします。KLCD_78K0R.pdf 記載の条件に従ってください。

本キットは無保証です。本キットの使用によるいかなる損害も補償しません。

開発環境

下記の統合環境 PM+および CC78K0R コンパイラ等。

・バージョンは、PM+：6.0 以上、CC78K0R：2.0 以上、RA78K0R：1.2 以上に対応。

・動作確認は、PM+：6.31、CC78K0R：2.12、RA78K0R：1.33、ID78K0RQB：3.60

CubeSuite 統合環境でも動くかもしれませんが確認はしておりません。

(8) 主な更新情報

R1.40 2010/12/24 新規公開版（KLCD_78K0R の対応リビジョンに一致させています）

(9) サポート

問い合わせ先：l00-softsupportlltq@memoad.jp （注：@@を@に変えて下さい）

できるだけタイトル先頭に【サポート依頼】を付けて下さい。

通常は3日以内に返答しますが、見落としや旅行・入院などで返答できない場合があります。

FAQ や追加情報がある場合は、下記の筆者サイトに掲載します。

<http://www2u.biglobe.ne.jp/~tequila/>

ビルド・エラー、動作不具合の問い合わせ

問い合わせ時にプロジェクト・ファイル（*.prj）を添付してください。

エラーなどのメッセージが出た場合は、コピー&ペーストして添付してください。

2010年12月24日 てきーらサンドム