12ステップで作る

組込みOS自作入門

KOZOS用ユティリティ

kz\_h8write

# kz\_h8write

kz\_h8writeは、12ステップで作る組込みOS自作入門(KOZOS)で使用しているH8ボード用に作られたH8フラッシュ書き込みプログラムです。著名な既存書き込みプログラムにおいて、書き込みに失敗する現象が多数報告されています。この書き込みプログラムの不具合を調査した上で新たに作られたのがkz\_h8writeです。

# kz\_h8writeの特徴

* 著名な既存ツールの書き込み失敗問題を解消したKOZOS専用H8フラッシュ書き込みプログラム。
* Linux、Windows、Mac OSに対応。
* 書き込みステップが見える動作表示出力機能付き。
* MITライセンスを採用し、商用、非商用を問わず自由に再利用可能。

# 著名な既存書き込みプログラムの不具合

著名な既存書き込みプログラムの不具合の一つは、ターゲットデバイスにビットレートを自動検出させる過程にあります。ホストから送信したコードに対して、ビットレートの検出が完了した時点でターゲットからある返答が返ってきます。本来であれば、この返答を検出した時点でビットレートを自動検出させるためのコードの送出をやめなければなりません。



著名な既存書き込みプログラムの場合、自動検出用のコードを送りすぎているために、ターゲットデバイスが期待する次の動作と異なる入力を与えてしまっているのです。上記動作だけを修正する事も考えましたが、KOZOSはフルスクラッチです。だったら書き込みプログラムも自作してしまえ！という事でフルスクラッチで実装しました。

# プロジェクトページ

プロジェクトはsourceforgeで管理しています。

<http://sourceforge.jp/projects/kz-h8write/>

12ステップで作る

組込みOS自作入門

KOZOS用ユティリティ

kz\_xmodem

# kz\_xmodem

12ステップで作る組込みOS自作入門(KOZOS)の第8ステップ以降では、ROMにブートローダ(kzload)を書き込み、ブートローダ経由でOSをRAMに流し込んで動作させるようになっています。転送プロトコルにはXMODEMを採用していますが、複数のプラットフォーム上に存在する汎用ツールの多くが、何ステップかの操作をした後でようやく転送が開始されるような仕組みになっています。また、環境やタイミングに依存して、汎用ツールとKOZOSブートローダの組み合わせでうまく転送できない事もあるようです。

kz\_xmodemは、上記の「複数の操作を段階的に行なう事の煩雑さ」や「転送に失敗する事の面倒さ」などを取り除く事を目的に、設計実装されたKOZOS専用XMODEMプログラムです。

# kz\_xmodemの特徴

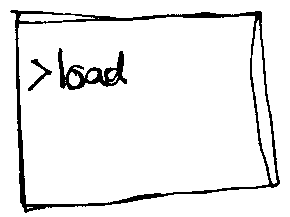
* KOZOSブートローダの動作に合わせて設計したKOZOS専用XMODEMプログラム。
* Linux、Windows、Mac OSに対応。
* KOZOSブートローダに対して内部でload状態に自動遷移。
* ターゲットをリセットしてkz\_xmodemを実行するだけで転送完了。
* コマンドプログラムなのでMakefileからの自動実行なども可能。
* MITライセンスを採用し、商用、非商用を問わず自由に再利用可能。

# 汎用ツールとkz\_xmodemの比較

## 汎用ツールの場合

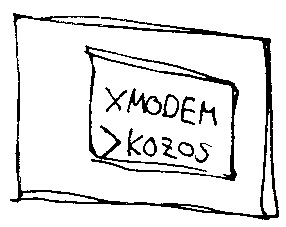
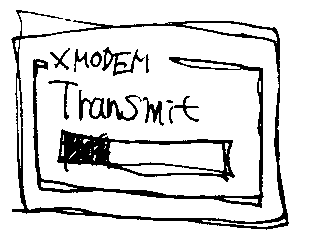
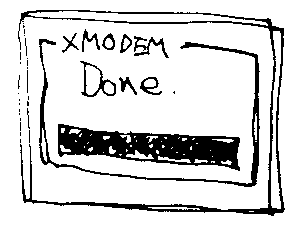
汎用ツールの場合、操作は大まかにわけて3段階必要です。

まず、KOZOSブートローダをload状態にします。

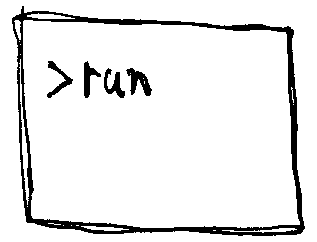


次にXMODEM転送の為にファイルを指定して転送を開始します。

ツールによっては、ファイル名の入力などに手間取ると期待したような動作にならない事があるようです。うーん。ここは気合いを入れて操作したいところではありません。

転送が正常に終了したらKOZOSブートローダのプロンプトに戻ってrunコマンドを実行します。

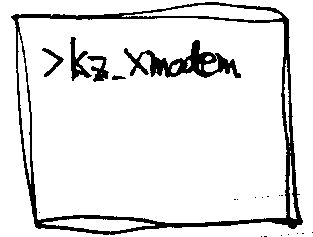


上記のように汎用ツールの場合、ツールの中の画面をいちいち行ったり来たりしなければなりません。OSの動作確認をしたくてうずうずしている時に、色々な操作が伴いとても面倒です。

## kz\_xmodemの場合

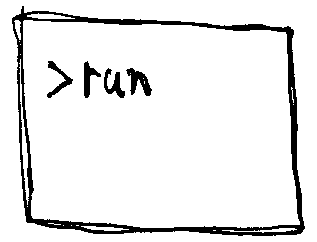
kz\_xmodemを使うと、シンプルな2つのステップでOSの動作確認まで進めます。

kz\_xmodemに、書き込み対象ファイル名と使用するシリアルポート名を与えて実行します。



書き込みが完了したらKOZOSブートローダにシリアルコンソールを使って接続します。

後はrunコマンドを実行するだけ。

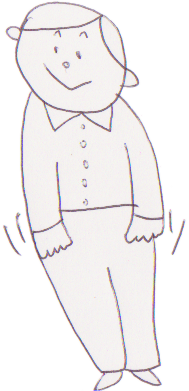
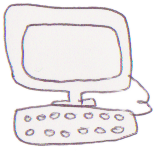


kz\_xmodemはコマンドツールですから、Makefileでビルド後に呼ぶようにすれば自動ダウンロードが可能です。要するに、ビルドした後でOSの動作確認に至るまでがmake一発で一気にできてしまうのです。従来のようにわざわざ別のツールを立ち上げてからアレコレ操作する必要はありません。

# プロジェクトページ

プロジェクトはsourceforgeで管理しています。

<http://sourceforge.jp/projects/kz-xmodem/>



CuBeatSystems