

CP/M の改良ではありません。「AVR と Z80 で CP / M」の改良の話です。

AKI-80 が販売終了？

[http://star.gmobbb.jp/koji/data/AKI\\_01.jpg](http://star.gmobbb.jp/koji/data/AKI_01.jpg)

ふと秋月さんのオンライン通販ページを見てみたら、AKI-80 が無い(2012年11月現在。スーパーAKIは販売中)。CPUはまだ在庫があるようですが、さすがにもう再販しないでしょう。TMPZ84C015は、周辺が素直に通り(SIO,PIO,CTC)内蔵されているところがよかったです。数年前に1個買ってあったので、ちょっと動かしてみたくなりました。ROMは面倒なので、AVR と Z80 で CP / M と同じ方法で動かすことにしました。

AVR と Z80 で CP / M ではとにかく動かすことだけ考えた端折った回路なので高クロックには対応できず、試したところ6MHzまででした(ファームを少し変更するとどうにか8MHzでも動く)。AKI-80の最高クロックは12MHzですが、とりあえず10MHzで動かすことを目標に改良を試みました。

#### 高クロックに対応できなかった原因の考察と改善策

私は測定器はオシロスコープしか持っていないので正確には確認できておらず推測ですが、Z80がAVRの仮想I/Oからの入力(AVRのデータポートが出力状態)をした直後、AVRはデータポートを入力に戻すわけですが、それが間に合わず、次の命令フェッチでAVRのデータポートの値をOPコードとして読み込んで誤動作しているようです。

ならばM1を取り込んで、仮想I/Oアクセスの直後にもう一発WAITが入るようにしたのがこの回路(A4を仮想I/Oと実装I/Oの切り分けにしている例)。これなら、ポートを入力に戻した後にWAITを開放するので、いくらクロックが上がっても問題ないはず。でもだいぶごちゃごちゃしてきたのでここはCPLDに置き換えることにしました。ついでにRD、WRのトライステートや、I/OのCSのデコードなど一切合財入れてしまいました。

#### 回路図(CPLD版)

AVR と Z80 で CP / M ではSDカードを入れないで起動すると、AVR内のEEPROM(2kB)の内容をRAMにロードしてZ80をリセットする仕組みになっていましたが、AKI-80が32kBのROMを搭載する設計になっているので、32kBのシリアルEEPROMを外付けして同等にしました。なので起動優先順位は、SDカード、外付けEEPROM、AVR内EEPROMです。

#### 組み立て

[http://star.gmobbb.jp/koji/data/AKI\\_02.jpg](http://star.gmobbb.jp/koji/data/AKI_02.jpg) 拡大 [http://star.gmobbb.jp/koji/data/AKI\\_03.jpg](http://star.gmobbb.jp/koji/data/AKI_03.jpg) 拡大

[http://star.gmobbb.jp/koji/data/AKI\\_04.jpg](http://star.gmobbb.jp/koji/data/AKI_04.jpg) 拡大

[http://star.gmobbb.jp/koji/data/AKI\\_05.jpg](http://star.gmobbb.jp/koji/data/AKI_05.jpg) 拡大 [http://star.gmobbb.jp/koji/data/AKI\\_06.jpg](http://star.gmobbb.jp/koji/data/AKI_06.jpg) 拡大

[http://star.gmobbb.jp/koji/data/AKI\\_08.jpg](http://star.gmobbb.jp/koji/data/AKI_08.jpg) 拡大

AKI-80と同じサイズに基板をカットし、最近流行？のスタッキング基板になるようにコネクタを配置しました。秋月さんはCPLD XC9536XL-10VQG44C が長らく在庫切れ。仕方なくPLCCタイプを千石さんで購入しました。ソケットも買っておいいたのですが、やたらスペースをとるので使うのをやめ直配線にしました。

AKI-80 本体にも少しだけ改造が必要です。ROM と RAM を使わないので、RAM を無効にするためパターンを一箇所だけカットします。やり方は、[古典電脳物語 \(Office Tetsu さん\)](#) のページに詳しく書かれています。

AKI-80 の改造方法 (戻し方も書いてあるので安心)

AKI-80 のバスコネクタには 2 つほど空きピンがあるので、ひとつを AVR からの CPU クロック出力に、もうひとつを CPLD から外付け I/O 一個用の CS 出力に割り当てています。なので上に載せる CPU 基板は AKI-80 でなくても Z80 互換な CPU なら何でも載せられ、しかも I/O デバイス 1 個を載せるだけならプルアップ抵抗とパスコンだけで済むようになっています。[未検証]

## ソフトウェア

シリアル EEPROM を追加した分の変更以外はほとんど AVR と Z80 で CP / M と同じです。シリアル EEPROM のアクセスモジュールは、[がた老さんの I2C ミニ LCD ドライバモジュール](#) を基に改造して作りました。シリアル EEPROM への書き込み手段を 2 つ提供しています。

- ・ PC からダウンロード (sepromw.exe)
  - ・ AVR の PB7 を L レベルにして起動した場合は書き込みモードとなり、PC の書き込みツールの制御下で HEX ファイルをダウンロードできます。PC ツール (Win32 コマンドライン) は [ChaN さんの AVR の書き込みツール](#) を基に改造して作りました。sepromw.ini ファイルに予め COM ポート No. を設定しておくことができます。
- ・ Z80 上の RAM からロード (EROMW.COM)
  - ・ 書込指示専用の I/O ポートを新設しています。RAM の 2000 番地から 1kB 単位で 32 kB まで任意のサイズの内容を書込むことができます。例えば、CP/M 上で 0 番地スタートのプログラムを作成して HEX ファイルを生成し、DDT で 2000 番地からにずらして RAM に配置します。この状態で書き込み用の I/O ポートを叩くと、後は AVR が全部やってくれます。CP/M 上での書き込みソフト (EROMW.COM) を作りました。これにより、端末こそ使いますがソフト作成から ROM 焼きまでセルフでできます。

AKI-80 用ソフトウェアは変更なしで動きました。実行画面

## CPLD の書込みについて

今回初めて CPLD を使いました。[西田ラヂオさんのページ](#) に、FT232 で書き込める USB プログラマーが UP されており、これを使わせていただきました。[AE-UM232R](#) 等を持っている人はライター不要です。[USB CPLD Programmer](#)

[http://star.gmob.jp/koji/data/AKI\\_10.jpg](http://star.gmob.jp/koji/data/AKI_10.jpg) 拡大

FT232RQ を使った AE-UM232R にピン互換な USB 変換器を自作して CPLD に書き込む

## 仕様のまとめ

[http://star.gmob.jp/koji/data/AKI\\_DIAG.png](http://star.gmob.jp/koji/data/AKI_DIAG.png)

ソースを見てもらえば分かるのですがまとめておきます。

今回は I/O 内蔵の Z80 互換 CPU に対応するため I/O アドレスを大幅に変更しています。(CP/M のディスクイメージは作り直しになります) [CP / M ディスクイメージ作成手順メモ](#)

起動 (シーク順序は上から)

デバイス	動作
------	----

SD カード	カードの先頭 128 バイトを RAM の 2000 番地から配置し、0 番地に 2000 番地へのジャンプコードを設定して Z80 をリセット
外付 EEPROM	EEPROM の内容を RAM の 0000 番地から配置し、Z80 をリセット
AVR 内 EEPROM	EEPROM の内容を RAM の 0000 番地から配置し、Z80 をリセット

AVR 仮想 I/O ポート (アドレスはヘッダーファイルで変更可能。I/O は AVR から見て)

名称	アドレス	I/O	設定値の意味、動作
CON_STS	80	O	シリアル入力ステータス。入力あり：FF、なし：0
CON_IN	81	O	シリアル入力
CON_OUT	82	I	シリアル出力
TRACK_SEL_L	A0	I	トラック No. LOW セット
TRACK_SEL_H	A1	I	トラック No. HIGH セット
SECTOR_SEL	A2	I	セクタ No. セット
ADR_L	A4	I	SD カードのデータ授受バッファ (128 バイト) の先頭アドレス LOW セット
ADR_H	A5	I	SD カードのデータ授受バッファ (128 バイト) の先頭アドレス HIGH セット
EXEC_DMA	A6	I	SD カードの R/W 実行。 1: リード、2: ライト
DMA_RS	A7	O	DMA の実行結果。0 :OK、1:NG
ROM_SAVE	A8	I	書込みサイズ (kB 単位) を指定し、RAM の 2000 番地以降を外付け EEPROM に書込実行する。

I/O マップ (CPLD の書き換えで変更可能)

[http://star.gmob.jp/koji/data/AKI\\_IOMap.png](http://star.gmob.jp/koji/data/AKI_IOMap.png)

## あとがき

CPLD を使ったことでだいぶ部品が減りました。こちらをベースに TK-80 のレプリカを作ったらもっとスマートなものができそうですね。誰か作ってみませんか。

[2014.05.31] さらに部品を減らす検討をしています。 [部品を減らす工夫](#)

[http://star.gmob.jp/koji/data/AKI\\_07.jpg](http://star.gmob.jp/koji/data/AKI_07.jpg) 拡大 [http://star.gmob.jp/koji/data/AKI\\_09.jpg](http://star.gmob.jp/koji/data/AKI_09.jpg) 拡大

## 関連リンク

### CPLD

[http://www.piclist.com/images/www/hobby\\_elec/cpld.htm](http://www.piclist.com/images/www/hobby_elec/cpld.htm)

<http://homepage3.nifty.com/ARTWEB/cpld1.htm>

### VHDL

<http://www.picfun.com/vhdl00.html>

[http://www.arch.cs.kumamoto-u.ac.jp/project/kite/3days/explain/VHDL/vhdl\\_tutorial.html](http://www.arch.cs.kumamoto-u.ac.jp/project/kite/3days/explain/VHDL/vhdl_tutorial.html)

マイコン PCB [PCB-ATMEGA128](ATmega64/128 対応)

<http://www.aitendo.com/product/2837>

### 当サイト内

[AVR と Z80 で CP / M](#)

[AVR と Z80 で TK-80](#)

[AVR ビデオ端末の製作](#)

[2716 用 ROM ライターの製作](#)

[mbed\(ST Nucleo F401RE\) 版 CP/M エミュレータ](#)

[部品を減らす工夫](#)

[その他の AVR 関連の製作](#)

[AVR パラレルライターの製作](#)

[AVR 学習リモコンの製作](#)

[Mega64 で LED GAME](#)

- 
- ・TMPZ84C015 販売してるよ - 名無しさん (2014 年 07 月 05 日 15 時 48 分 06 秒)
  - ・はい、秋月電子さんで保守用在庫限りで販売されています。AKI-80 が販売終了になったのは CPU の在庫数が製造ロットを割ったためと推測しています。この回路は Z80 互換な CPU に適用できるので、TMPZ84C015 でなくても構いません。 - 管理人 (2014 年 07 月 05 日 23 時 45 分 00 秒)
  - ・2017.1. 末 AKI80 基板在庫復活したようです 在庫クラス AAA - Z80 (2017 年 02 月 04 日 22 時 35 分 42 秒)
  - ・基板単体だけと思ったら部品付きも復活してますね。 - 管理人 (2017 年 02 月 05 日 03 時 05 分 35 秒)