

サンプルプログラムの使い方 Ver 1.00 2006.02.21

対象CPUボード 【BCH8SX1657】

動作環境 【ルネサステクノロジ製 HEW4がインストールされているパソコン】

サンプルプログラム内容

以下に示す4つのサンプルプログラムが1つになったソースファイル「test1657s.c」、実行ファイル「test1657s.mot」がダウンロードできます。フォースライタで「BCH8SX1657」に書き込み、フォースビューで動作確認することができます。ソースファイルも添付されています。

```
test_da()      //D/A出力テスト
test_ad()      //A/D入力テスト
test_io()      //I/O入出力テスト
test_sio()     //SIO入出力テスト
```

開発環境

ファイルは株式会社ルネサステクノロジ社 HEW4のプロジェクト形式で構成されています。中のソースファイルは(株)秋月電子通商のCbar(GCC on Windows) 有限会社イエローソフト社のYCH8、YLINKでもコンパイル、実行ファイル作成できます。それぞれの環境でファイルに変更等を加える必要はありません。*1

この3つの環境で作成される実行ファイルはすべて「フォースライタ」で「BCH8SX1657」に書き込みます。

* 1 (株)秋月電子通商のCbarでは警告が出ますが、動作は問題ありません。気になる場合、`void main(void)`を`main()`に書き換えると消えます。

【使い方概要】

1. 初めにダウンロードしてください。「フォースライタ」のダウンロードで¥BRE¥H8SX1657ディレクトリの中に添付されてきます。



2. H8SX/1657用のフォースライタで「test1657s.mot」を「BCH8S

X 1 6 5 7」に書き込みます。

3. フォースビューで0, 1, 2, 3キーをクリックするとそれぞれ

```
0 = test_da() // D / A 出力テスト
1 = test_ad() // A / D 入力テスト
2 = test_io() // I / O 入出力テスト
3 = test_sio() // S I O 入出力テスト
```

の動作が確認できます。リセットは「CPU RESET」をクリックしてください。また次の動作を受け付ける待ちに入ります。

【使い方詳細】

【HEW4でプログラムを修正、コンパイルする場合】

1. ダウンロードするとH8SX1657の中にtest1657sディレクトリがあります。

名前	サイズ	種類	更新日時
Document		ファイル フォルダ	2006/02/20 15:39
test1657s		ファイル フォルダ	2006/02/20 15:26
frw1657	313 KB	アプリケーション	2006/02/21 8:58
frw1657.par	1 KB	PAR ファイル	2006/02/15 19:00

2. ホルダごとコピーしてHEW4がプログラムホルダ用に作成したWorkspaceディレクトリの中へペーストします。HEWによっては別ディレクトリになる場合もあります。

Workspace		ファイル フォルダ	2006/02/20 15:25
-----------	--	-----------	------------------

3. HEW4を開いて「最近使用したプロジェクトワークスペースを開く」で

C:\¥Workspace¥test1657s¥test1657s.hws 迄設定し、「OK」をクリックします。

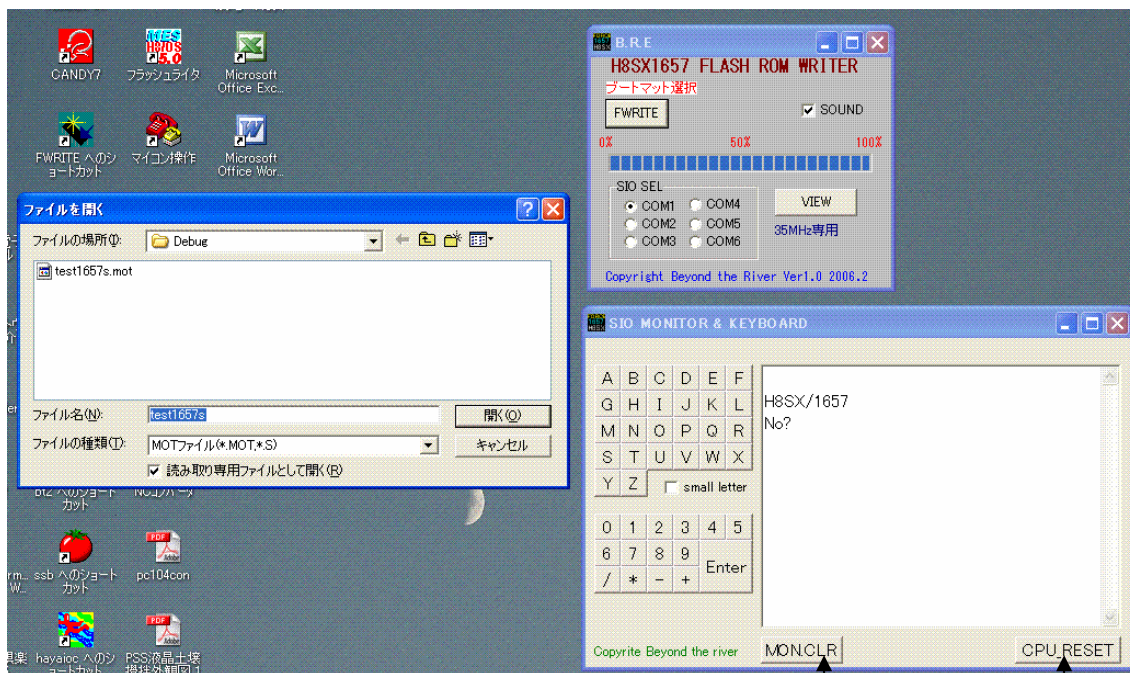
4. HEW4で修正、コンパイルできる状態になります。

【motファイルをH8SX1657に焼きこみ動作させてみる場合】

フラッシュROMに書き込むtest1657.motファイルは標準的なダウンロードの場合、以下の場所にあります。

C:\¥BRE¥H8SX1657¥test1657s¥test1657s¥Debug

【サンプルプログラムの書き込み、動作】



フォースビューモニタ画面のクリア CPUリセット

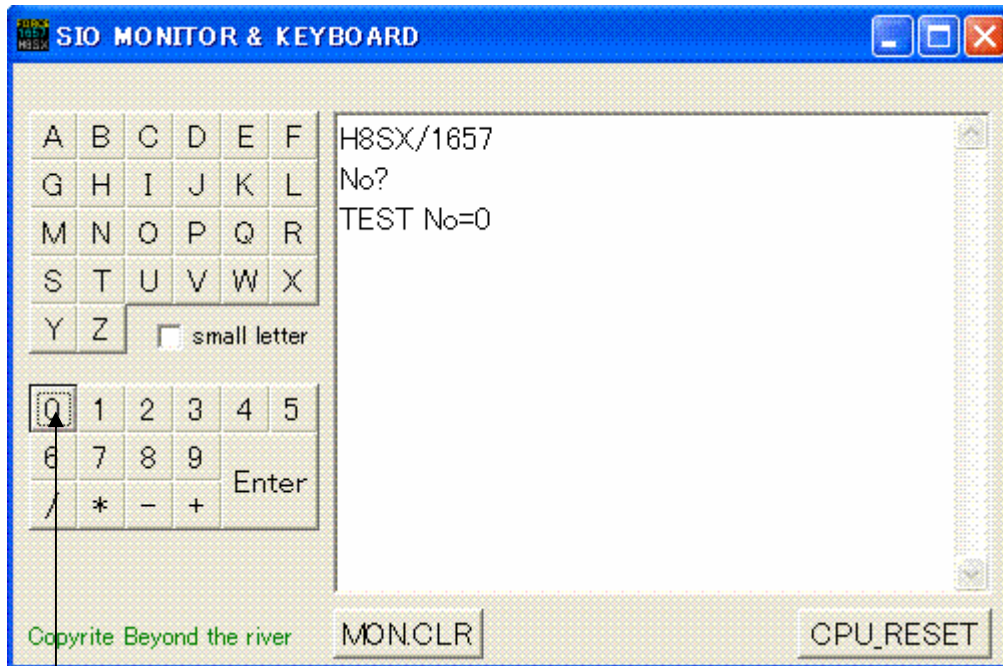
フォースライタのVIEWをクリックして、フォースビューを表示しておきます。test1657s.motファイルを書き込みます。書き込み終了とともにフォースビューの画面にはH8SX/1657 No?と表示されます。

プログラムは入力待ちです。フォースビューで0, 1, 2, 3キーをクリックするとそれぞれの動作が選択されます。

- 0 = test_da() // D / A 出力テスト
- 1 = test_ad() // A / D 入力テスト
- 2 = test_io() // I / O 入出力テスト
- 3 = test_sio() // S I O 入出力テスト

D/Aテスト

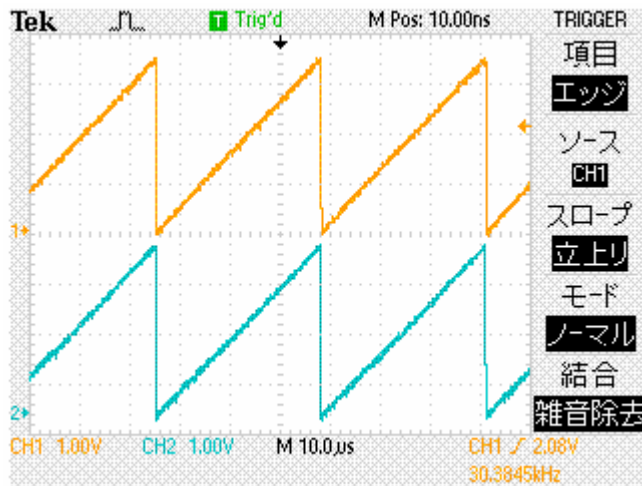
0をクリックし、D/Aコンバータのテストを行います。



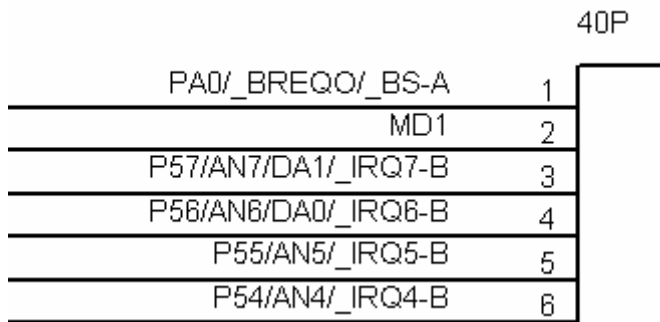
ここをマウスでクリックする。TEST No = 0と表示されます。

D/Aの波形を観測すると以下のように三角形の波形を繰り返します。

BCH8SX1657のCN4の3、4ピンを観測します。



0 ~ 2.55 (0 ~ 3.3V) までのD/A出力を30.3845kHzで繰り返しています。



プログラムは以下の通りです。

```

void test_da(void)
{
//DA Test
unsigned char dabuff;

MSTPCRA &=~0x0020;    //DA を使用する

dabuff = 0;
DACR01 = 0xff;        //DA0,1 出力

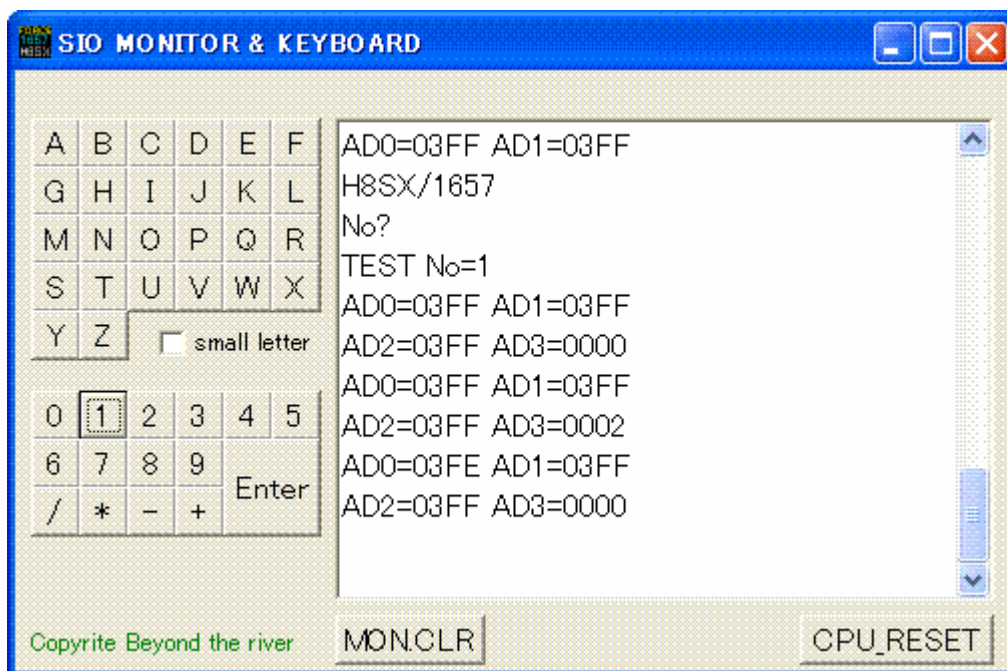
while(1)
{
DADRO = dabuff++;
DADR1 = dabuff++;
}
}

```

D/A0,1とも選択し、2つあるD/Aにそれぞれ0~255(0x00~0xFF)までの数をセットして出力電圧を変化させています。dabuffは255までは1ずつ加算され255の次は0になりますので、三角形の波形になります。

A/Dテスト

CPU_RESETをクリックし、CPUをリセットさせます。1をクリックし、A/Dテストを選択します。



A / D 0 ~ 3 の 4 つの値が表示されます。それぞれの A / D 端子 (A N 0 ~ A N 3) にポリウ
ム (1 0 K 以下) など を 接 続 し、電 圧 を 可 変 し て い く と 数 値 が 変 わ る の が 確 認 で き ま す。 0 ~
5 V で 0 から 0 x 3 F F (0 ~ 1 0 2 3) ま で 変 わ り ま す。

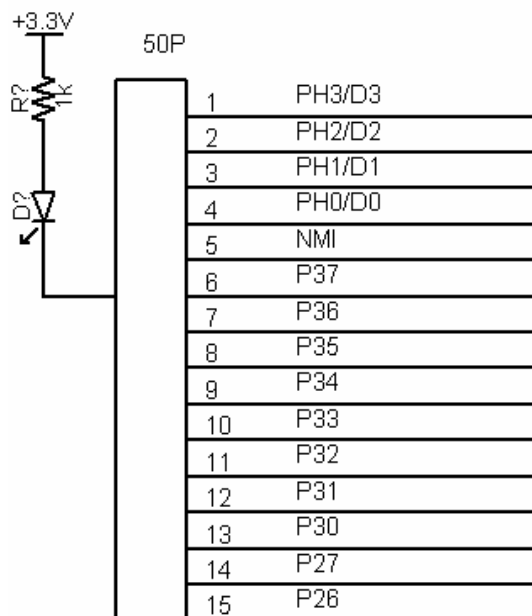
I O テ ス ト

P 2、P 3 を 出 力 に 設 定 し、0、0 x f f を 繰 り 返 し て い ま す。

```
void test_io(void)
{
    P2DDR = 0xff; //出力設定
    P3DDR = 0xff; //出力設定

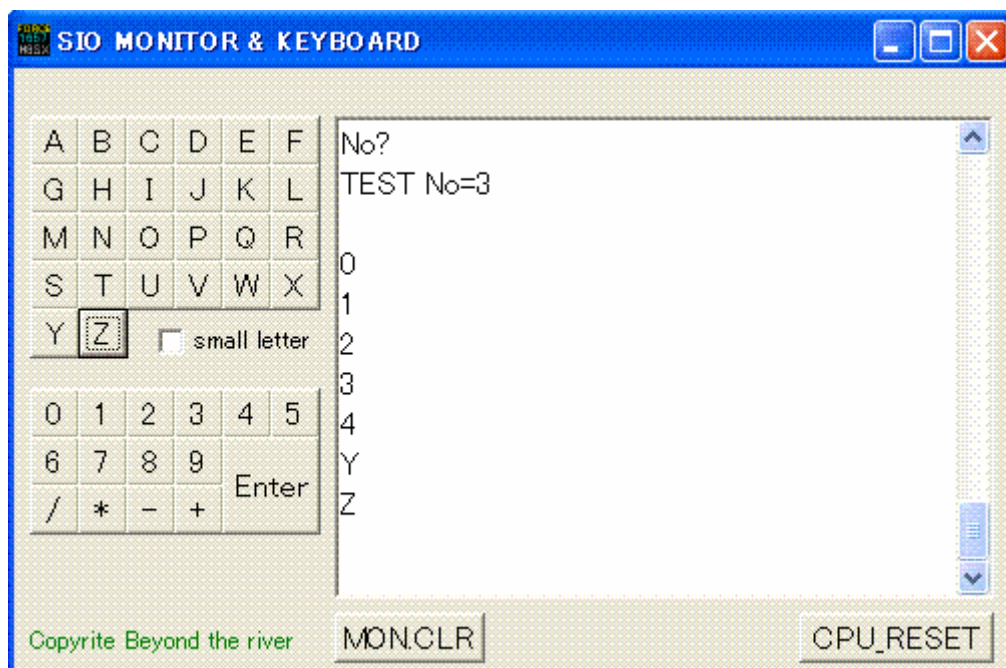
    while(1)
    {
        P2W = 0xff;
        P3W = 0xff;
        wait(10000);
        P2W = 0x00;
        P3W = 0x00;
        wait(10000);
    }
}
```

ウエイトで約10Hzの信号になっていますので、LEDを付けて点滅するのが確認できます。ウエイトの数を変えてLEDの点滅時間が変化するのを確認できます。



SIOテスト

CPU_RESETをかけ3番をクリックします。以降、押したキーボードの値が表示されればOKです。



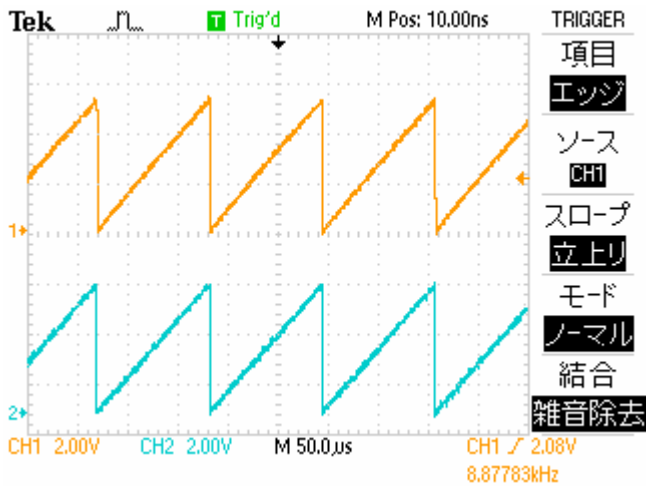
フォースビューのキーボードはクリックされた値をパソコンからマイコン側にアスキーコードで出力します。

```
void test_sio(void)
{
    while(1)
    {
        char_out4(char_in4());
    }
}
```

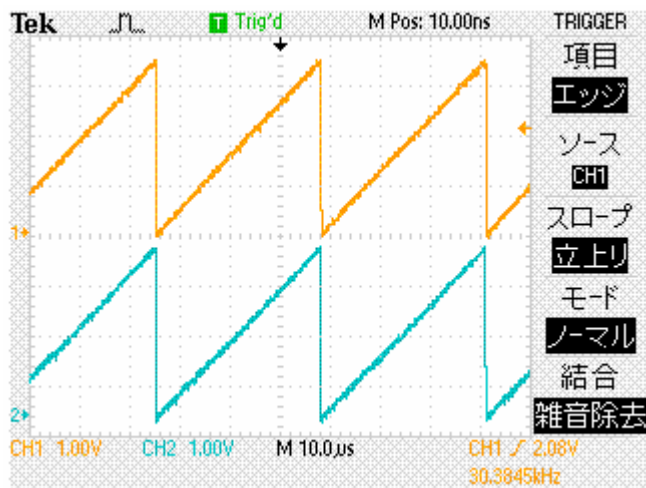
プログラムでは受信したデータをそのままパソコンに返しています。返ってきた値をモニタに表示しています。

速度比較

【BCH83052】動作波形



【BCH8SX1657】動作波形



今回、H8SX/1657の動作速度をH8/3052と比較してみました。前述のD/A出力プログラムを同じHEW4でコンパイルし、動作させ時間を計測しました。

BCH83052では8.87783kHz、BCH8SX1657では30.3845kHzと約3.4倍の速度です。十分有意義な数値と見るかどうか評価は様々でしょうが、クロック比は35MHz / 25MHz = 1.4倍ですから、内部32ビットバス幅 + 2段パイプライン + インストラクションFIFOの構成が利いているのか、かなり速くなっている印象は受けます。32ビット演算命令が追加されているので演算速度の比較も面白いと思われます。

ご注意

HEWは株式会社ルネサステクノロジ社の登録商標です。

Windowsは米国マイクロソフト社の登録商標です。

CbarはKENCH氏によるフリーソフトです。

GCC(GNU Compiler Collection)CはGNUプロジェクトによるフリーCコンパイラです。

YellowIDE、イエロースコープは有限会社イエローソフト社の登録商標です。

「フォース」機能、「フォースライタ」は弊社の発明です。知的財産権申請、登録されています。

- 1．本文章に記載された内容は弊社有限会社ビーリバーエレクトロニクスの調査結果です。
- 2．本文章に記載された情報の内容、使用結果に対して弊社はいかなる責任も負いません。
- 3．本文章に記載された情報に誤記等問題がありましたらご一報いただけますと幸いです。
- 4．本文章は許可なく転載、複製することを堅くお断りいたします。

〒350-1213 埼玉県日高市高萩 1141-1 TEL 042(985)6982 FAX 042(985)6720

Homepage : <http://beriver.co.jp>

e-mail : support@beriver.co.jp

有限会社ビーリバーエレクトロニクス