

## 今回のハードウェアは、以前作った USB-I2C変換アダプタを使用しました

今回の、ハードウェアの説明をしていませんでしたが、以前作った USB-I2C変換アダプタを使用しました。

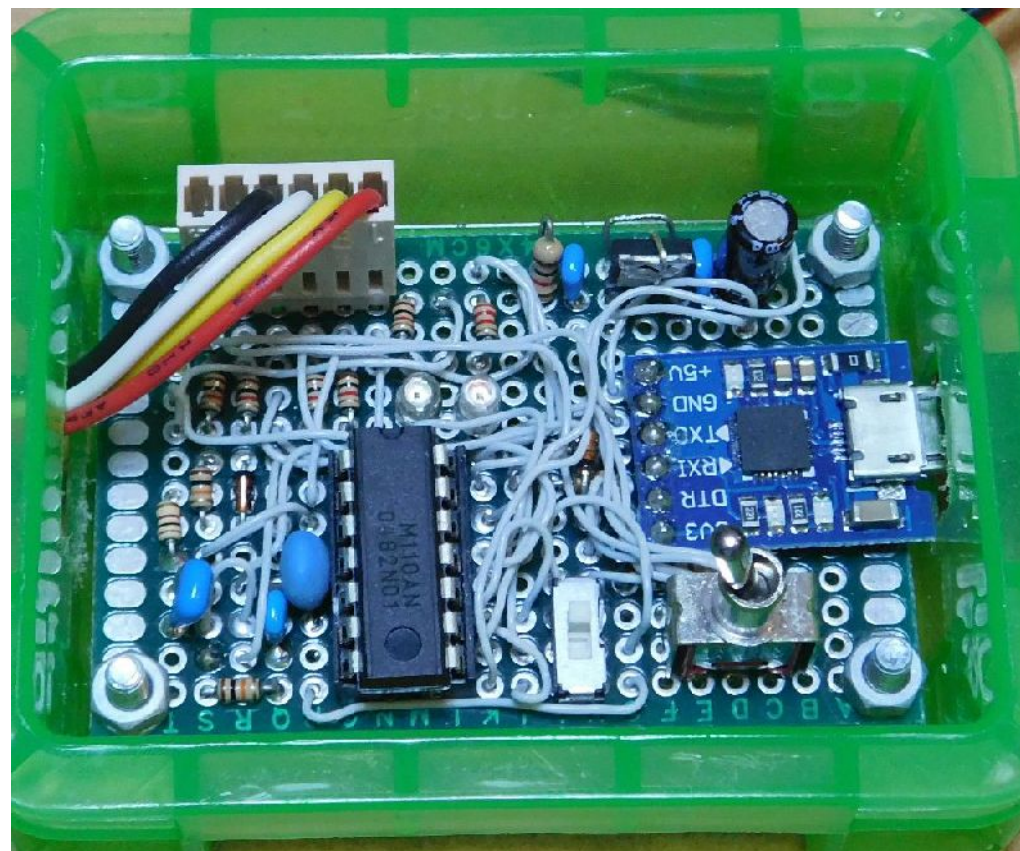
R8C/M110Aマイコンのプログラムを今回の OLED表示器の物に差し替えて実験しました。

プログラムを戻せば、元のUSB-I2C変換アダプタになります。

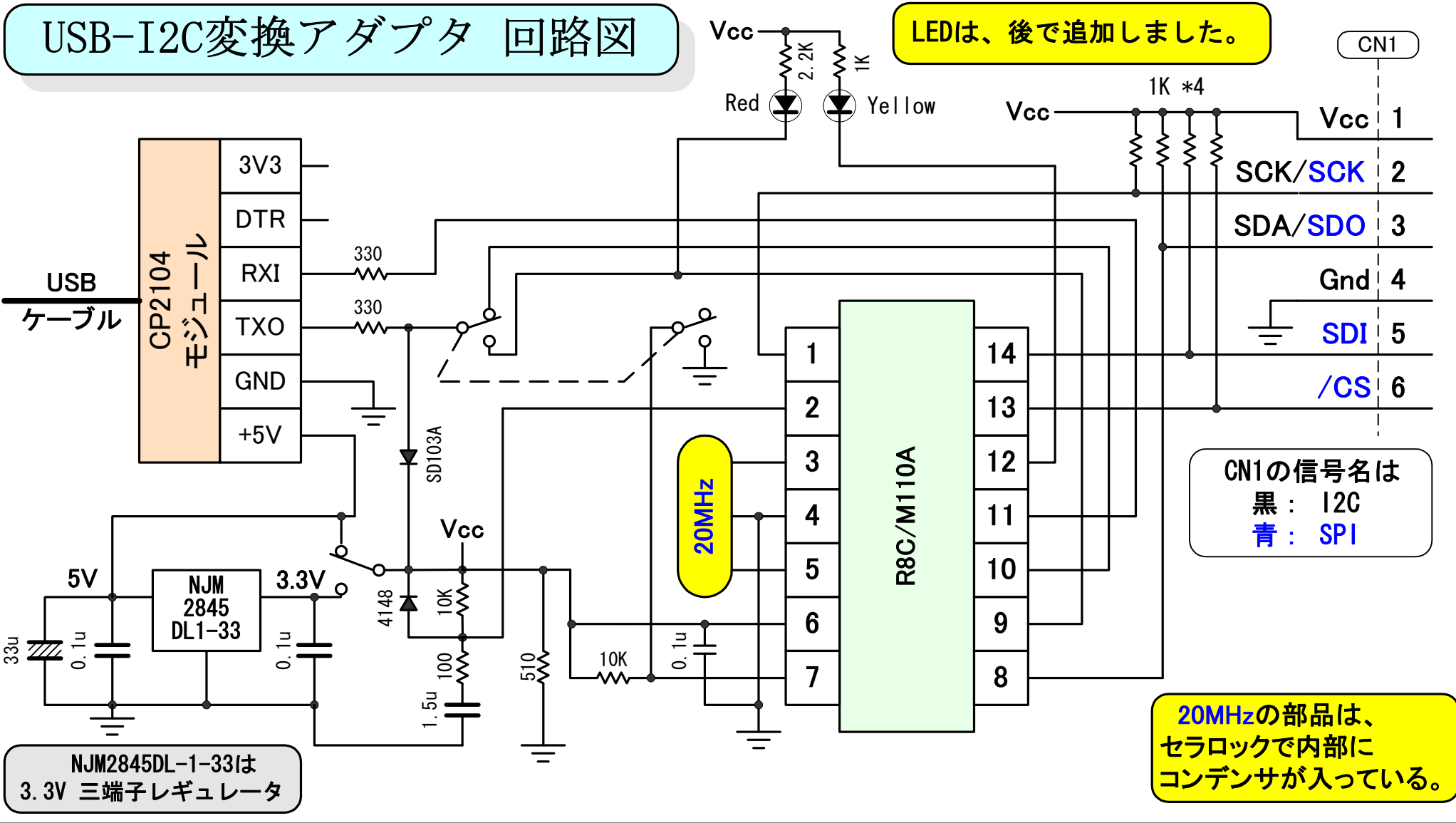
OLEDのインタフェースが I2Cだったので  
同じハードで使えたという事です。

USB-I2C変換アダプタの回路図を、次の  
ページに入れました。

14ピンの百円マイコンが 結構頑張ってます。



## USB-I2C変換アダプタ 回路図



## R8Cマイコンに送るコマンドについて

電源投入後、最初に Enterキー（ 0Dh ） を R8Cマイコンに送ります。 その後 ">"が表示され コマンド受付状態となります。 コマンドは

- ① cls[Enter] : OLED表示消去
- ② fxxyy:文字列[Enter] : 先頭が f の場合  
8x16 dotの 文字で表示します。  
xx : カラムロケート値 ( 0~15 )  
yy : 行ロケート値 ( 0 か 1 )  
文字列 : 文字列長は yy=0 の場合、  
右端に文字が当たったら、続きを xx=0、  
yy=1 に表示します。 文字が表示範囲を  
はみ出さないようにして下さい。
- ③ Fxxyy:文字列[Enter] : 先頭が F の場合  
16x32 dotの 文字で表示します。  
最大 8 文字です。 yy=0 でしか使えません。



## 百円マイコンは、割りきって使う

今回は、パソコンからコマンドを送信しUSB-シリアル変換モジュールを経由してR8Cマイコンで、送られてきた文字に合わせてOLEDディスプレイに、BitMapFontを転送しました。

実験する時は分かりやすいですが、通常パソコンに、この小さなOLEDディスプレイを接続しても意味がないので、マイコンで活用する事になります。

私は、設定時間が経過したら、音が鳴るタイマー（出来れば2チャンネル）を作りたいなと思います。

このようなタイマーぐらいであれば、R8Cマイコン1個でも十分出来ると思います。

しかし、より本格的な用途であれば、1msを切るような高速な A/Dスキャンングやその後の演算、判断処理を リアルタイムで行う必要が出てきます。

そのような場合は、高速な 32bit CPUを使う事になります。

しかし周辺の こざこざした処理まで全て一つのCPUに やらせるのは 場合によってはちょっと厳しい場合もあります。

そのような時は R8Cのようなローエンドマイコンを、サテライトCPUとして特定の機能を任せると処理が すっきり実現出来る場合があります。

今回も、R8CマイコンプログラムのHEWプロジェクトファイルをダウンロード出来るようにしておきます。