

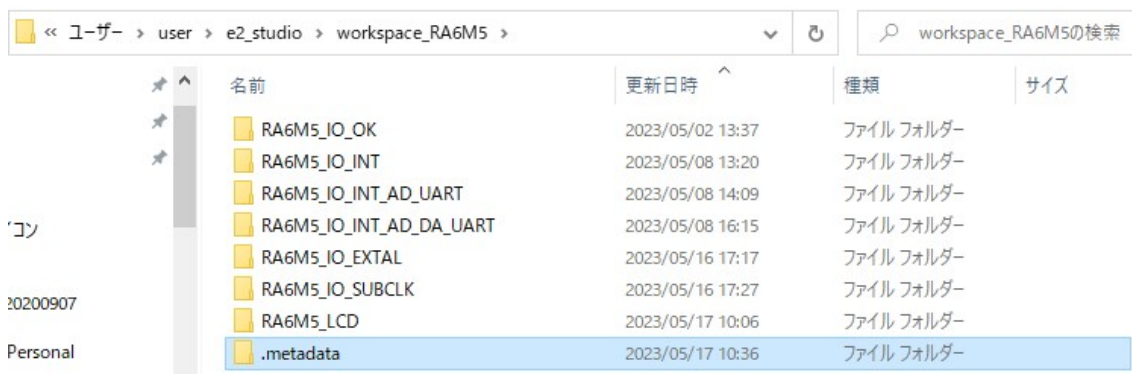
RA6M5_CPUボードサンプルプログラムの使い方 2023.05.17

■事前準備

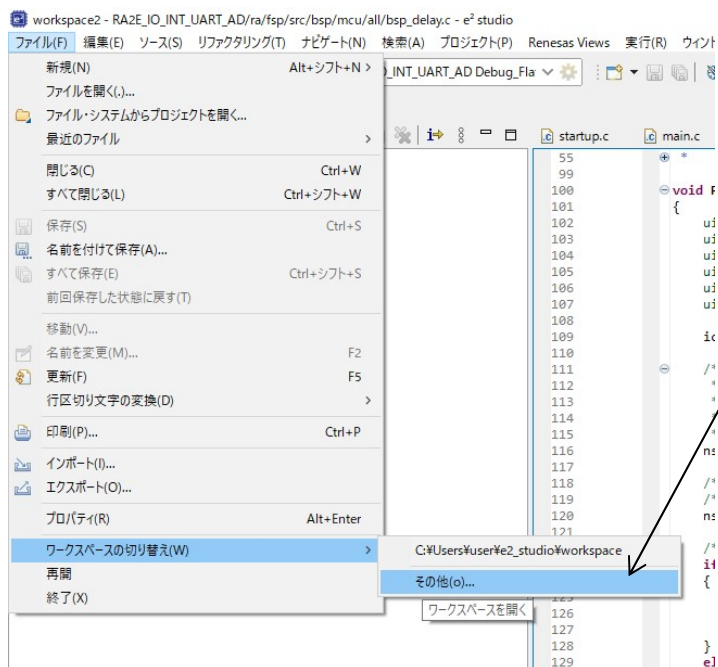
- windows 10以上が動くパソコン
- E2エミュレータ lite
- e2studioのダウンロード、インストール
- RA6M5_LCD では有機ELモジュール SO1602AWWBとケーブル
- RA6M5_IO_INT_AD_UART ではUART⇄USB変換器とケーブル等が必要です。

■動作の方法

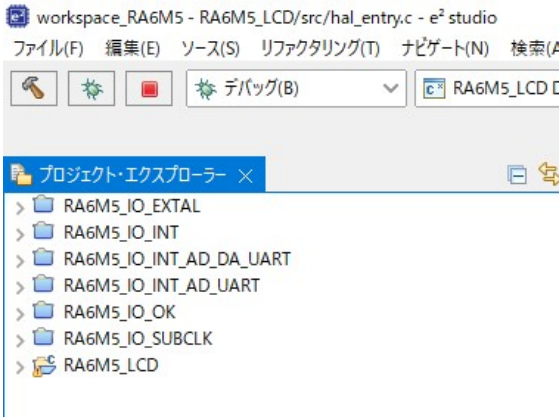
1. ダウンロードしたファイルを解凍し、workspace_RA6M5のホルダごと¥e2studioの下の場所に移します。例



2. e2studio ファイル→ワークスペースの切り替え→移したホルダのworkspace_RA6M5を指定してください。



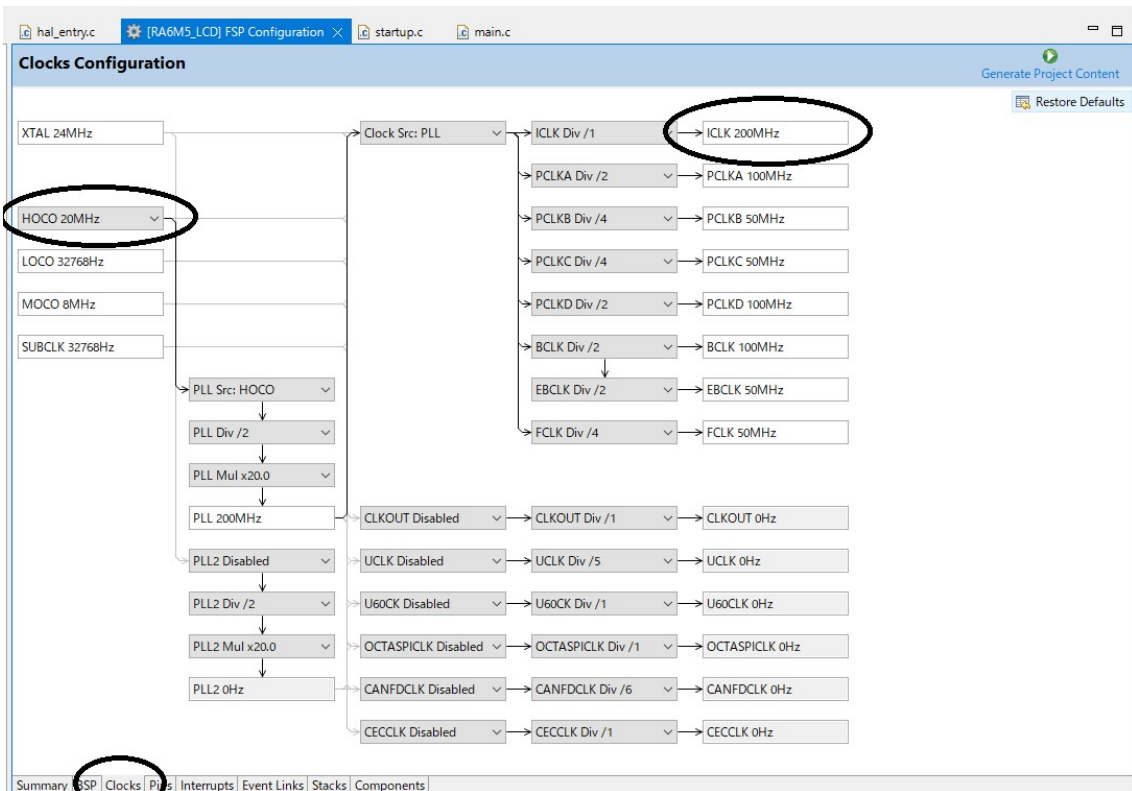
3. 正常に読み込めればプロジェクト・エクスプローラにいくつかのRA6M5用のプロジェクトが表示されます。



この例では上から

RA6M5_IO_EXTAL : I/Oポートの半田付け検査用全ポートON, OFFを繰り返し、未半田、他ポートとの接触を確認できるプログラムです。外部クリスタルで動作させています。

RA6M5_IO_INT : 上記と同じですが、内部発振器を使用しています。時間を1msecの定周期割り込みで作成しています。



RA6M5_IO_INT_AD_DA_UART : IO, INTは上記と同じですが、A/DコンバータのデータをD/AコンバータとUART (USB) に出力しています。

RA6M5_IO_INT_AD_UART : 上記と同じですが、D/Aコンバータ動作がありません。

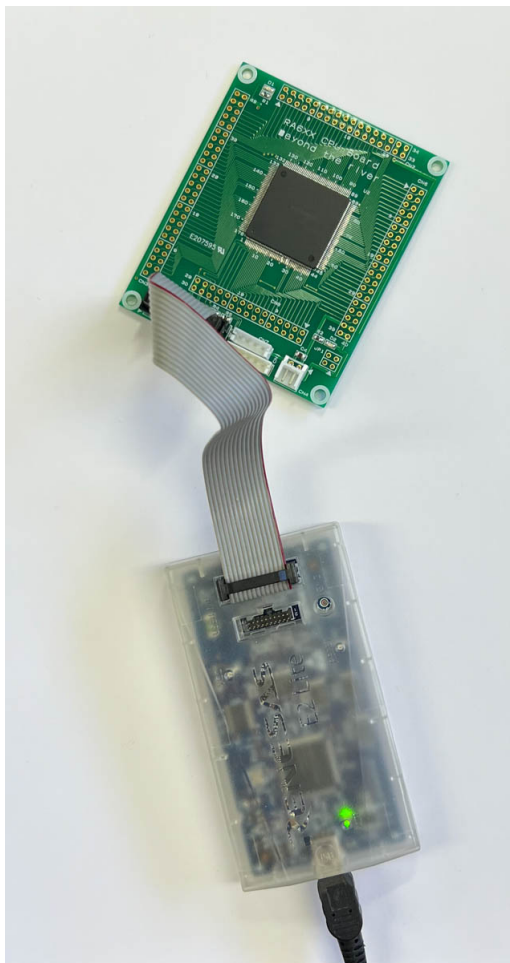
RA6M5_IO_OK : I/Oポート検査は同じですが、LED点滅周期をプログラムのループで作成しています（最もシンプルなプログラム）

RA6M5_IO_SUBCLK : I/Oポート検査は同じですが、動作クロックを外部クリスタルの32.768KHzを使用しています。外部クリスタル動作検査。

RA6M5_LCD : A/D変換した値を有機EL SO1602AWWB（秋月電子通商さん扱い）に表示させるプログラムです。

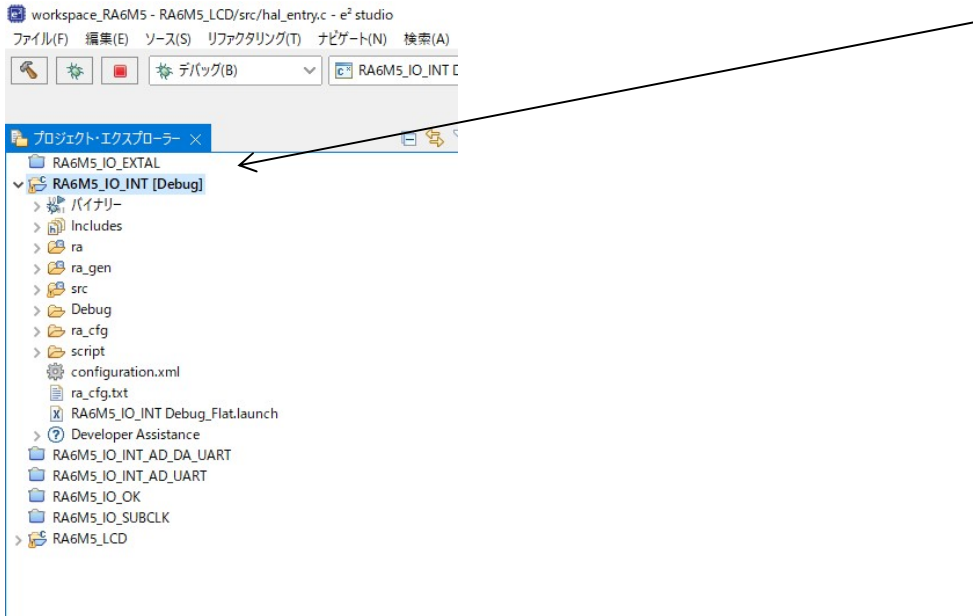
★プロジェクトを他のホルダ、例えばworkspaceにwindowsのコピペで移動させてプログラムを動かそうとしても動きません。e2studioのファイルの移動は基本的にインポート、エクスポート機能で行います。

プログラムを実行させる前にE2liteのケーブルをCPUボードのCN5に挿入します。全てのプログラムはE2liteから供給される3.3Vの電源で動くので、外部に用意する必要はありません。

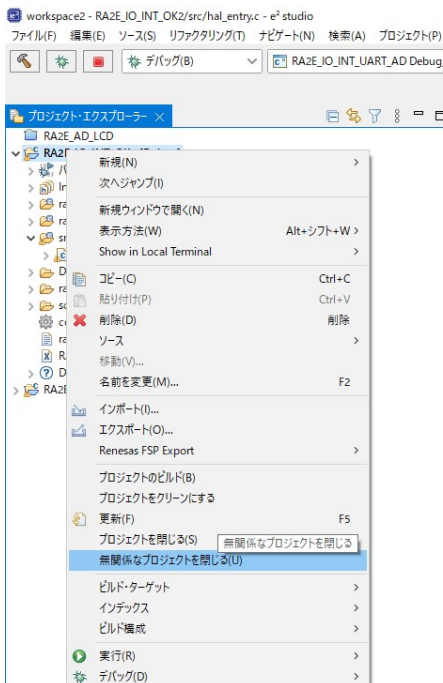


■ RA6M5__IO__INT の動作

RA6M5__IO__INTをダブルクリックします。ホルダが開いた形に変化し、色々なホルダ、ファイルが表示されます。

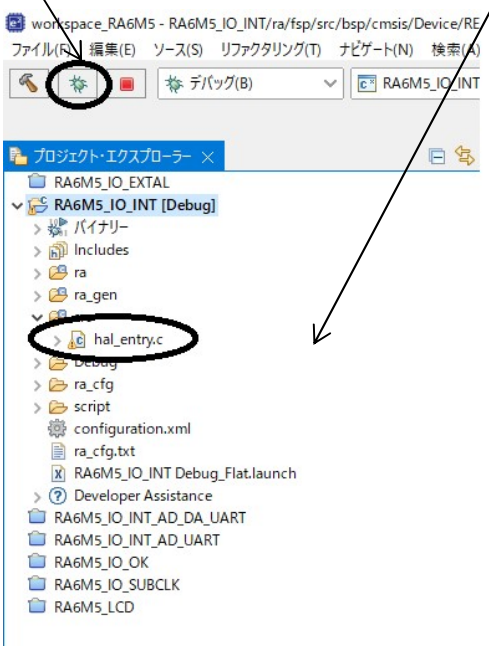


初めに、RA6M5__IO__INTを右クリック。「無関係なプロジェクトを閉じる」を実行してください。他のプロジェクトが開いていなければ選択できず、不要です。※1 詳細は後述



hal_entryの中にプログラムは書かれています。クリックするとプログラムが表示されます。

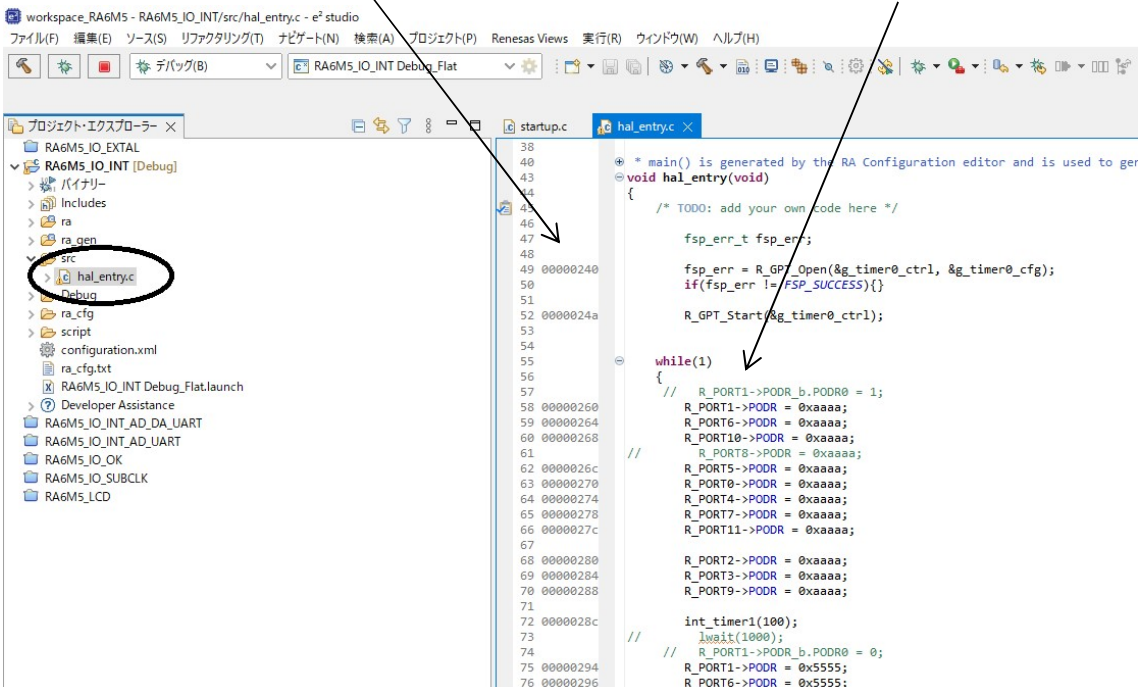
虫？マークをクリックし、デバック開始



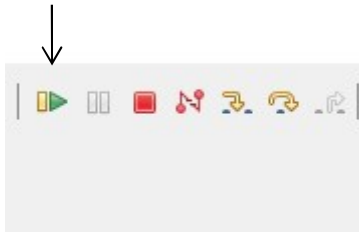
「パースpekティブの切り替え」→「切り替え」

正常にダウンロードされるとアドレスが表示されます。

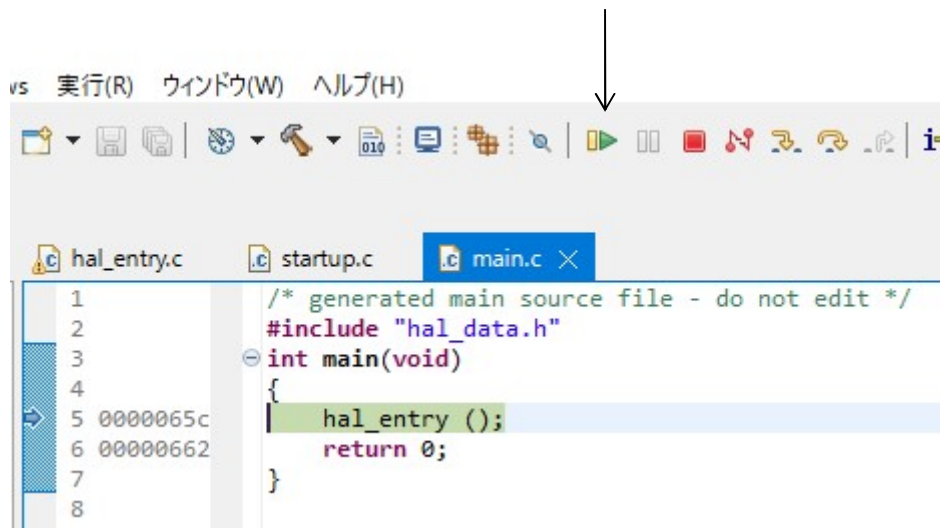
プログラムはポートのH, Lを0. 2秒周期で繰り返します。



再開をクリック。



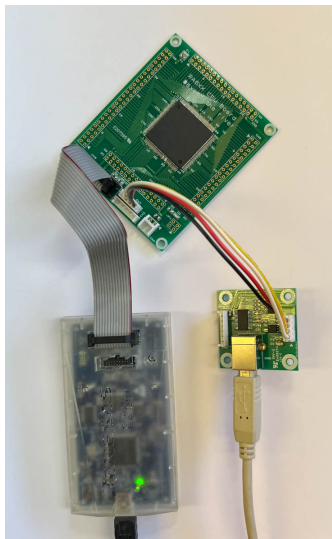
一度、hal_entryで止まりますので、再度、再開をクリック。



プログラムが実行され、LED D1が0.2秒周期で点滅します。

■ RA6M5_IO_INT_AD_DA_UARTの動作

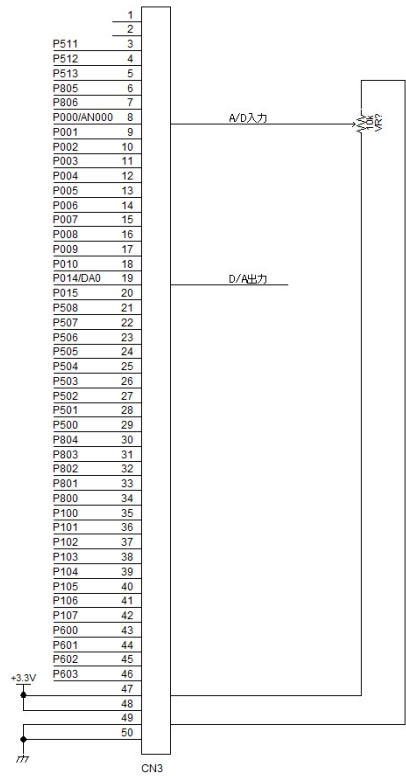
AN000/P000 CN3の8番ピンに0~3. 3Vを加え、A/D変換し、0-4095のデータをD/A、UART (USB) で出力し、PCに表示します。プログラムの動作方法は前例と同じです。



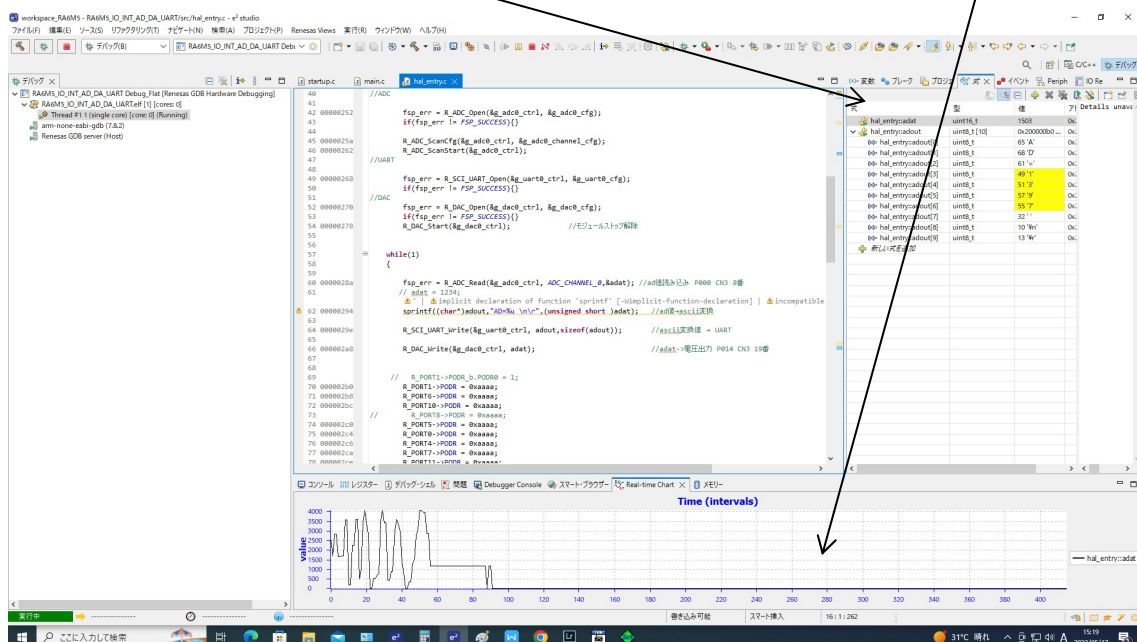
```
COM14 - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
AD=3633
AD=4094
AD=4094
AD=4095
AD=3657
AD=3213
AD=2372
AD=1428
AD=171
AD=0
AD=0
AD=0
AD=264
AD=1159
AD=3391
AD=4094
AD=4094
AD=4094
AD=4094
AD=4094
AD=4094
AD=4094
AD=4094
```

PC側はテラタームなど通信プログラムで受信します。115200bps です。

P000/AN000 にポリウムを付けると0-3.3Vの変化でA/D値が0-4095と変化するのが確認できます。無しでも何らかの数値は出ます。P014/DA0をテスターやオシロスコープで観測すると、入力に同期して変化するD/A出力を見ることが出来ます。



e2studioは変数をリアルタイムで見るだけでなく、以下のようにグラフ化して見ることも可能です。



上例はAD値を入れている `adat` 変数をグラフ化しています。

式表示では変数をリアルタイムで表示でき、前と違う数値が黄色になります。adout[]はadat 16進データをUARTから出力し表示するためにASCII文字に変換したデータです。UART→USB経由で表示される文字列と同じになります。

```
R_ADC_Read(&g_adc0_ctrl, ADC_CHANNEL_0,&adat); //ad 値読み込み P000 CN3 8 番
sprintf((char*)adout, "AD=%u %n¥r", (unsigned short )adat); //ad 値→ascii 変換
R_SCI_UART_Write(&g_uart0_ctrl, adout, sizeof(adout)); //ascii 変換値 → UART
R_DAC_Write(&g_dac0_ctrl, adout); //adat->電圧出力 P014 CN3 19 番
```

式	型	値	ア
hal_entry::adat	uint16_t	3468	0x:
hal_entry::adout	uint8_t[10]	0x200000b0...	0x:
(*)= hal_entry::adout[0]	uint8_t	65 'A'	0x:
(*)= hal_entry::adout[1]	uint8_t	68 'D'	0x:
(*)= hal_entry::adout[2]	uint8_t	61 '='	0x:
(*)= hal_entry::adout[3]	uint8_t	51 '3'	0x:
(*)= hal_entry::adout[4]	uint8_t	54 '6'	0x:
(*)= hal_entry::adout[5]	uint8_t	50 '2'	0x:
(*)= hal_entry::adout[6]	uint8_t	51 '3'	0x:
(*)= hal_entry::adout[7]	uint8_t	32 ' '	0x:
(*)= hal_entry::adout[8]	uint8_t	10 '¥n'	0x:
(*)= hal_entry::adout[9]	uint8_t	13 '¥r'	0x:

式の更新時間は変数→右クリック→Real-time Refresh→リアルタイムリフレッシュ間隔 で変更できます。チャート表示は→チャートに追加。

式	型	値	ア
hal_entry::adat	uint16_t	3495	0x:
hal_entry::adout	uint8_t[10]	0x200000b0...	0x:

- すべて選択(A) Ctrl+A
- 式のコピー(E) Ctrl+C
- 削除(O)
- すべて削除(L)
- 数値の書式 >
- Add Expression Group >
- 検索(F)... Ctrl+F
- 詳細の表示(D) >
- 監視式を追加(A)...
- 使用不可にする
- 使用可能にする
- 監視式を編集(E)...
- 監視ポイントの追加 (C/C++)...
- 型にキャスト...
- 配列として表示...
- リアルタイム・リフレッシュを無効にする
- Real-time Refresh >
- 監視式の作成(T)

リアルタイム・リフレッシュ間隔
式の値を編集

CPUボード CN5のUART信号 TXD9, RXD9 をUSB変換器に接続しています。

<https://beriver.cart.fc2.com/ca7/122/p-r-s/>

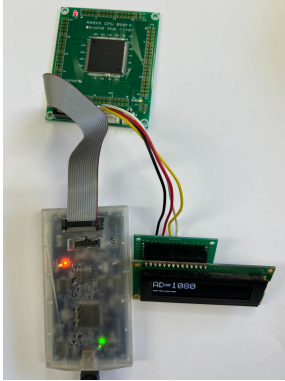
ホットプラグで絶縁されているので、CPU本体の電源OFFのたびにデバイスドライバを設定しなおす手間が不要です。

秋月電子通商さんの以下も使用できると思います。

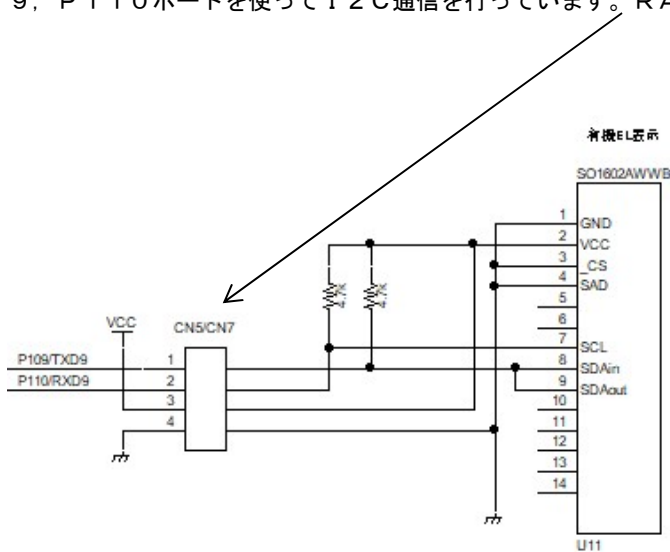
<https://akizukidenshi.com/catalog/g/gM-08461/>

■ RA6M5_LCDの動作

A/D変換した値を有機EL SO1602AWWB（秋月電子通商さん扱い）に表示させるプログラムです。有機ELは自分で光るから、後ろの光の透過を見る液晶とは段違いに視認性が良いです。プログラムの動作の方法は前例と同じです。



配線は4本の線を以下のように接続して下さい。プルアップ抵抗4.7Kが2本必要です。ここではP109、P110ポートを使ってI2C通信を行っています。RA6M5基板はCN5→CN7になります。



CN7からの圧着済みハーネスが販売されています。

<https://beriver.cart.fc2.com/ca7/117/p-r-s/>

※1 「無関係なプロジェクトを閉じる」必要性について

複数のプロジェクトがプロジェクト・エクスプローラで開いた状態でコンパイルをかけると複数のプロジェクトにまたがったコンパイルになり、意図しない出力が出ます。それを避けるためにはこの「無関係なプロジェクトを閉じる」必要があります。

有限会社ビーリバーエレクトロニクス

Mail: info@beriver.co.jp

〒350-1213 埼玉県日高市高萩1141-1