

今回の 128x64dotの OLED 2個セットの商品です。
Amazonにて購入しました。



前回の
128x32dotのプログラムで
初期化後の表示です。



その後、
128x32dotのプログラムで
画面消去した表示です。



A small LCD screen is mounted on a breadboard. The screen displays the numbers 12345678 in a pixelated, cyan-colored font. The breadboard is yellow and has several screws. A white connector with red, white, and blue wires is plugged into the breadboard. A blue tab is visible on the left side of the LCD screen. The background is a wooden surface.

そして、
128x32dotのプログラムで
16x32dotFontで“12345678”を
描画した表示です。

初期化と、画面消去の関数を改修して上半分の縦 32dotに対し、画面消去そして文字表示が出来るようになりました。






12345678
ABCDEFGH

そして、縦 64dotを、フルに
アクセス出来るようになりました。

8x16 dot Fontの文字列も表示出来ました。



ABCDEFGHIJKLMNOP
QRSTUVWXYZ
abcdefghijklmnop
qrstuvwxyz

128x32 OLED と 128x64 OLEDに 対する コマンドの出し方

今回の 128x32OLEDと 128x64OLEDはどちらも同じ制御IC(SSD1306)を使用しています。 2つの表示器は、縦のdot数が異なる以外は、同じようです。よって 縦 64dotに関わるコマンドの出し方が、部分的に異なります。

初期化処理は、縦32dotと 縦64dotで処理を、切り替える必要があります。

画面消去も、縦32dotと 縦64dotで処理を、切り替える必要があります。

幸い**文字表示処理**は、縦32dotと 縦64dotで、共通に使えました。

今回は、異なる CPUで、IICの関数がある。 という想定で 移植性を優先して**初期化処理**、**画面消去処理**、**文字表示処理**の OLEDに出す、コマンドバイト列のサンプルを 示します。

一応、私の環境では動きましたが
どのような場合でも動く事を保障する
訳ではありませんので、自己責任で
宜しくお願いします。

128x32 初期化／計:19byte

00h // コマンドモード

A8h, 3fh // Set Mux Ratio

D3h, 00h // Set Display Offset

40h // Set Display Start Line

A0h // Set Segment Re-map

C0h // Set COM Output Scan Direction

DAh, 02h // Set COM Pins hardware configuration

81h, 7Fh // Set Contrast Control

A4h // Disable Entire Display On

A6h // Set Normal Display

D5h, 80h // Set OSC Frequency

8Dh, 14h // Enable Charge pump regulator

AFh // Display On

Return

128x64 初期化／計:18byte

00h // コマンドモード

D5h, 80h // Set OSC Frequency

8Dh, 14h // Enable Charge pump regulator

20h, 00h // Set Memory Addressing Mode

DAh, 12h // Set COM Pins hardware configuration

81h, CFh // Set Contrast Control

D9h, F1h // Set Pre-charge Period

DBh, 40h // Set Vcomh Deselect Level

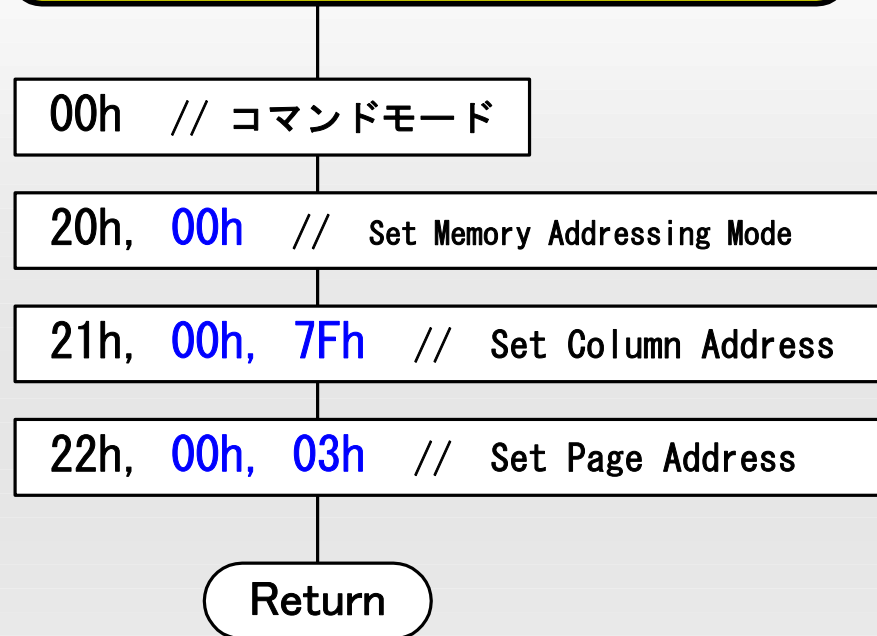
A4h // Disable Entire Display On

A6h // Set Normal Display

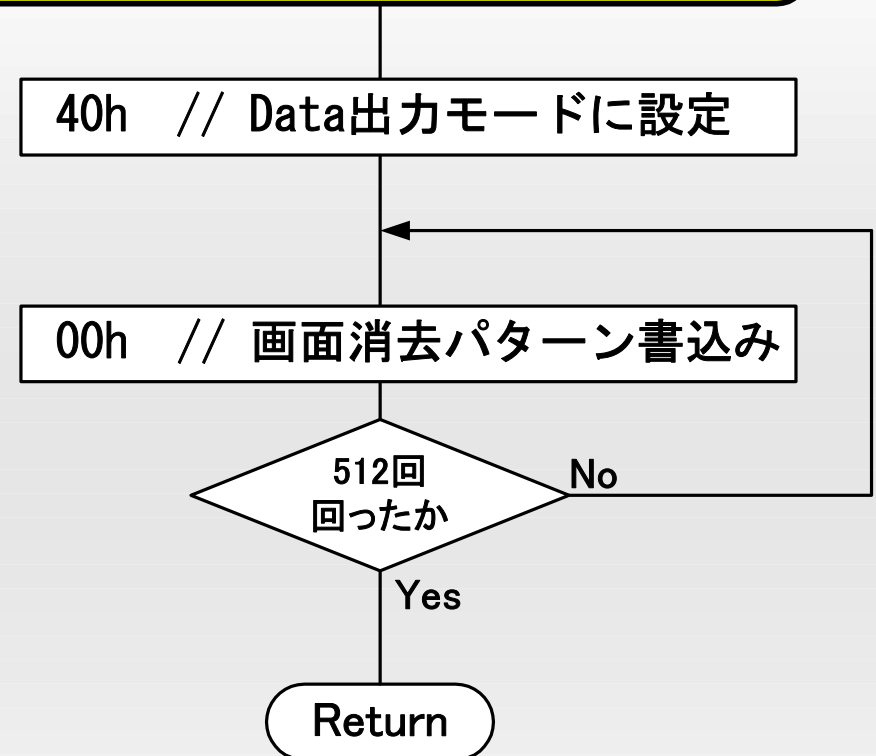
AFh // Display On

Return

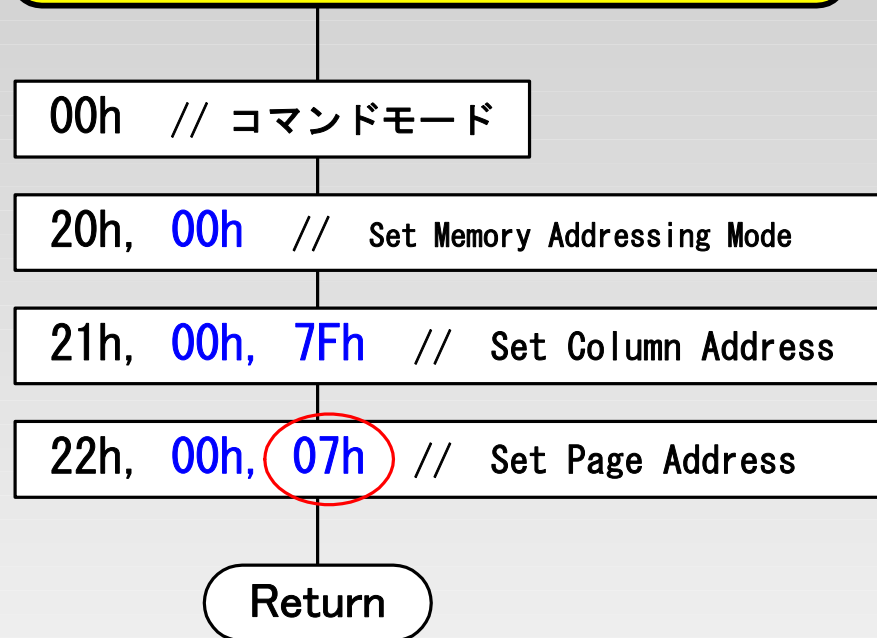
128x32 画面消去(1)／計:9byte



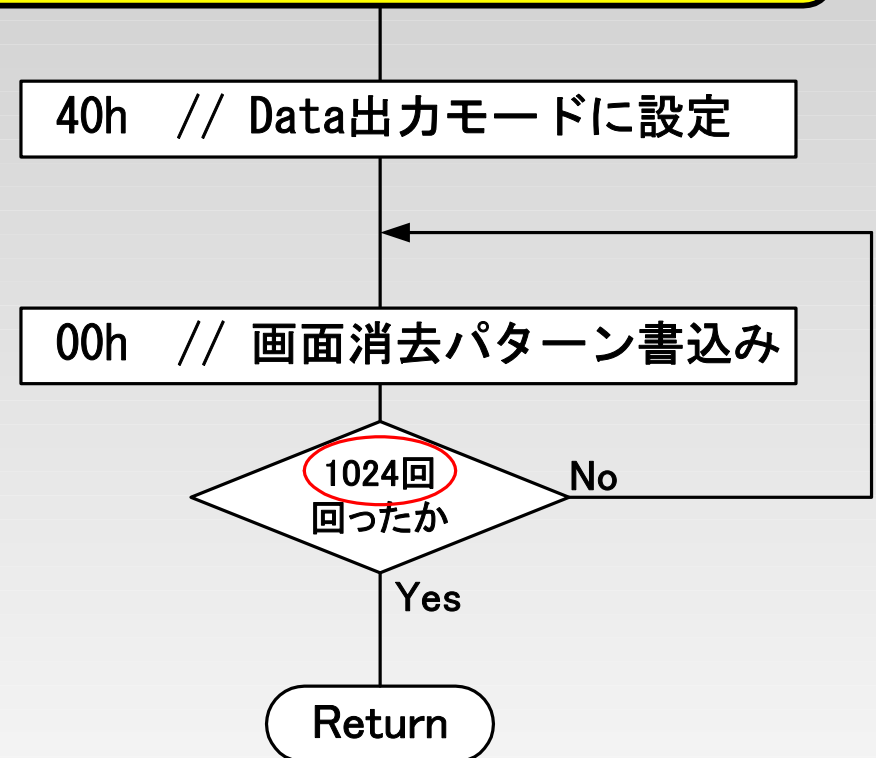
128x32 画面消去(2)／計: 513byte



128x64 画面消去(1)／計:9byte



128x64 画面消去(2)／計: 1025byte



(共通)文字表示(1)

00h // コマンドモード

20h, 00h // Set Memory Addressing Mode

22h, Ps, Pe // Set Page Address
// Ps : 開始ページ番号
// Pe : 最終ページ番号

21h, As, Ae // Set Column Address
// As : 開始アドレス
// Ae : 終了アドレス

Return

文字のロケートに関わる
PS, PE, As, Ae の計算式は
[dsd_OLED_sub.c](#)を 参照の事。

(共通)文字表示(2)

40h // Data出力モードに設定

??h // 文字FontData書込み

所定回数
回ったか

No

Yes

Return

所定回数は、FontDataのByte数で
8x16dotの場合は、16byteで
16x32dotの場合は、64byteに
なります。

IICアドレスの設定

今回の OLEDのアドレスは、128x32の物と、128x64の物は、デフォルトでは同じ **3Ch** の IICアドレスです。

さしあたり IIC BUSに 1個接続するだけであれば、そのまま接続して使用する事が出来ます。

今回、IIC BUSに 128x32dot OLEDと 128x64dot OLEDの 2個のデバイスをパラって接続しようと考えます。

その場合どちらか片方のIICアドレスを、変更する必要があります。

128x32dotの OLEDは、見る限り IICアドレスの設定箇所が、ありません。

128x64dotの OLEDには、IICアドレスの設定が裏側に ありました。

IIC設定箇所の アドレス表示が、0x78 と 0x7A になっています。0x78側が **3Ch** で 0x7A側が **3Dh** です。(0x78 と 0x7Aを 1bit右シフトすると 0x3Cと 0x3Dです。)

基板には、0x78側に **472のチップ抵抗**がハンダ付け

されています。
(ややピンボケですみません。)

ちなみに抵抗の
大きさは 1608
長さ : 1.6mm
幅 : 0.8mmです。



IICアドレスの変更

という事で、472(4.7K Ω)のチップ抵抗を、一旦外して、隣の0x7A側に 移し換える必要があります。過去にチップ抵抗を 基板に半田付けした事がありますが、半田付けしたチップ抵抗を取り去る事は、やった事はありませんでした。

外す時は、チップ抵抗両端の端子にハンダを盛って両端を半田でブリッジさせてハンダごての熱を両端の端子に伝える事で簡単に外せるという事です。

外したチップ抵抗を隣に移し替えてハンダ付けしたのですが、**OLEDが、動かなくなりました。**

で、1608サイズの 472チップ抵抗を別途用意して、チップ抵抗を交換したらOLEDが、動きだしました。

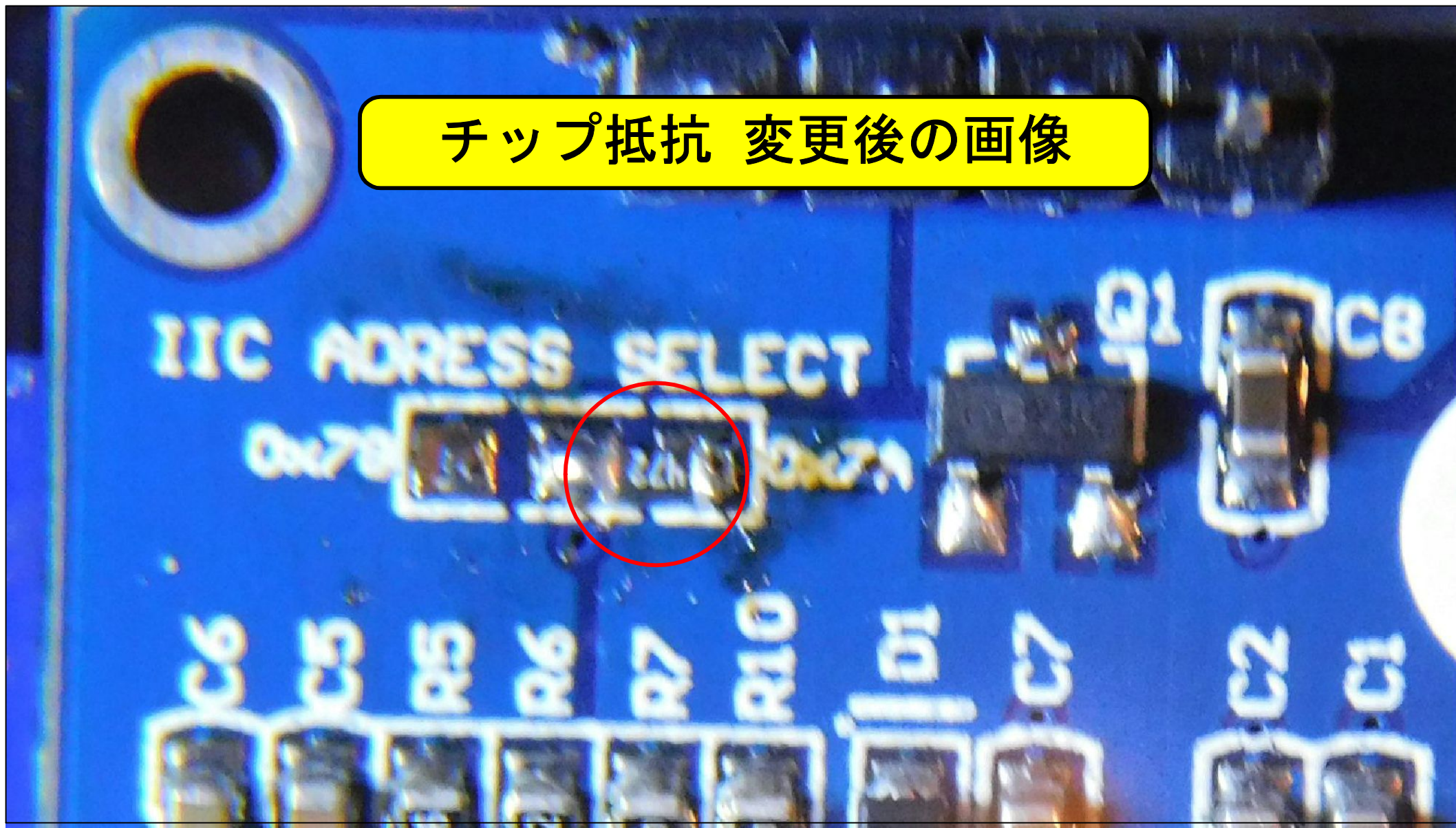
外す時、チップ抵抗を壊したのかもしれない。両側の端子の片側が、外れたように見えてました。(**アドレス変更を行う際は、予備のチップ抵抗を用意しておいた方が、いいです。**)



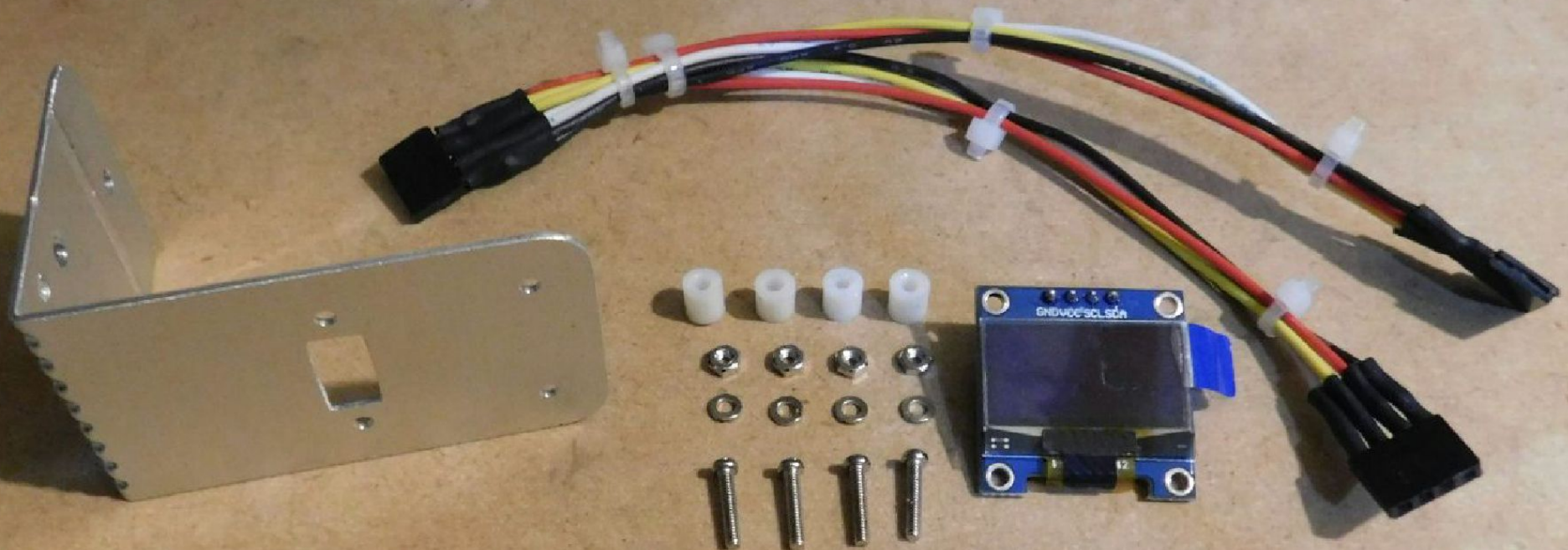
チップ抵抗の入ったテープです。
通常リールに巻いてあります。

ハンダ付けの際は、先の尖ったハンダこてとピンセット それとフラックス
ハンダ吸取り線が、必要となります。

チップ抵抗 変更後の画像



128x64dot OLEDのスタンドを 今回も用意しました。



それと、IICの 二股ケーブルも用意しました。

OLED表示器を2つ並べて
見れる様にしました。

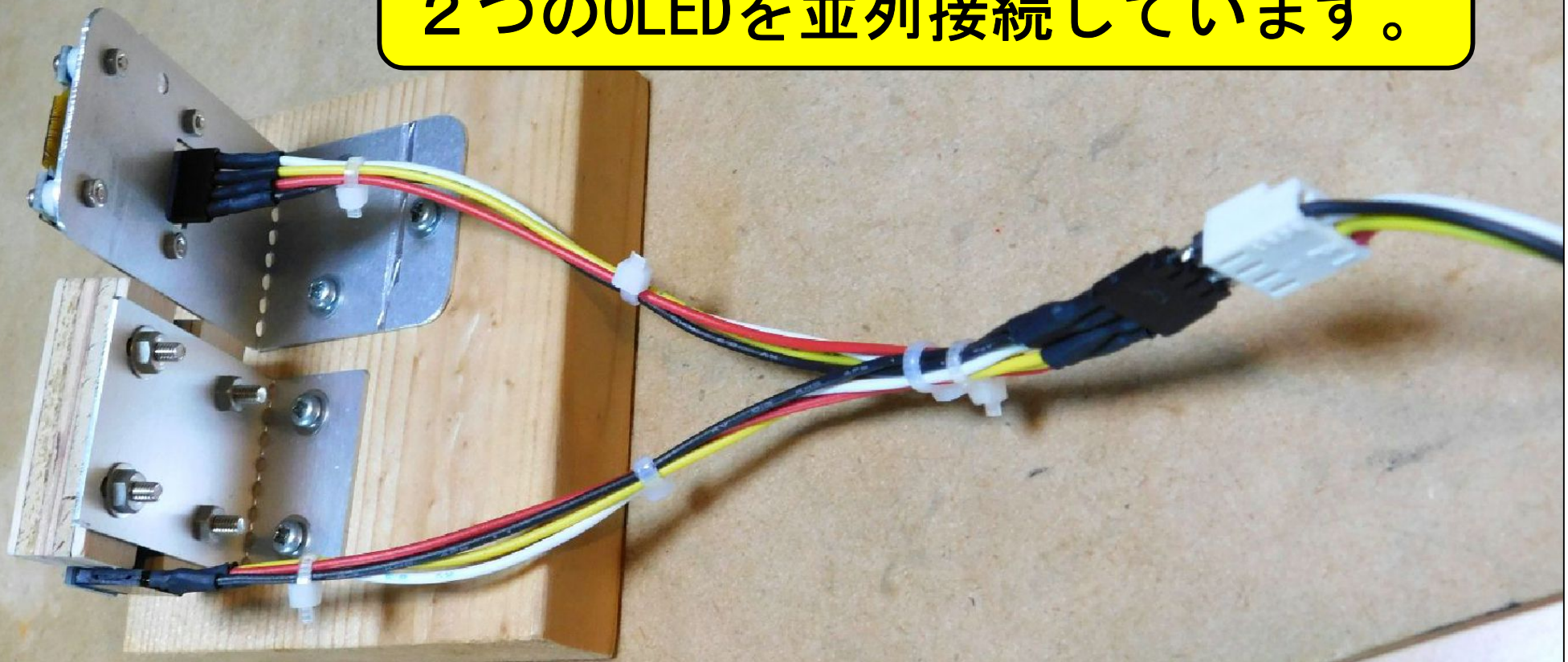
21-09-03
17:00:00

3D

12:34:56

3C

裏側は 二股ケーブルを使い
2つのOLEDを並列接続しています。



PC-マイコン間の通信コマンド

前回のコマンドのままでは、2台のOLEDをアクセスする事が出来ないので今回、通信コマンドの仕様変更を行いました。

まずコマンド文字列の先頭に @ を付ける事にしました。先頭に @ が無いとマイコン側ではコマンドを無視します。コマンドは、5種類です。

1. @ini=ad,sz[Cr] // 初期化
2. @sel=ad[Cr] // カレントデバイス選択
3. @cls[Cr] // 画面表示消去
4. @fxy:abcdefg[Cr] // 8x16dot文字表示
5. @Fxy:abcdefg[Cr] // 16x32dot文字表示

[Cr]は、[Enter]キー(0Dh)です。

1. 初期化コマンド
@ini=ad,sz[Cr] // adは IIC Address
// szは OLED縦dot数(32 or 64)
2. カレントデバイス選択
@sel=ad[Cr] // adは IIC Address
3. 画面表示消去
@cls[Cr] // デバイスは、@ini また
// は、@selで最後に選択
// した物になります。
4. 8x16dot文字列表示
@fxy:abcdefg[Cr]
// xx : カラム位置 0~15
// y : 行位置0~1、0~3
// abcdefg : 表示文字列
5. 16x32dot文字列表示
@Fxy:abcdefg[Cr] // パラメータは
// 4. と同じ

終りに

前回、今回と 小さいOLED（有機EL）ディスプレイのアクセスに関わる動画を作りましたがいかがだったでしょうか。

この OLED表示器は、小さいけれど明るくて見やすいので、今後携帯用などの小型機器に使用されそうな気がします。

消費電力も少ないです。 バックライト付きのLCDディスプレイよりも少ないのではと思われます。

用途によっては、OLEDディスプレイを複数使用も有りかなと思ったので、IIC BUSに複数のデバイスを接続出来るようにしました。

この小さい OLEDディスプレイを使ってみたいという方に参考になれば幸いです。

今回も、R8Cマイコンのプログラム（ HEWの Project File 一式 ）もダウンロード出来るようにしておきます。