今回の R8C/M120A I/Oポート表

ポート	ビット	ピン	出/入	接続先
ADC	AN0	20	A in	VR
Port.1	P1_1	19	in	PBSW_1
	P1_2	18	in	PBSW_2
	P1_3	17	out	NC
	TxD	16	S out	CP2102/RXI
	RxD	15	S in	CP2102/TXO
	RxD_{PRG}	14	S in	CP2102/TXO
	P1_7	13	out	/CS1

ポート	ビット	ピン	出/入	接続先
Port.3	P3_3	11	out	/CS2
	P3_4	10	out	赤LED
	P3_5	9	out	緑LED
	P3_7	2	out	基板LED

ポート	ビット	ピン	出/入	接続先
Port.4	P4_2	1	out	SDA/MOSI
	P4_5	12	out	SCL/SCLK
	P4_6	6	in	MISO
	P4_7	4	out	/CS0

私のR8Cマイコンのプログラム

過去の私の動画を見て、百円マイコン（R8C）の HEWプロジェクト一式を ダウンロードされた方も、おられると思います。

但しプログラムソースに関わる説明を、今まで殆どしていませんでした。 私の R8Cマイコンのプログラムソースは、初心者にとっては、ソースファイルが、やたら多くて ちょっと難しく見えるかなと思います。

ここでは、C言語を 多少扱われた方を 対象に説明します。 私のプログラムは、C言語とR8Cのアセンブラと 混在させた、I/O処理のサブルーチンライブラリプログラムを 作成しています。

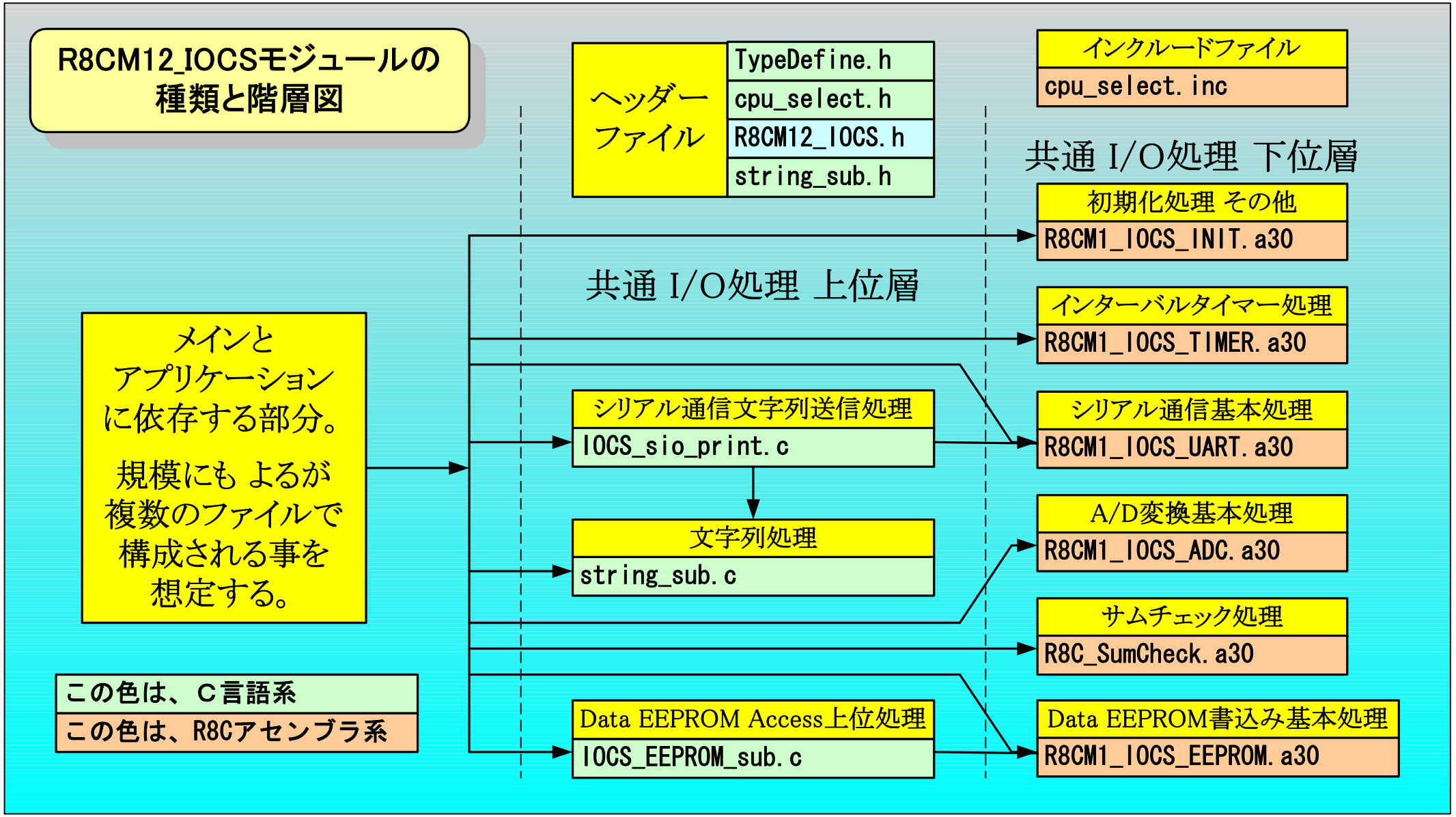
C言語のソースファイル拡張子は .c で、ヘッダーファイルが .h です。

R8Cマイコンの アセンブラ言語のソースファイル拡張子は .a30 で、インクルードファイルが .inc です。

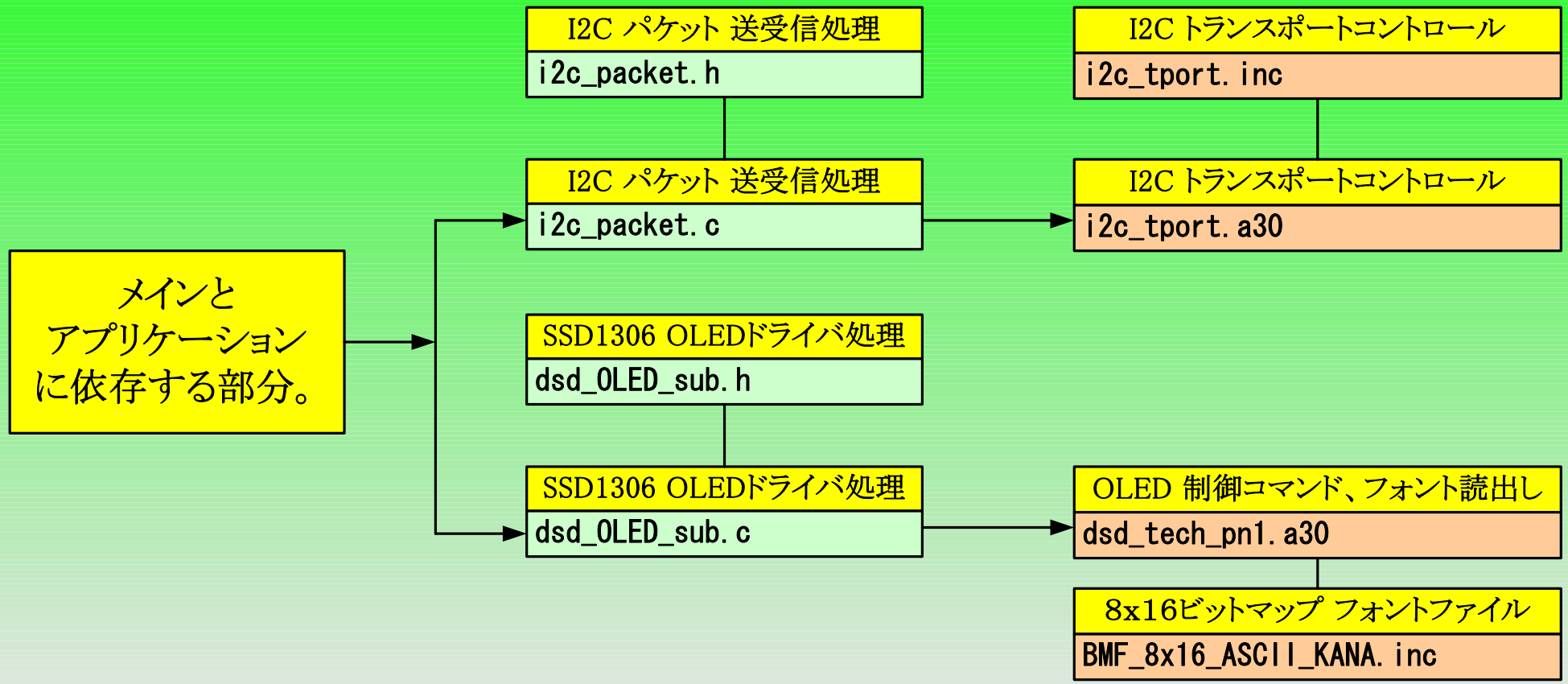
私が作ったI/O処理ライブラリのソースファイルは、R8CM1_IOCS.... というファイル名と、IOCS.... というファイルの ファイルが複数存在します。 I/O処理サブルーチンが、機能ごとに分かれたファイルになっています。

次のページで、私の IOCSプログラムの階層構造図を示します。 各ファイルが、どのような役割を果たしているか、凡そ 分かると思います。

関数の一覧は、R8CM12_IOCS.h を 参照して下さい。 各関数の引数の仕様等は、その関数を実装するソースファイルを 参照して下さい。


R8CM12_IOCSモジュールの
種類と階層図
ヘッダー
ファイル
TypeDefine.h
cpu_select.h
R8CM12_IOCS.h
string_sub.h
インクルードファイル
cpu_select.inc
共通 I/O処理 下位層
初期化処理 その他
R8CM1_IOCS_INIT.a30
共通 I/O処理 上位層
インターバルタイマー処理
R8CM1_IOCS_TIMER.a30
メインと
アプリケーション
に依存する部分。
規模にも よるが
複数のファイルで
構成される事を
想定する。
シリアル通信文字列送信処理
IOCS_sio_print.c
シリアル通信基本処理
R8CM1_IOCS_UART.a30
文字列処理
string_sub.c
A/D変換基本処理
R8CM1_IOCS_ADC.a30
サムチェック処理
R8C_SumCheck.a30
この色は、C言語系
この色は、R8Cアセンブラ系
Data EEPROM Access上位処理
IOCS_EEPROM_sub.c
Data EEPROM書込み基本処理
R8CM1_IOCS_EEPROM.a30

I2Cモジュールと OLED制御IC SSD1306モジュール 種類と階層図



HEWの 新規Projectで IOCSや I2Cを使うには

今回、IOCSや、I2C、OLEDのドライバソフトの構成の概要を 説明しました。

でも HEWで、新規Projectを作成して、そのProjectに IOCSや I2Cの機能を 組み込むにはどうすれば、いいのか。？
HEW初心者の方は、悩みますよね。

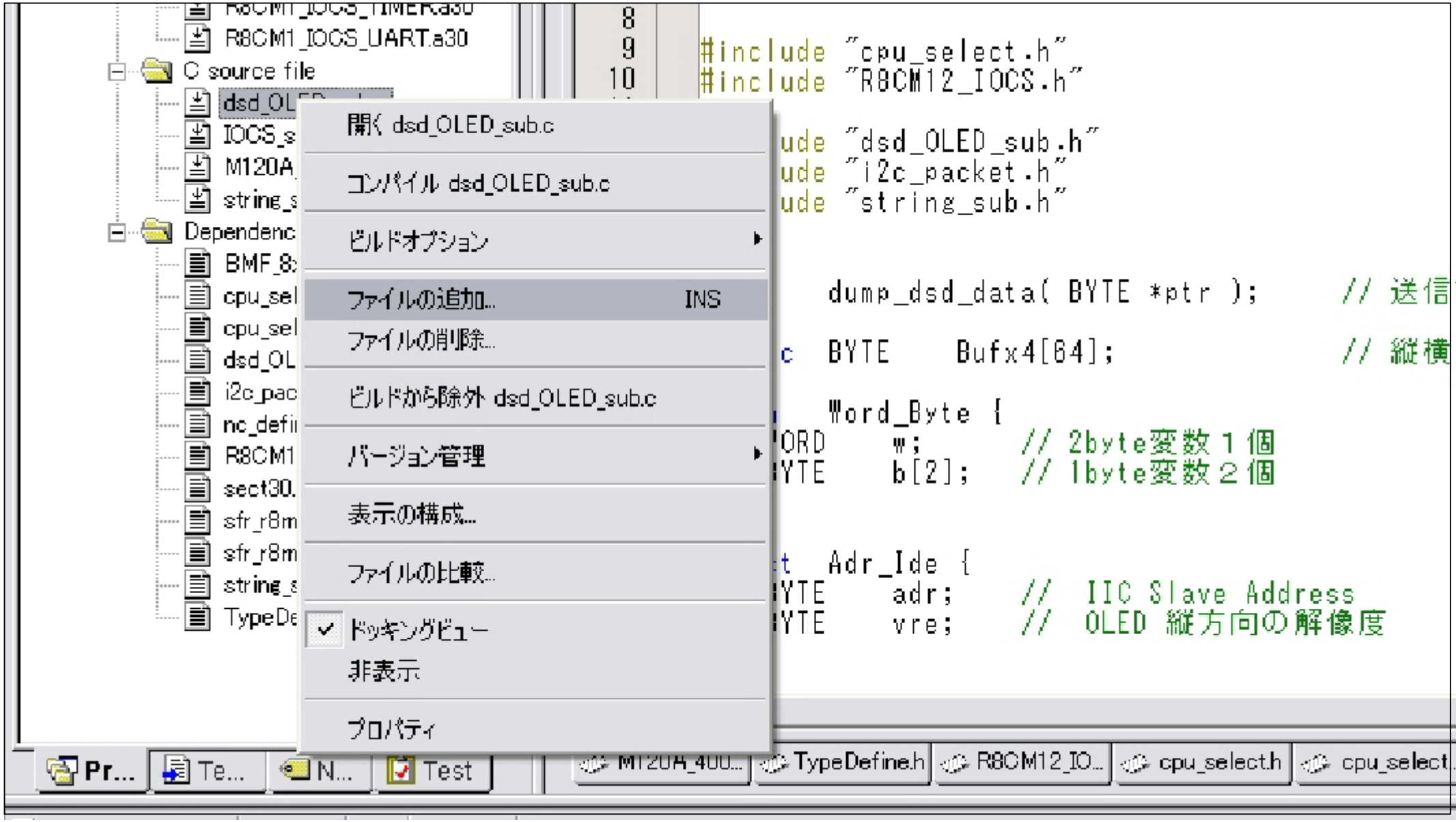
秋月電子の R8C開発キットには、最低限のHEWの使い方が、説明されていますが、ソースファイルが、メインのファイル１本で、俗にいうLチカの単純なプログラムなので、本格的な使い方とは、ほど遠いですよね。

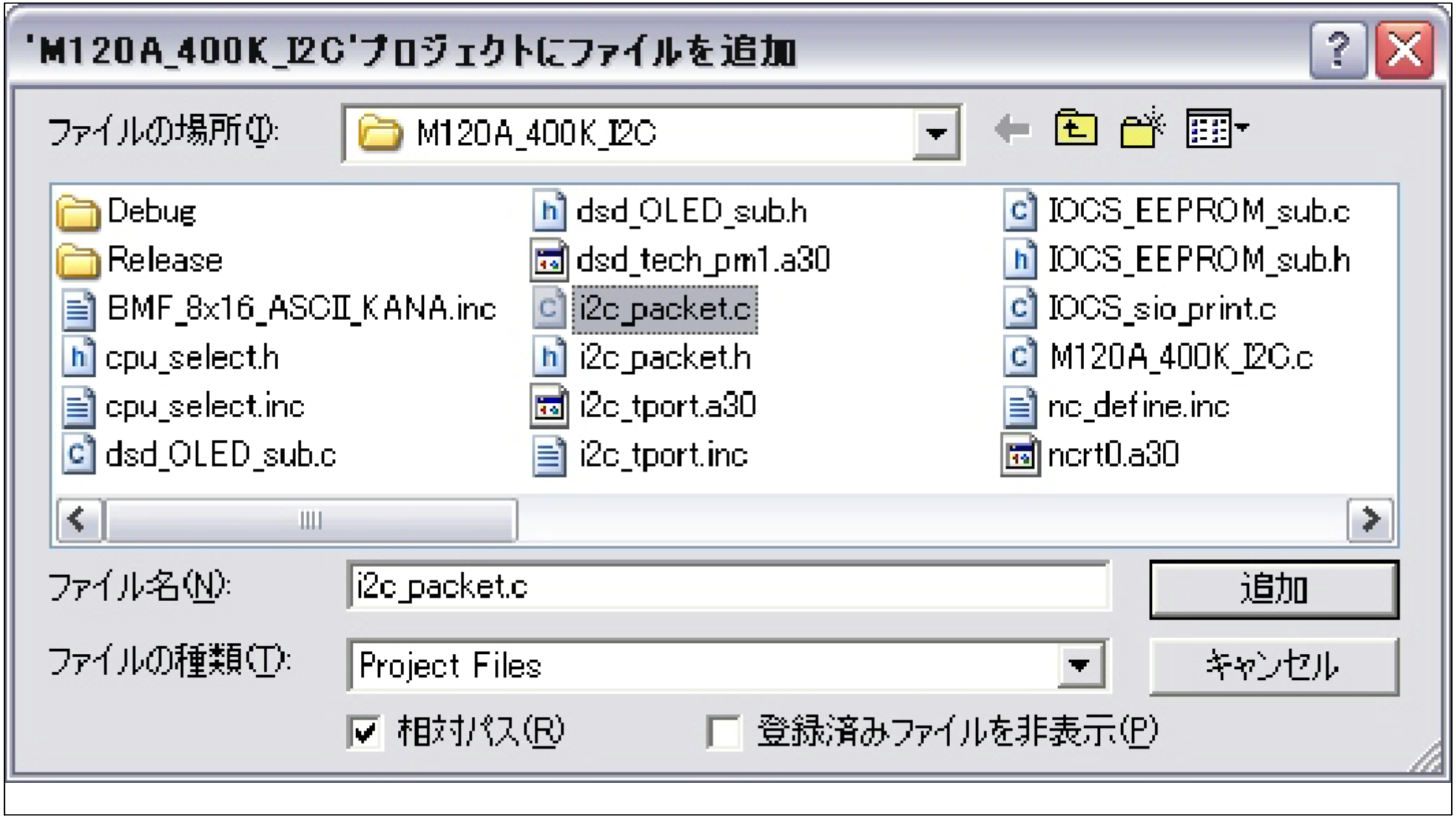
よって今回、IOCSなどの 複数のソースを、HEWProjectに登録して、使える様にする方法を、少し説明します。

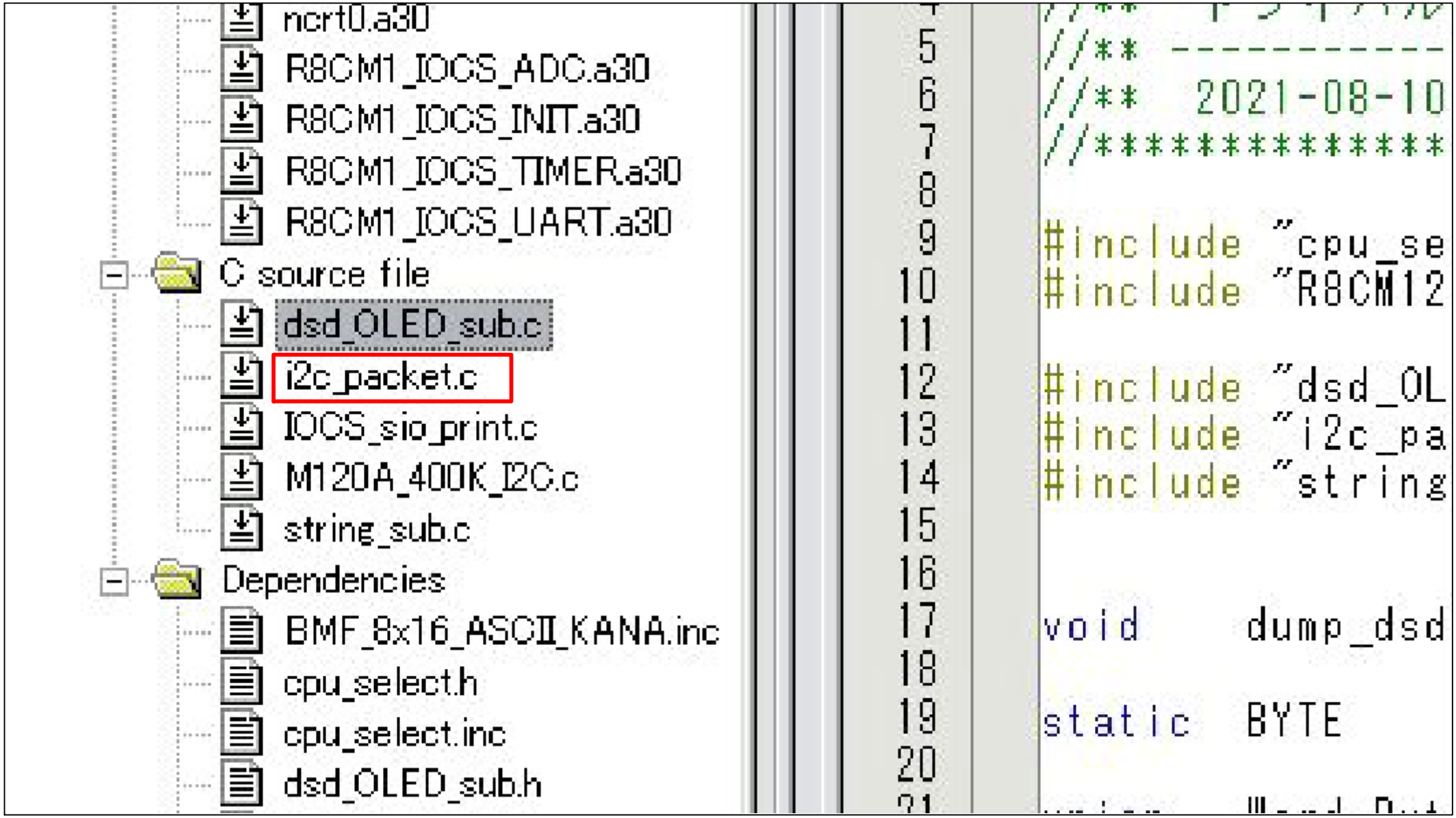
まず、ダウンロードしたファイルから、IOCSの種類と階層図で説明したソースファイル、ヘッダーファイル、インクルードファイルを、目的のHEW Projectの メインのソースファイルが、入っているフォルダにコピーして下さい。
必ず Projectを 生成した後に、コピーして下さい。

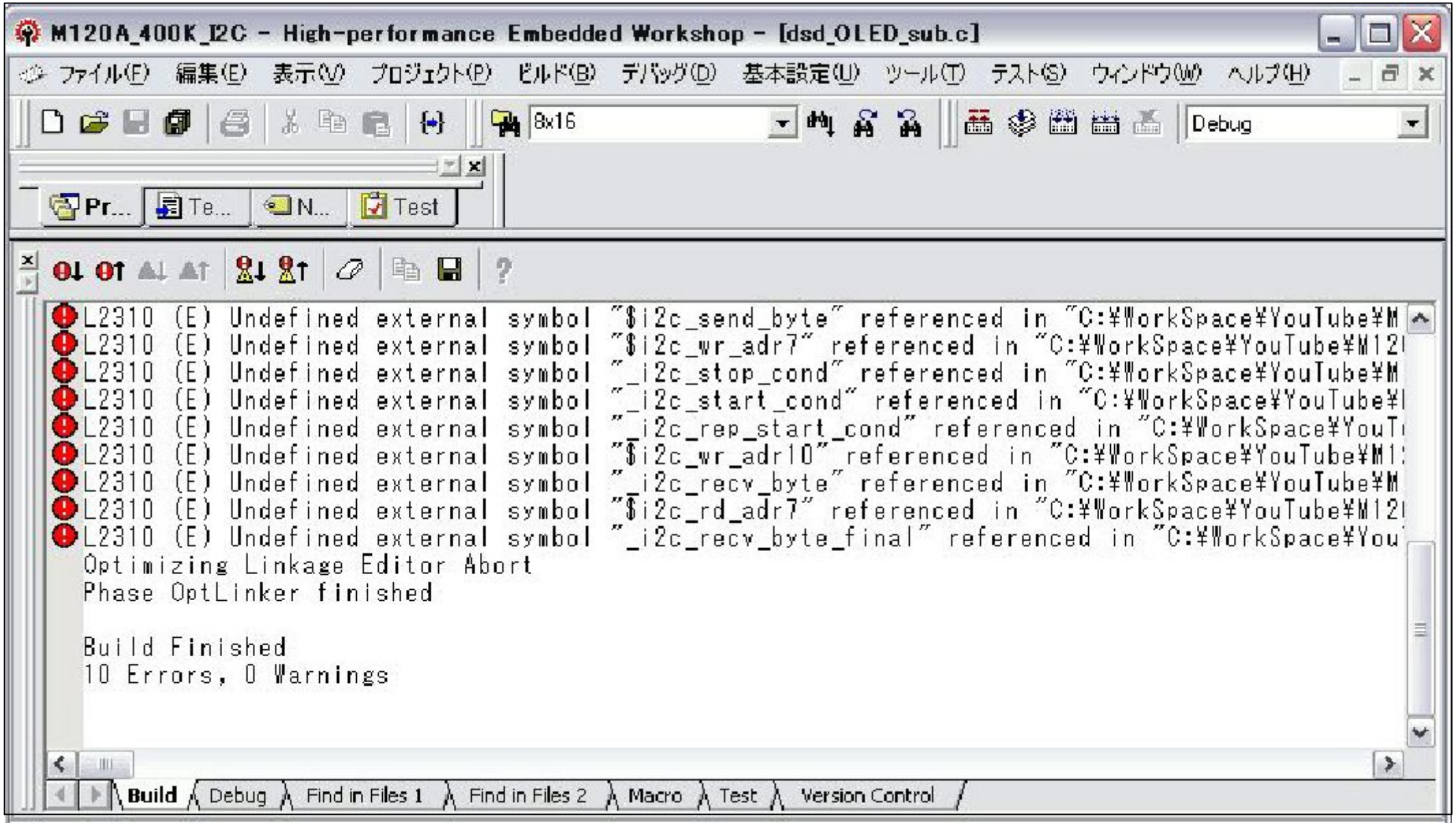
でも、Projectの ソースフォルダに IOCSのソースを、コピーしただけでは、ダメで Projectに 追加するソースファイルを、認識させるため追加登録する必要があります。

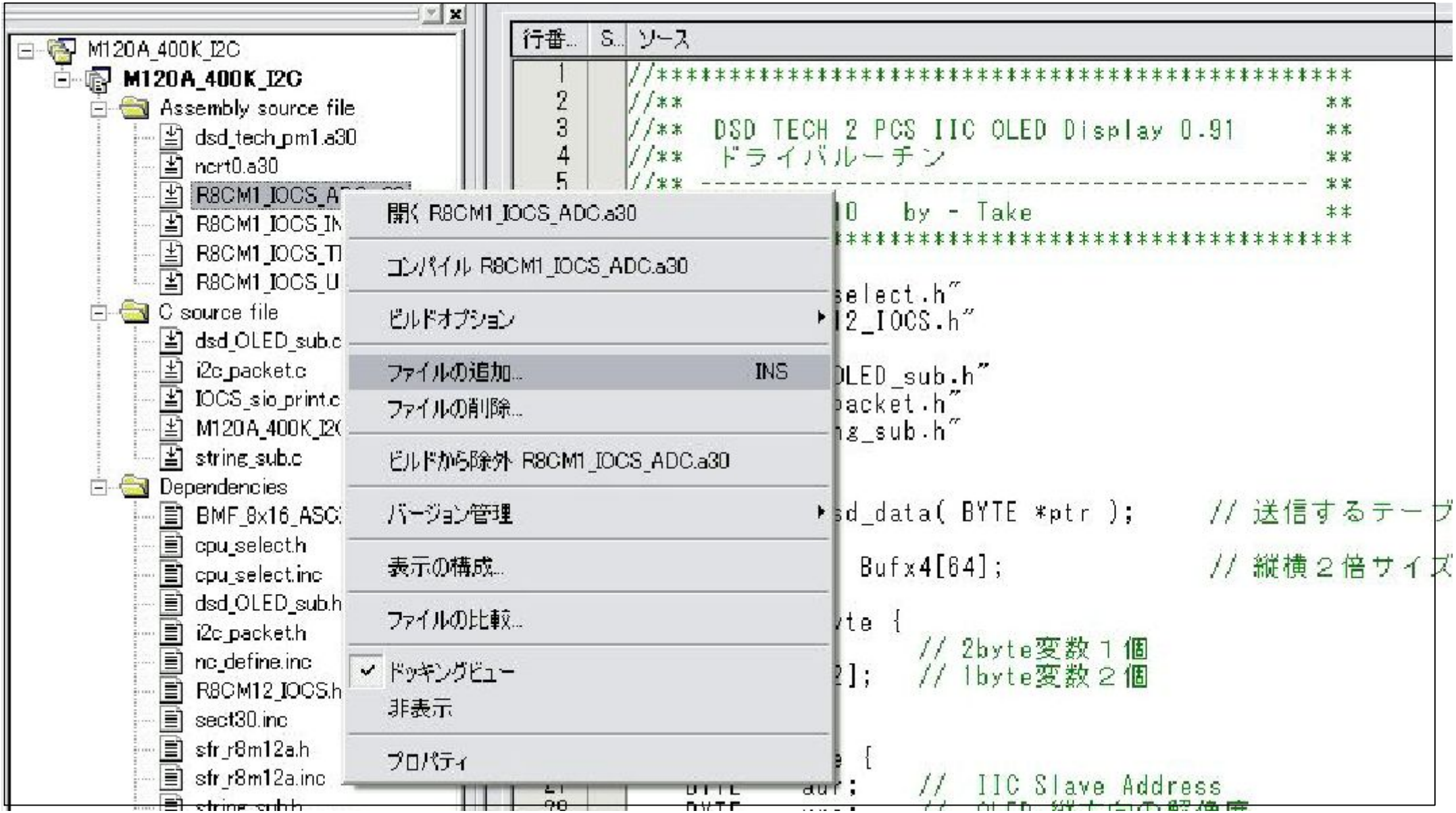
その方法を、画面のビットマップを見ながら説明します。

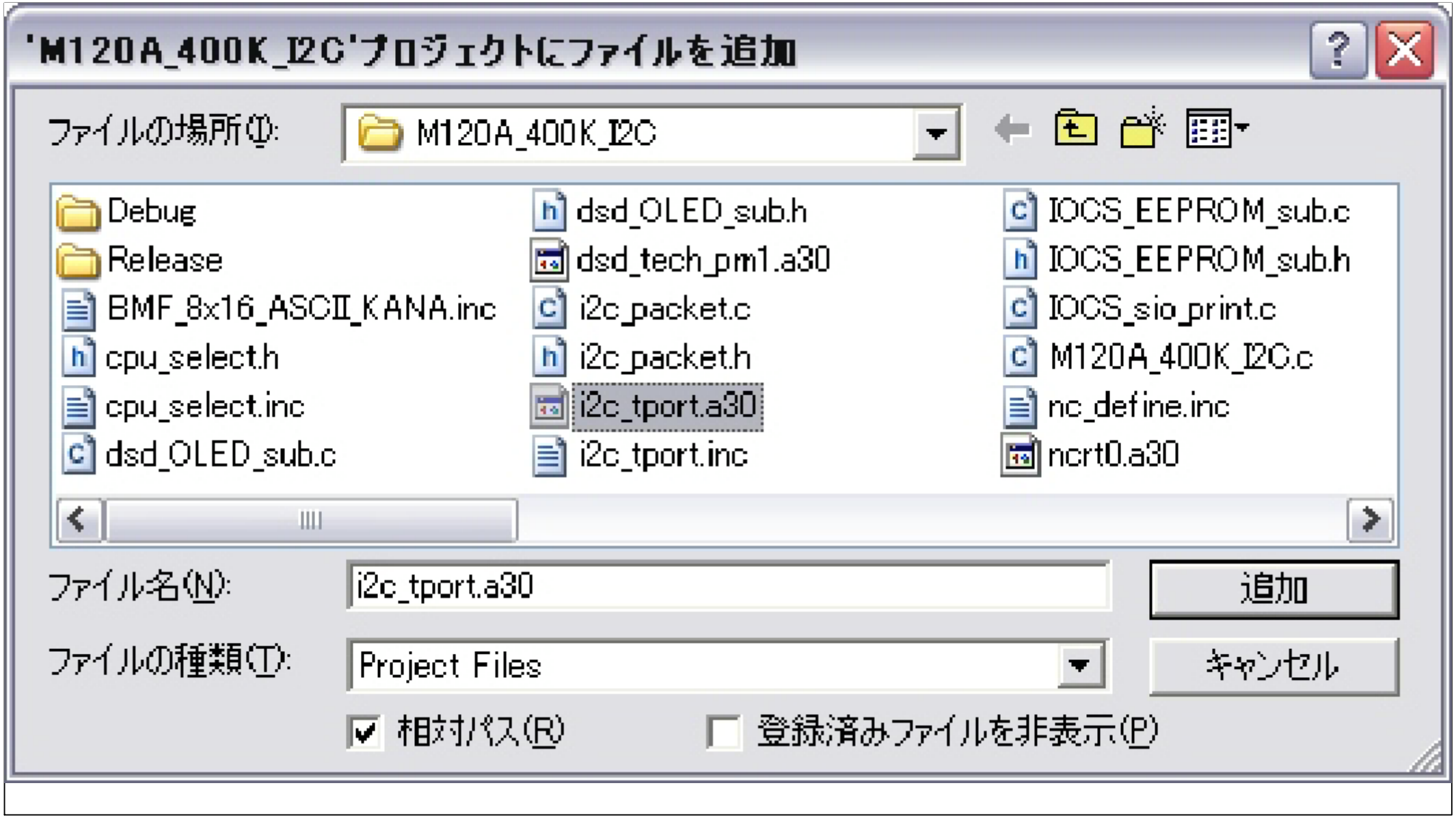

R8CM1_IOCS_UART.a30
C source file
dsd_OL
IOCS_s
M120A
string_s
Dependenc
BMF_8
cpu_sel
cpu_sel
dsd_OL
i2c_pac
nc_defi
R8CM1
sect30.
sfr_r8m
sfr_r8m
string_s
TypeDe
開く dsd_OLED_sub.c
コンパイル dsd_OLED_sub.c
ビルドオプション
ファイルの追加... INS
ファイルの削除...
ビルドから除外 dsd_OLED_sub.c
バージョン管理
表示の構成...
ファイルの比較...
ドッキングビュー
非表示
プロパティ
Pr...
Te...
N...
Test
8
9 #include "cpu_select.h"
10 #include "R8CM12_IOCS.h"
ude "dsd_OLED_sub.h"
ude "i2c_packet.h"
ude "string_sub.h"
dump_dsd_data(BYTE *ptr); // 送信す
c BYTE Bufx4[64]; // 縦横２
Word_Byte {
ORD w; // 2byte変数１個
YTE b[2]; // 1byte変数２個
t Adr_Ide {
YTE adr; // IIC Slave Address
YTE vre; // OLED 縦方向の解像度
M120A_400...
TypeDefine.h
R8CM12_IO...
cpu_select.h
cpu_select...

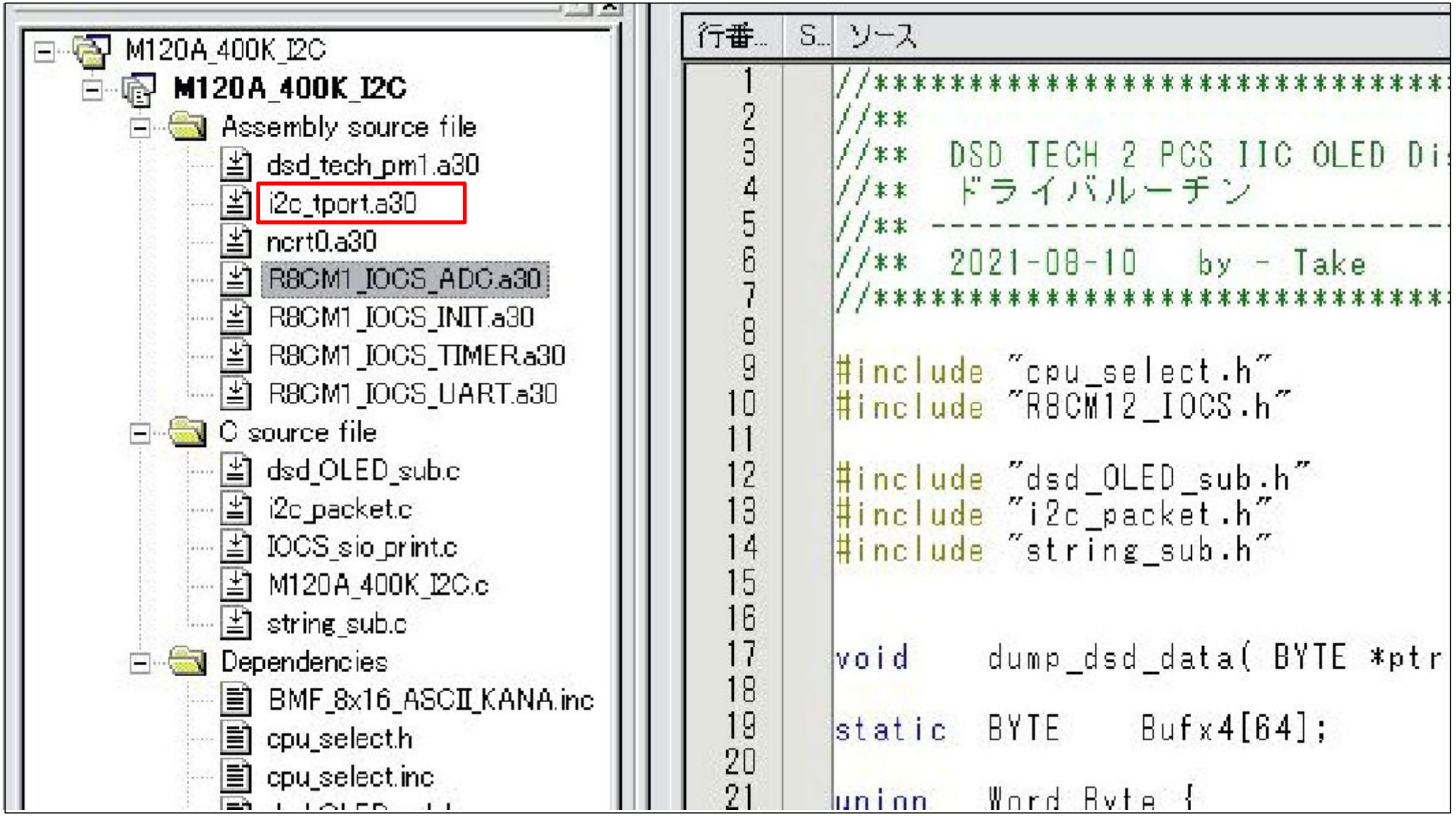

'M120A_400K_I2C'プロジェクトにファイルを追加
ファイルの場所(I):
M120A_400K_I2C
Debug
Release
BMF_8x16_ASCII_KANA.inc
cpu_select.h
cpu_select.inc
dsd_OLED_sub.c
dsd_OLED_sub.h
dsd_tech_pm1.a30
i2c_packet.c
i2c_packet.h
i2c_tport.a30
i2c_tport.inc
IOCS_EEPROM_sub.c
IOCS_EEPROM_sub.h
IOCS_sio_print.c
M120A_400K_I2C.c
nc_define.inc
ncrt0.a30
ファイル名(N):
i2c_packet.c
追加
ファイルの種類(T):
Project Files
キャンセル
相対パス(R)
登録済みファイルを非表示(P)

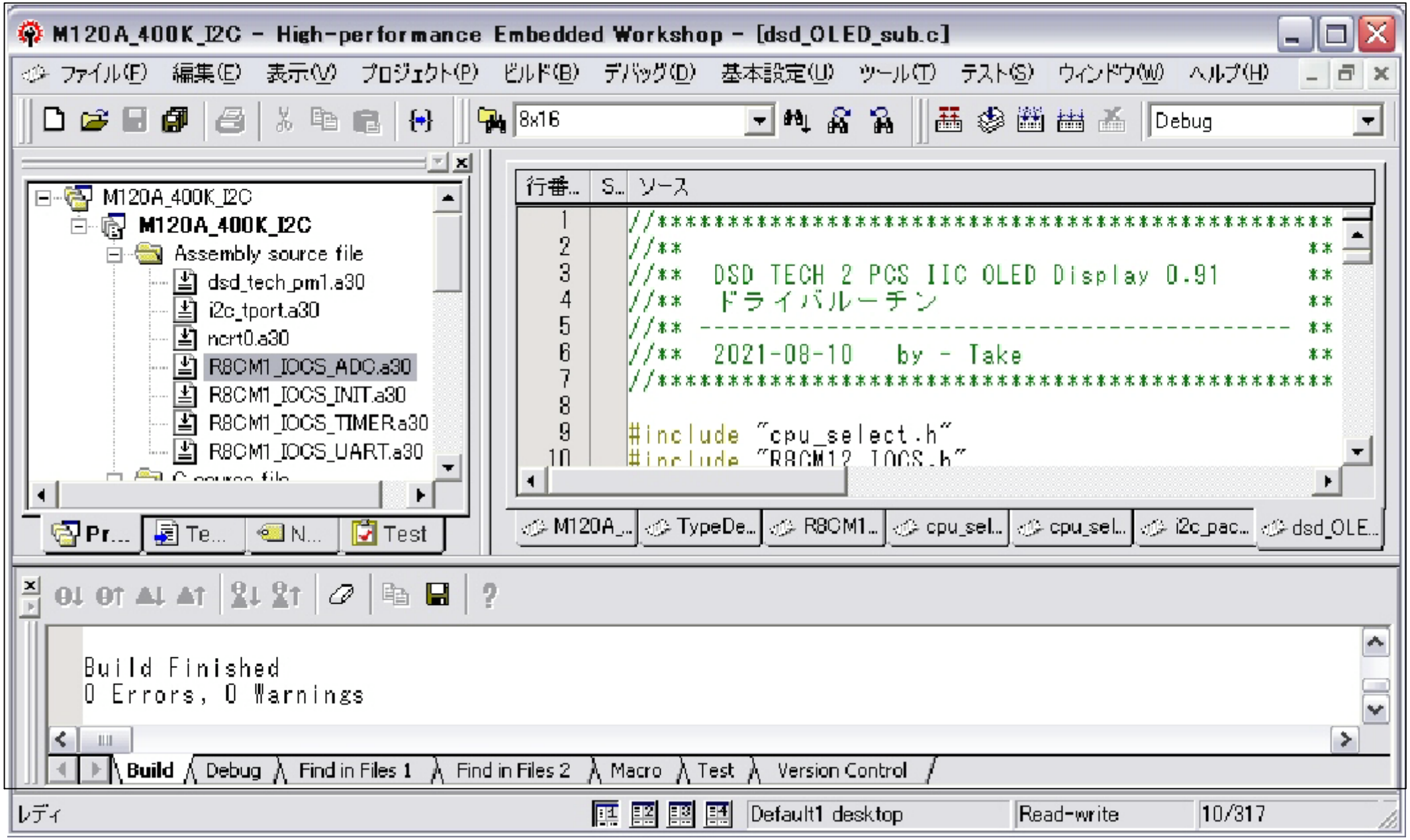

ncrt0.a30
R8CM1_IOCS_ADC.a30
R8CM1_IOCS_INIT.a30
R8CM1_IOCS_TIMER.a30
R8CM1_IOCS_UART.a30
C source file
dsd_OLED_sub.c
i2c_packet.c
IOCS_sio_print.c
M120A_400K_I2C.c
string_sub.c
Dependencies
BMF_8x16_ASCII_KANA.inc
cpu_select.h
cpu_select.inc
dsd_OLED_sub.h
5 //** ----------
6 //** 2021-08-10
7 //**************
9 #include "cpu_se
10 #include "R8CM12
12 #include "dsd_OL
13 #include "i2c_pa
14 #include "string
17 void dump_dsd
19 static BYTE

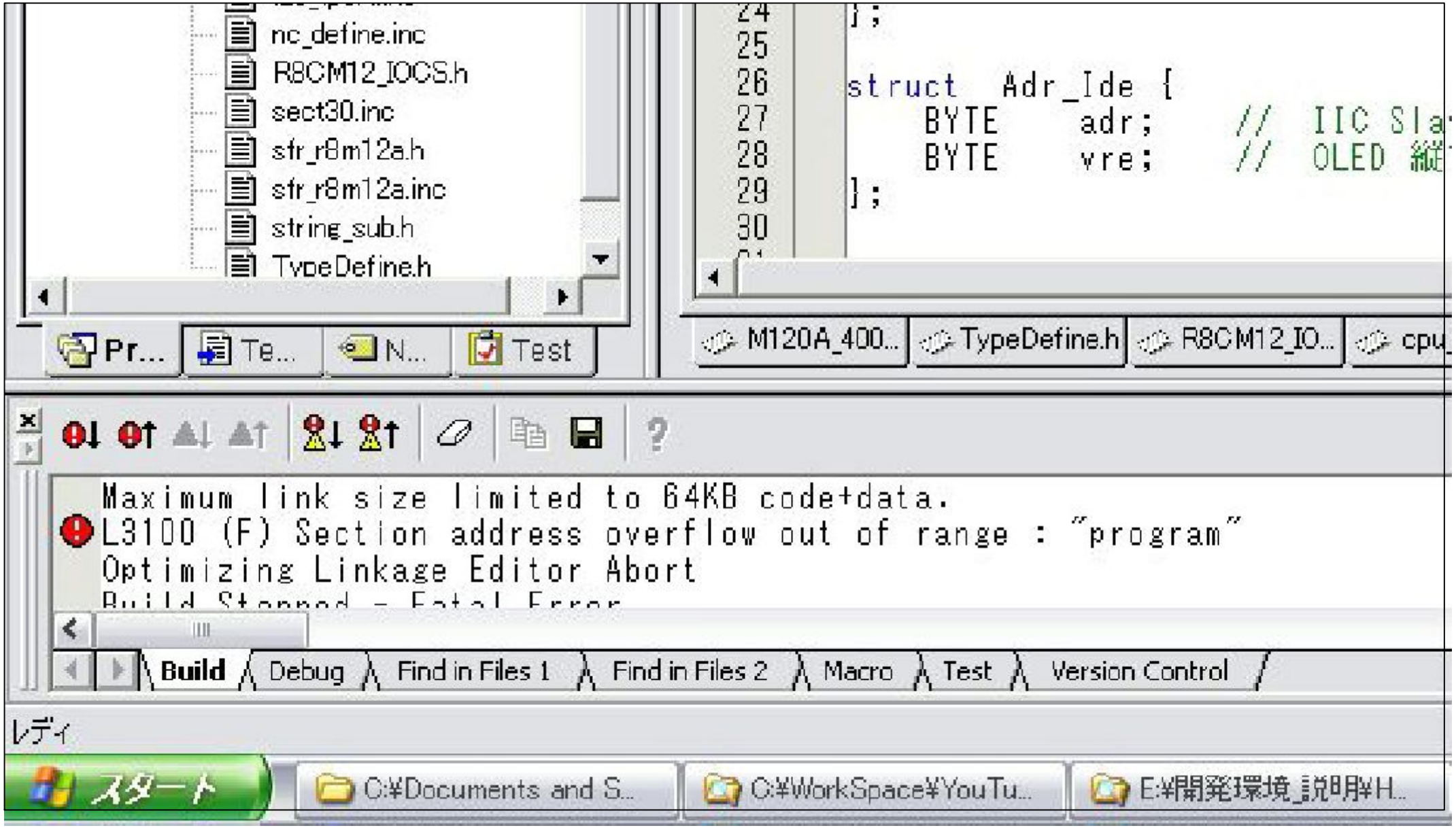

M120A_400K_I2C - High-performance Embedded Workshop - [dsd_OLED_sub.c]
ファイル(F) 編集(E) 表示(V) プロジェクト(P) ビルド(B) デバッグ(D) 基本設定(U) ツール(T) テスト(S) ウィンドウ(W) ヘルプ(H)
8x16
Debug
Pr... Te... N... Test
L2310 (E) Undefined external symbol "$i2c_send_byte" referenced in "C:¥WorkSpace¥YouTube¥M
L2310 (E) Undefined external symbol "$i2c_wr_adr7" referenced in "C:¥WorkSpace¥YouTube¥M12
L2310 (E) Undefined external symbol "_i2c_stop_cond" referenced in "C:¥WorkSpace¥YouTube¥M
L2310 (E) Undefined external symbol "_i2c_start_cond" referenced in "C:¥WorkSpace¥YouTube¥I
L2310 (E) Undefined external symbol "_i2c_rep_start_cond" referenced in "C:¥WorkSpace¥YouT
L2310 (E) Undefined external symbol "$i2c_wr_adr10" referenced in "C:¥WorkSpace¥YouTube¥M1
L2310 (E) Undefined external symbol "_i2c_recv_byte" referenced in "C:¥WorkSpace¥YouTube¥M
L2310 (E) Undefined external symbol "$i2c_rd_adr7" referenced in "C:¥WorkSpace¥YouTube¥M12
L2310 (E) Undefined external symbol "_i2c_recv_byte_final" referenced in "C:¥WorkSpace¥You
Optimizing Linkage Editor Abort
Phase OptLinker finished
Build Finished
10 Errors, 0 Warnings
Build Debug Find in Files 1 Find in Files 2 Macro Test Version Control

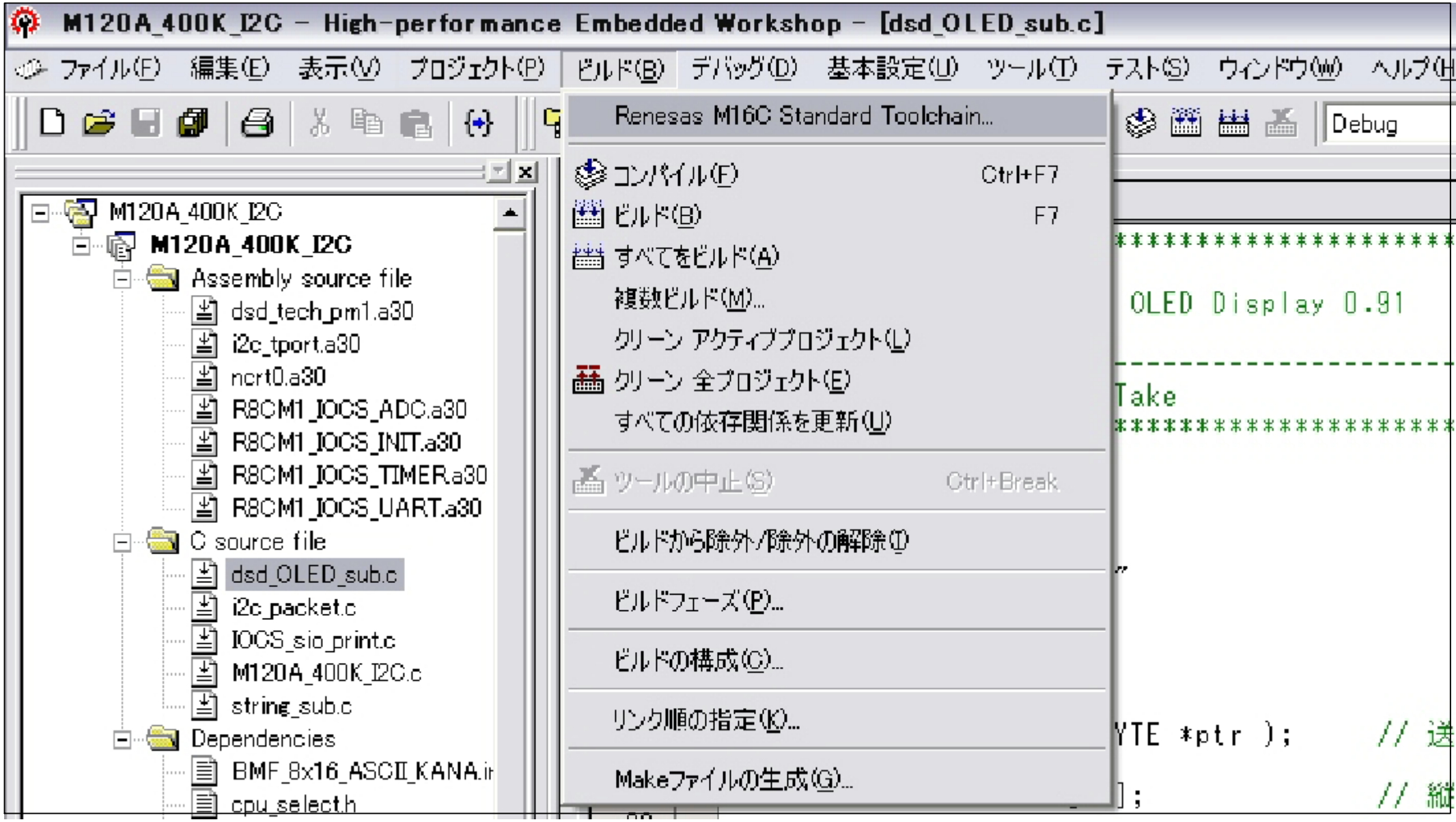

M120A_400K_I2C
M120A_400K_I2C
Assembly source file
dsd_tech_pm1.a30
ncrt0.a30
R8CM1_IOCS_A
R8CM1_IOCS_I
R8CM1_IOCS_T
R8CM1_IOCS_U
C source file
dsd_OLED_sub.c
i2c_packet.c
IOCS_sio_print.c
M120A_400K_I2
string_sub.c
Dependencies
BMF_8x16_ASC
cpu_select.h
cpu_select.inc
dsd_OLED_sub.h
i2c_packet.h
nc_define.inc
R8CM12_IOCS.h
sect30.inc
sfr_r8m12a.h
sfr_r8m12a.inc
開く R8CM1_IOCS_ADC.a30
コンパイル R8CM1_IOCS_ADC.a30
ビルドオプション
ファイルの追加... INS
ファイルの削除...
ビルドから除外 R8CM1_IOCS_ADC.a30
バージョン管理
表示の構成...
ファイルの比較...
ドッキングビュー
非表示
プロパティ
行番...
S...
ソース
//**********
//** **
//** DSD TECH 2 PCS IIC OLED Display 0.91 **
//** ドライバルーチン **
//** **
0 by - Take **
select.h"
2_IOCS.h"
OLED_sub.h"
packet.h"
ng_sub.h"
sd_data(BYTE *ptr); // 送信するテーブ
Bufx4[64]; // 縦横２倍サイズ
yte {
// 2byte変数１個
2]; // 1byte変数２個
e {
aur; // IIC Slave Address

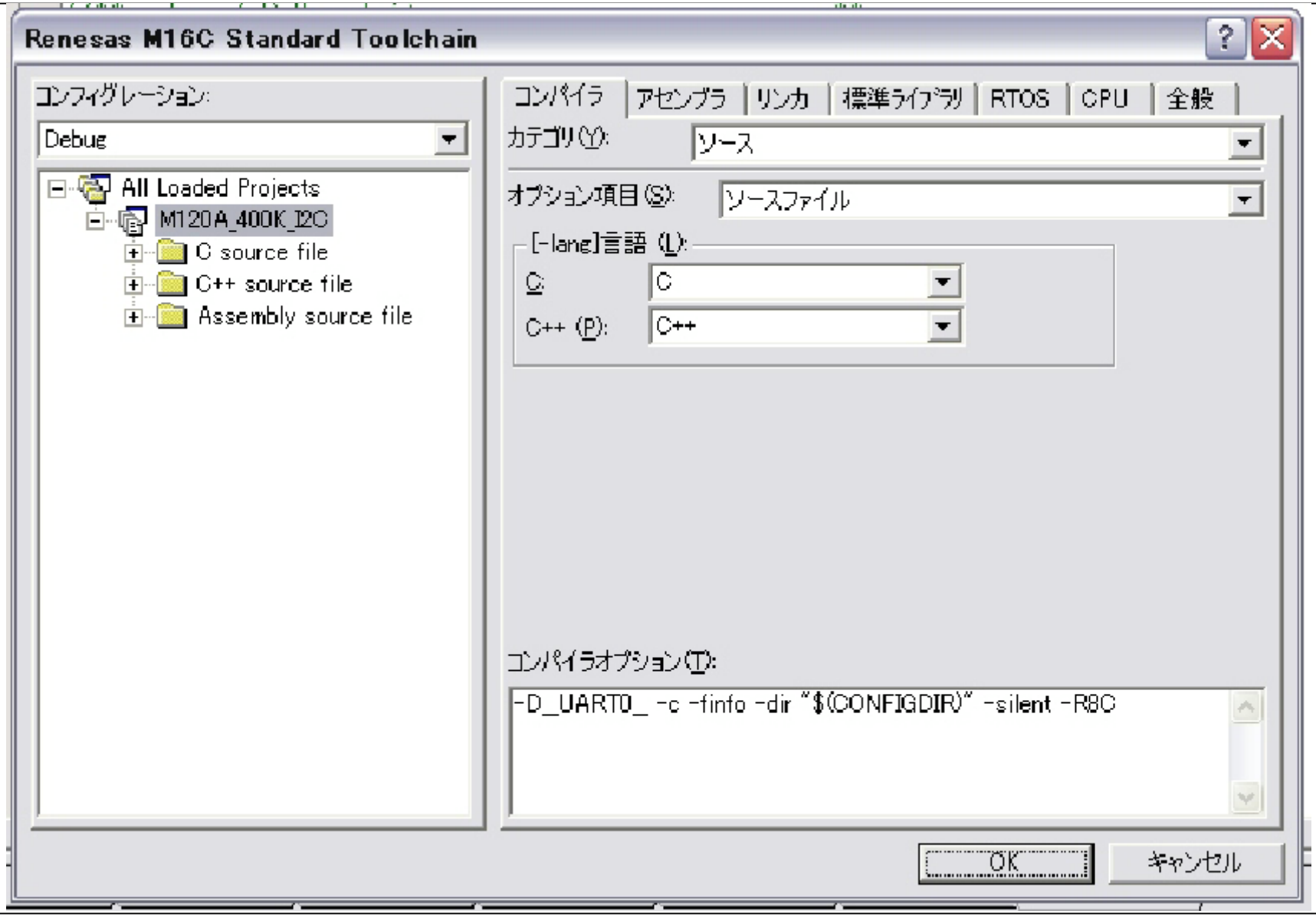

'M120A_400K_I2C'プロジェクトにファイルを追加
ファイルの場所(I):
M120A_400K_I2C
Debug
Release
BMF_8x16_ASCII_KANA.inc
cpu_select.h
cpu_select.inc
dsd_OLED_sub.c
dsd_OLED_sub.h
dsd_tech_pm1.a30
i2c_packet.c
i2c_packet.h
i2c_tport.a30
i2c_tport.inc
IOCS_EEPROM_sub.c
IOCS_EEPROM_sub.h
IOCS_sio_print.c
M120A_400K_I2C.c
nc_define.inc
ncrt0.a30
ファイル名(N):
i2c_tport.a30
追加
ファイルの種類(T):
Project Files
キャンセル
相対パス(R)
登録済みファイルを非表示(P)

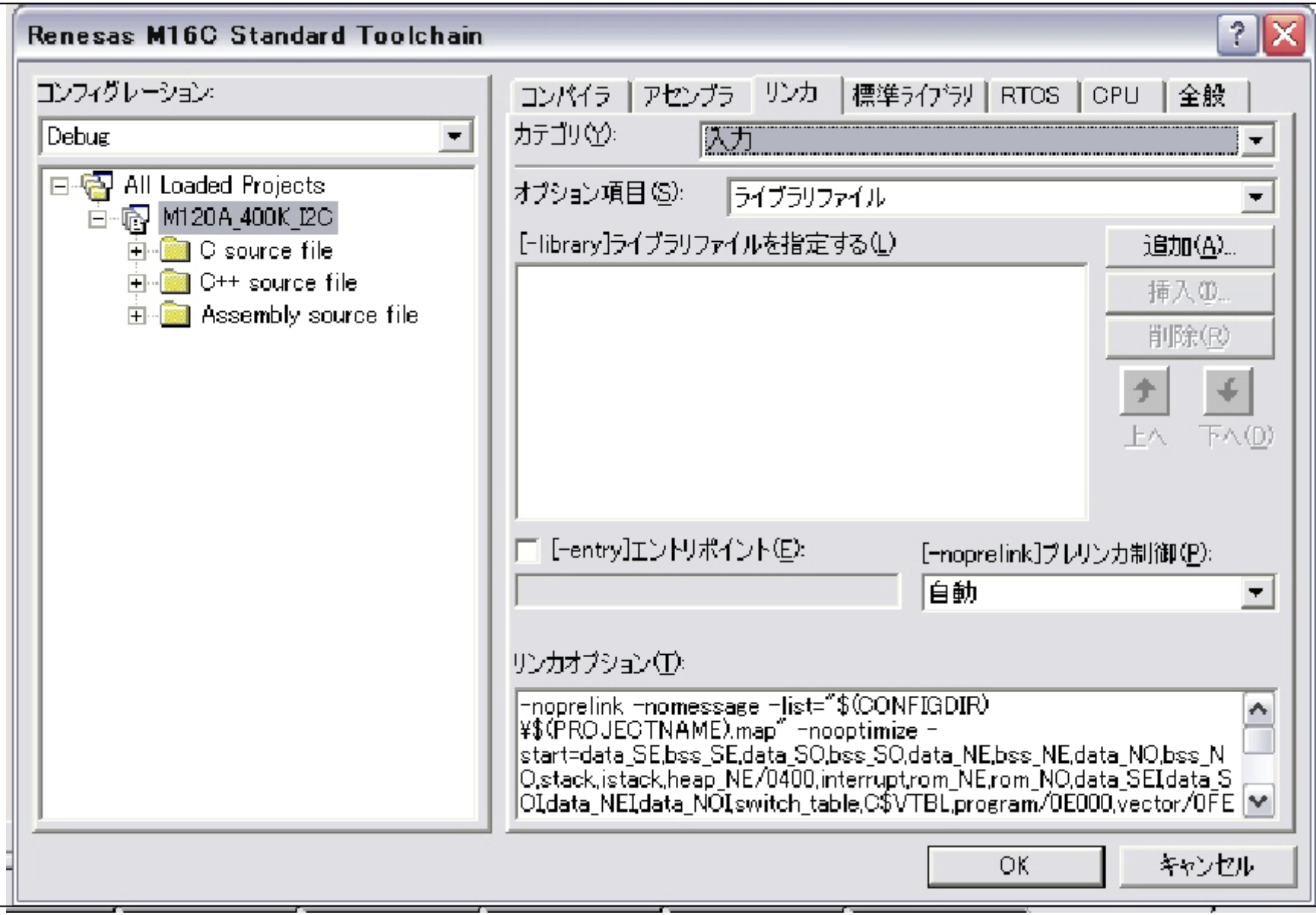

M120A_400K_I2C
M120A_400K_I2C
Assembly source file
dsd_tech_pm1.a30
i2c_tport.a30
ncrt0.a30
R8CM1_IOCS_ADC.a30
R8CM1_IOCS_INIT.a30
R8CM1_IOCS_TIMER.a30
R8CM1_IOCS_UART.a30
C source file
dsd_OLED_sub.c
i2c_packet.c
IOCS_sio_print.c
M120A_400K_I2C.c
string_sub.c
Dependencies
BMF_8x16_ASCII_KANA.inc
cpu_select.h
cpu_select.inc
行番...
S...
ソース
1 //**********
2 //**
3 //** DSD TECH 2 PCS IIC OLED Di
4 //** ドライバルーチン
5 //** ----------
6 //** 2021-08-10 by - Take
7 //**********
8
9 #include "cpu_select.h"
10 #include "R8CM12_IOCS.h"
11
12 #include "dsd_OLED_sub.h"
13 #include "i2c_packet.h"
14 #include "string_sub.h"
15
16
17 void dump_dsd_data(BYTE *ptr
18
19 static BYTE Bufx4[64];
20
21 union Word Byte {

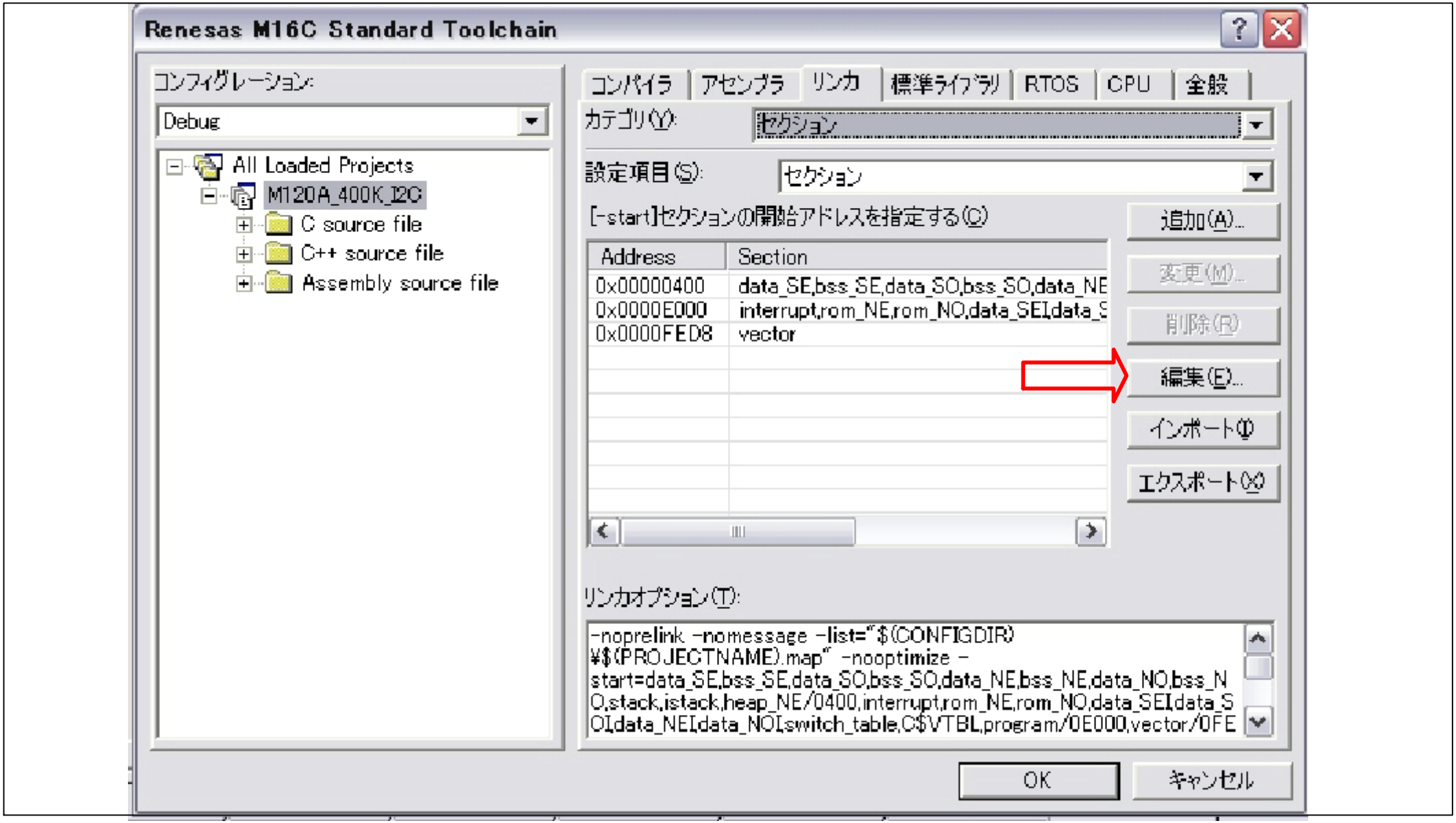

M120A_400K_I2C - High-performance Embedded Workshop - [dsd_OLED_sub.c]
ファイル(F) 編集(E) 表示(V) プロジェクト(P) ビルド(B) デバッグ(D) 基本設定(U) ツール(T) テスト(S) ウィンドウ(W) ヘルプ(H)
8x16
Debug
M120A_400K_I2C
M120A_400K_I2C
Assembly source file
dsd_tech_pm1.a30
i2c_tport.a30
ncrt0.a30
R8CM1_IOCS_ADC.a30
R8CM1_IOCS_INIT.a30
R8CM1_IOCS_TIMER.a30
R8CM1_IOCS_UART.a30
Pr...
Te...
N...
Test
行番...
S...
ソース
1 //***
2 //** **
3 //** DSD TECH 2 PCS IIC OLED Display 0.91 **
4 //** ドライバルーチン **
5 //** ---------------------------------- **
6 //** 2021-08-10 by - Take **
7 //***
8
9 #include "cpu_select.h"
10 #include "R8CM12_IOCS.h"
M120A_...
TypeDe...
R8CM1...
cpu_sel...
cpu_sel...
i2c_pac...
dsd_OLE...
Build Finished
0 Errors, 0 Warnings
Build
Debug
Find in Files 1
Find in Files 2
Macro
Test
Version Control
レディ
Default1 desktop
Read-write
10/317


nc_define.inc
R8CM12_IOCS.h
sect30.inc
str_r8m12a.h
str_r8m12a.inc
string_sub.h
TypeDefine.h
Pr...
Te...
N...
Test
24 };
25
26 struct Adr_Ide {
27 BYTE adr; //
28 BYTE vre; //
29 };
30
M120A_400...
TypeDefine.h
R8CM12_IO...
Maximum link size limited to 64KB code+data.
L3100 (F) Section address overflow out of range : "program"
Optimizing Linkage Editor Abort
Build
Debug
Find in Files 1
Find in Files 2
Macro
Test
Version Control
レディ
スタート
C:¥Documents and S...
C:¥WorkSpace¥YouTu...
E:¥開発環境_説明¥H...


M120A_400K_I2C - High-performance Embedded Workshop - [dsd_OLED_sub.c]
ファイル(F) 編集(E) 表示(V) プロジェクト(P) ビルド(B) デバッグ(D) 基本設定(U) ツール(T) テスト(S) ウィンドウ(W) ヘルプ(H)
Debug
M120A_400K_I2C
M120A_400K_I2C
Assembly source file
dsd_tech_pm1.a30
i2c_tport.a30
ncrt0.a30
R8CM1_IOCS_ADC.a30
R8CM1_IOCS_INIT.a30
R8CM1_IOCS_TIMER.a30
R8CM1_IOCS_UART.a30
C source file
dsd_OLED_sub.c
i2c_packet.c
IOCS_sio_print.c
M120A_400K_I2C.c
string_sub.c
Dependencies
BMF_8x16_ASCII_KANA.ir
cpu_select.h
Renesas M16C Standard Toolchain...
コンパイル(F) Ctrl+F7
ビルド(B) F7
すべてをビルド(A)
複数ビルド(M)...
クリーン アクティブプロジェクト(L)
クリーン 全プロジェクト(E)
すべての依存関係を更新(U)
ツールの中止(S) Ctrl+Break
ビルドから除外/除外の解除(I)
ビルドフェーズ(P)...
ビルドの構成(C)...
リンク順の指定(K)...
Makeファイルの生成(G)...
OLED Display 0.91
Take
YTE *ptr); // 送信
]; // 縦横

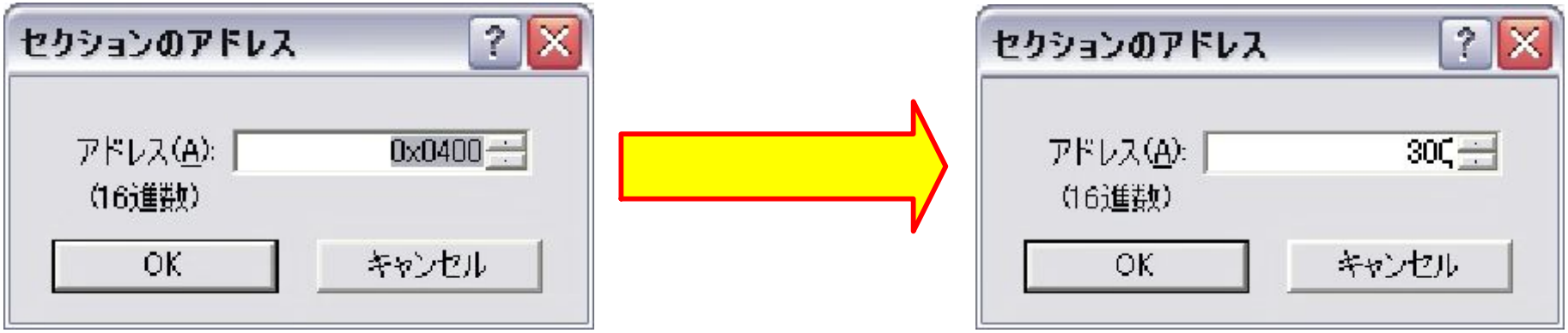

Renesas M16C Standard Toolchain
コンフィグレーション:
Debug
All Loaded Projects
M120A_400K_I2C
C source file
C++ source file
Assembly source file
コンパイラ
アセンブラ
リンカ
標準ライブラリ
RTOS
CPU
全般
カテゴリ(Y):
ソース
オプション項目(S):
ソースファイル
[-lang]言語(L):
C:
C
C++ (P):
C++
コンパイラオプション(T):
-D_UART0_ -c -finfo -dir "$(CONFIGDIR)" -silent -R8C
OK
キャンセル


Renesas M16C Standard Toolchain
コンフィグレーション:
Debug
All Loaded Projects
M120A_400K_I2C
C source file
C++ source file
Assembly source file
コンパイラ
アセンブラ
リンカ
標準ライブラリ
RTOS
CPU
全般
カテゴリ(Y):
入力
オプション項目(S):
ライブラリファイル
[-library]ライブラリファイルを指定する(L)
追加(A)...
挿入(I)...
削除(R)
上へ
下へ(D)
[-entry]エントリポイント(E):
[-noprelink]プレリンカ制御(P):
自動
リンカオプション(T):
-noprelink -nomessage -list="$(CONFIGDIR)
¥$(PROJECTNAME).map" -nooptimize -
start=data_SE,bss_SE,data_SO,bss_SO,data_NE,bss_NE,data_NO,bss_N
O,stack,istack,heap_NE/0400,interrupt,rom_NE,rom_NO,data_SEI,data_S
OI,data_NEI,data_NOI,switch_table,C$VTBL,program/0E000,vector/0FE
OK
キャンセル

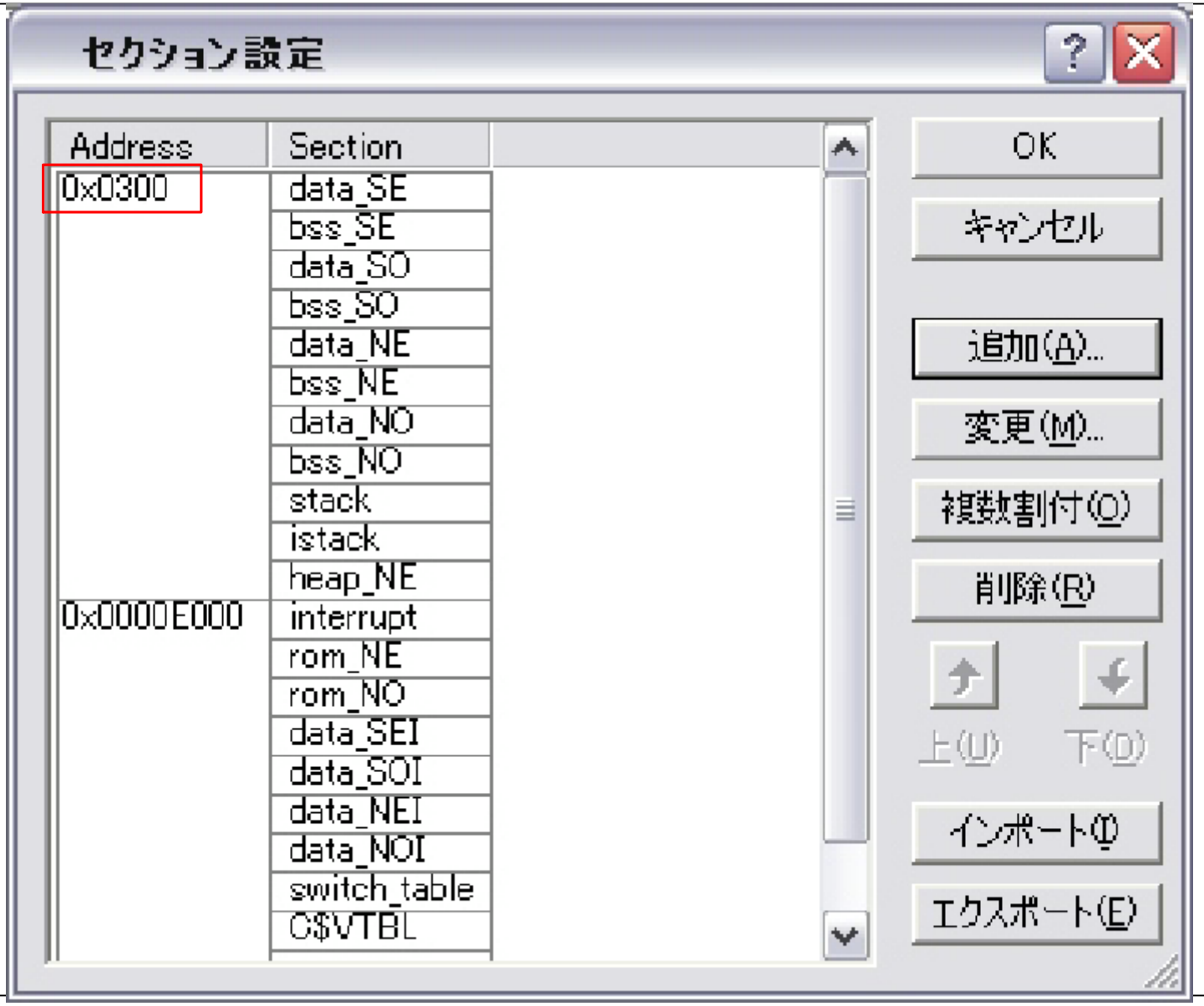

Renesas M16C Standard Toolchain
コンフィグレーション:
Debug
All Loaded Projects
M120A_400K_I2C
C source file
C++ source file
Assembly source file
コンパイラ
アセンブラ
リンカ
標準ライブラリ
RTOS
CPU
全般
カテゴリ(Y):
セクション
設定項目(S):
セクション
[-start]セクションの開始アドレスを指定する(C)
追加(A)...
変更(M)...
削除(R)
編集(E)...
インポート(I)
エクスポート(X)
Address
Section
0x00000400
data_SE,bss_SE,data_SO,bss_SO,data_NE
0x0000E000
interrupt,rom_NE,rom_NO,data_SEI,data_S
0x0000FED8
vector
リンカオプション(T):
-noprelink -nomessage -list="$(CONFIGDIR)
¥$(PROJECTNAME).map" -nooptimize -
start=data_SE,bss_SE,data_SO,bss_SO,data_NE,bss_NE,data_NO,bss_N
O,stack,istack,heap_NE/0400,interrupt,rom_NE,rom_NO,data_SEI,data_S
OI,data_NEI,data_NOI,switch_table,C$VTBL,program/0E000,vector/0FE
OK
キャンセル


セクション設定
Address
Section
0x00000400
data_SE
bss_SE
data_SO
bss_SO
data_NE
bss_NE
data_NO
bss_NO
stack
istack
heap_NE
0x0000E000
interrupt
rom_NE
rom_NO
data_SEI
data_SOI
data_NEI
data_NOI
switch_table
C$VTBL
OK
キャンセル
追加(A)...
変更(M)...
複数割付(O)
削除(R)
上(U)
下(D)
インポート(I)
エクスポート(E)

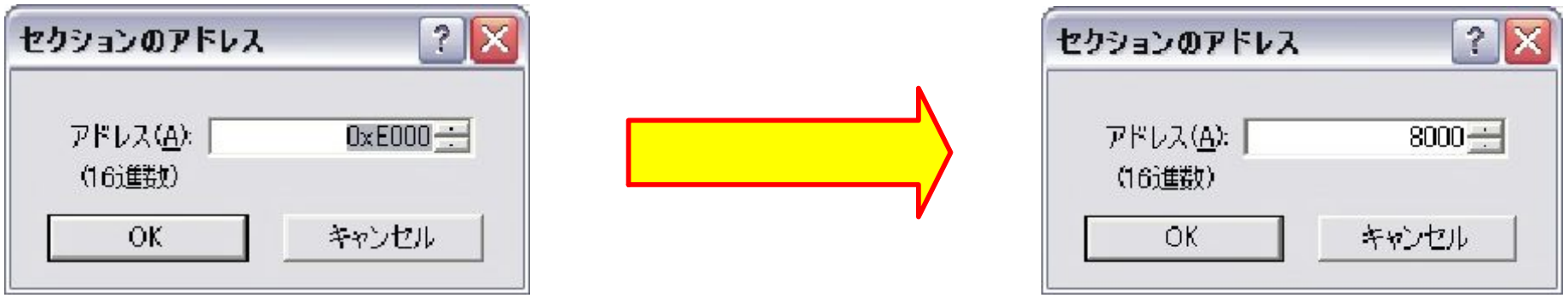

セクション設定
Address
Section
0x00000400
data_SE
bss_SE
data_SO
bss_SO
data_NE
bss_NE
data_NO
bss_NO
stack
istack
heap_NE
0x0000E000
interrupt
rom_NE
rom_NO
data_SEI
data_SOI
data_NEI
data_NOI
switch_table
C$VTBL
1
2
OK
キャンセル
追加(A)...
変更(M)...
複数割付(O)
削除(R)
上(U)
下(D)
インポート(I)
エクスポート(E)



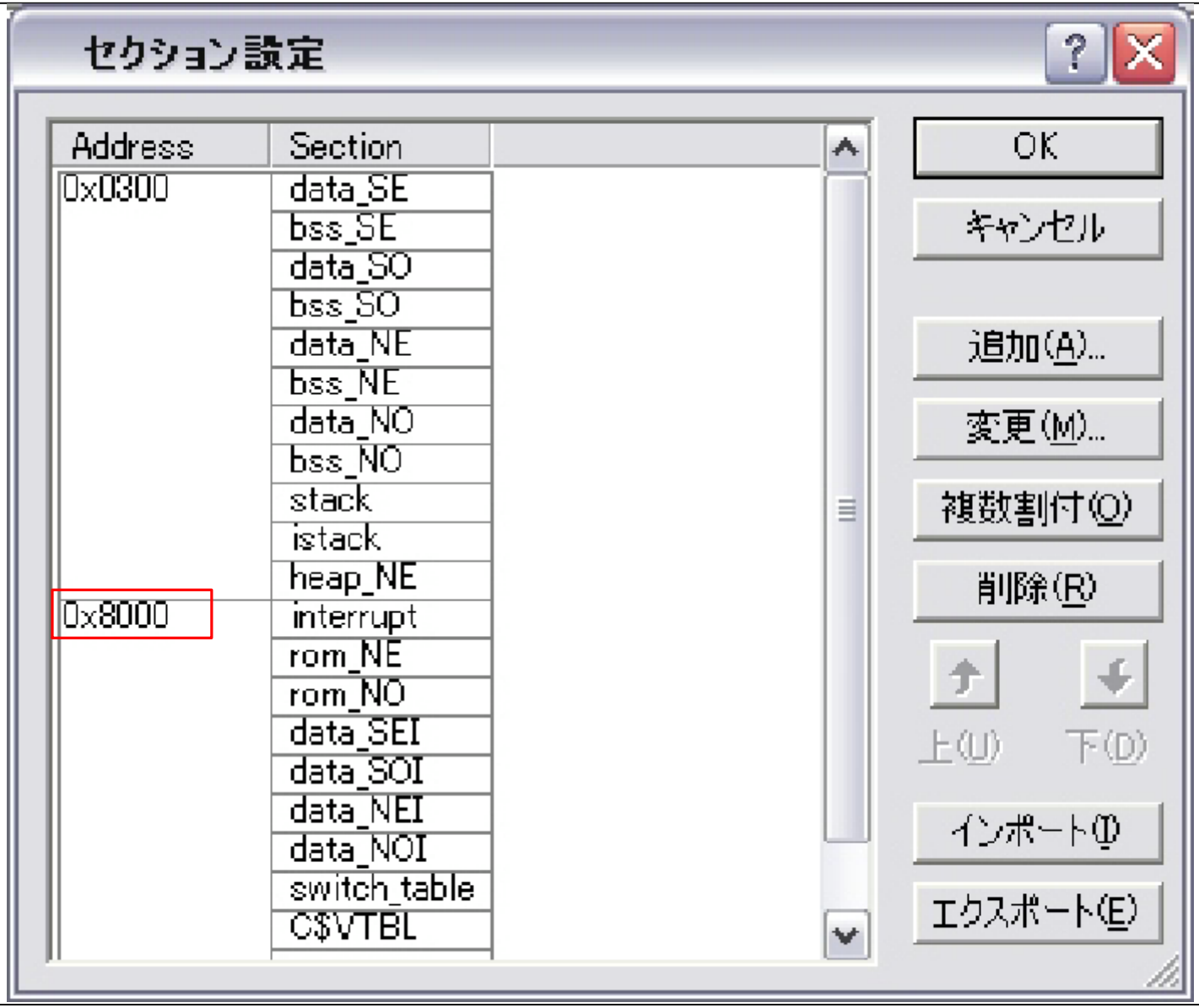
RAMの 先頭アドレス：400 を　300 に変えて
［ OK ］をクリックする。

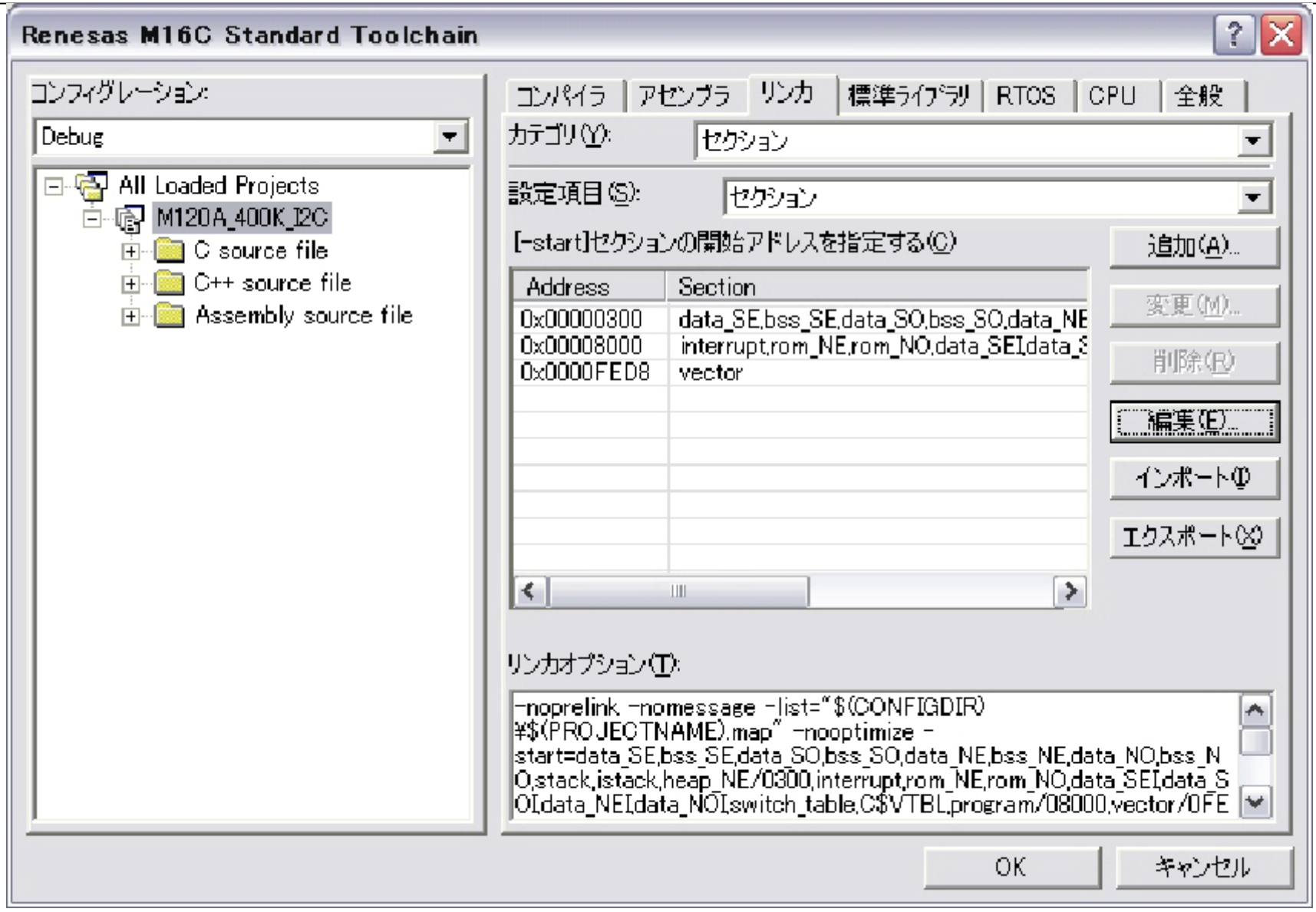

セクション設定
Address
Section
0x0300
data_SE
bss_SE
data_SO
bss_SO
data_NE
bss_NE
data_NO
bss_NO
stack
istack
heap_NE
0x0000E000
interrupt
rom_NE
rom_NO
data_SEI
data_SOI
data_NEI
data_NOI
switch_table
C$VTBL
OK
キャンセル
追加(A)...
変更(M)...
複数割付(O)
削除(R)
上(U)
下(D)
インポート(I)
エクスポート(E)

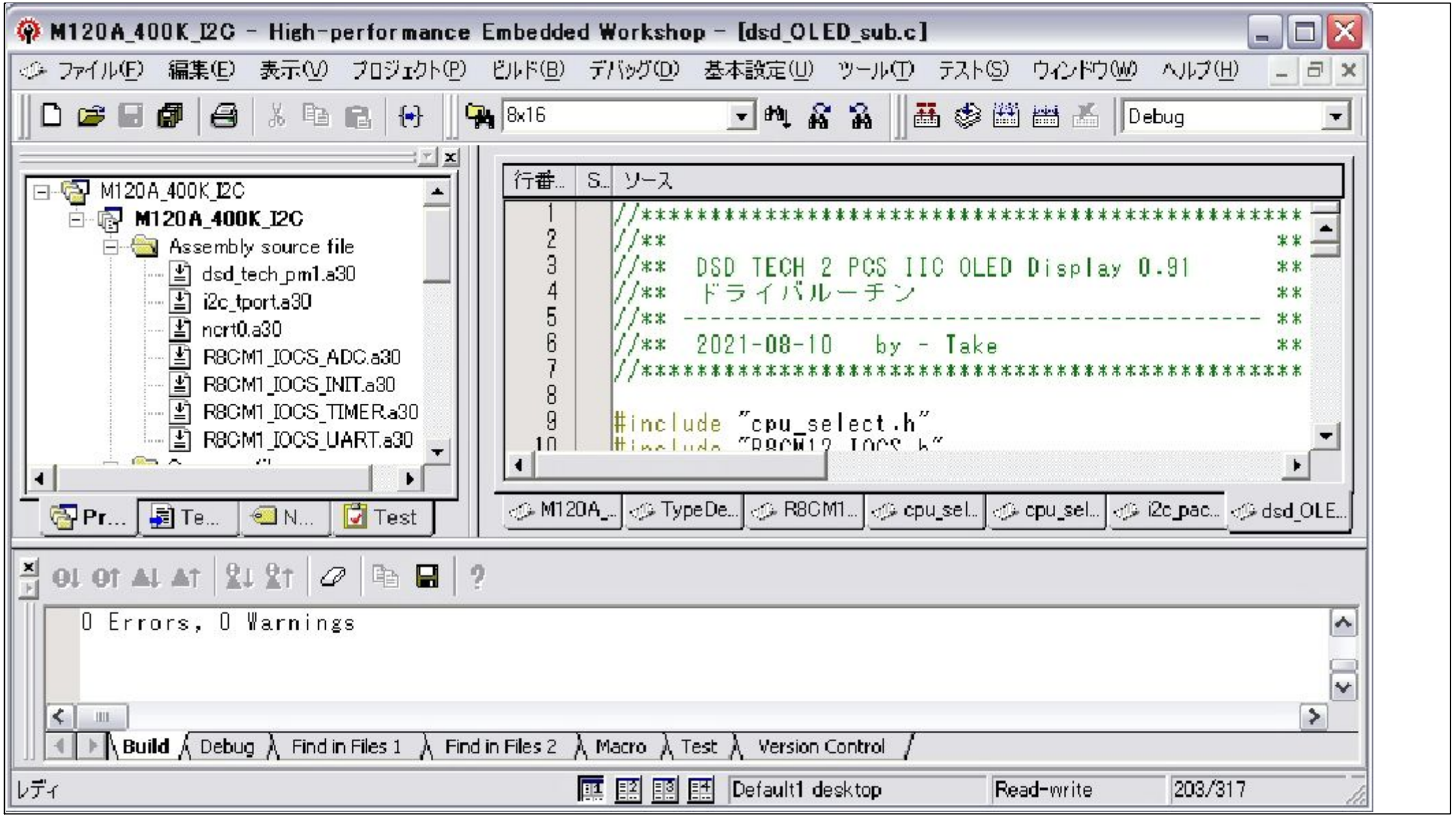

セクション設定
Address
Section
0x0300
data_SE
bss_SE
data_SO
bss_SO
data_NE
bss_NE
data_NO
bss_NO
stack
istack
heap_NE
0x0000E000
interrupt
rom_NE
rom_NO
data_SEI
data_SOI
data_NEI
data_NOI
switch_table
C$VTBL
OK
キャンセル
追加(A)...
変更(M)...
複数割付(O)
削除(R)
上(U)
下(D)
インポート(I)
エクスポート(E)
1
2

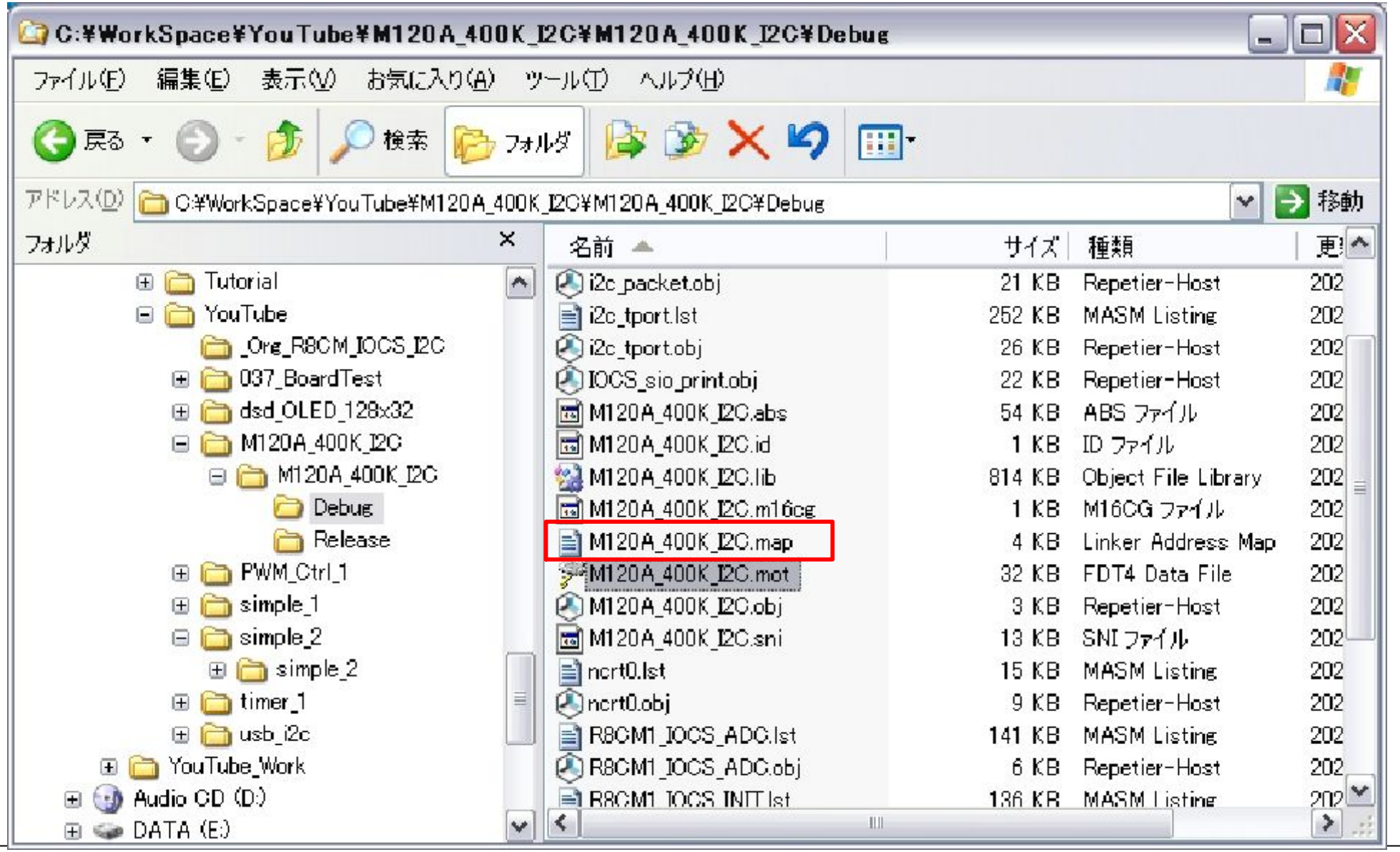


ROMの 先頭アドレス：E000 を　8000 に変えて
［ OK ］をクリックする。


セクション設定
Address
Section
0x0300
data_SE
bss_SE
data_SO
bss_SO
data_NE
bss_NE
data_NO
bss_NO
stack
istack
heap_NE
0x8000
interrupt
rom_NE
rom_NO
data_SEI
data_SOI
data_NEI
data_NOI
switch_table
C$VTBL
OK
キャンセル
追加(A)...
変更(M)...
複数割付(O)
削除(R)
上(U)
下(D)
インポート(I)
エクスポート(E)


Renesas M16C Standard Toolchain
コンフィグレーション:
Debug
All Loaded Projects
M120A_400K_I2C
C source file
C++ source file
Assembly source file
コンパイラ
アセンブラ
リンカ
標準ライブラリ
RTOS
CPU
全般
カテゴリ(Y):
セクション
設定項目(S):
セクション
[-start]セクションの開始アドレスを指定する(C)
追加(A)...
変更(M)...
削除(R)
編集(E)...
インポート(I)
エクスポート(X)
Address
Section
0x00000300
data_SE,bss_SE,data_SO,bss_SO,data_NE
0x00008000
interrupt,rom_NE,rom_NO,data_SEI,data_S
0x0000FED8
vector
リンカオプション(T):
-noprelink -nomessage -list="$(CONFIGDIR)
¥$(PROJECTNAME).map" -nooptimize -
start=data_SE,bss_SE,data_SO,bss_SO,data_NE,bss_NE,data_NO,bss_N
O,stack,istack,heap_NE/0300,interrupt,rom_NE,rom_NO,data_SEI,data_S
OI,data_NEI,data_NOI,switch_table,C$VTBL,program/08000,vector/0FE
OK
キャンセル


M120A_400K_I2C - High-performance Embedded Workshop - [dsd_OLED_sub.c]
ファイル(F) 編集(E) 表示(V) プロジェクト(P) ビルド(B) デバッグ(D) 基本設定(U) ツール(T) テスト(S) ウィンドウ(W) ヘルプ(H)
8x16
Debug
M120A_400K_I2C
M120A_400K_I2C
Assembly source file
dsd_tech_pm1.a30
i2c_tport.a30
ncrt0.a30
R8CM1_IOCS_ADC.a30
R8CM1_IOCS_INIT.a30
R8CM1_IOCS_TIMER.a30
R8CM1_IOCS_UART.a30
Pr...
Te...
N...
Test
行番...
S...
ソース
1 //***
2 //** **
3 //** DSD TECH 2 PCS IIC OLED Display 0.91 **
4 //** ドライバルーチン **
5 //** ---------------------------------- **
6 //** 2021-08-10 by - Take **
7 //***
8
9 #include "cpu_select.h"
10 #include "R8CM12_IOCS.h"
M120A_...
TypeDe...
R8CM1...
cpu_sel...
cpu_sel...
i2c_pac...
dsd_OLE...
0 Errors, 0 Warnings
Build
Debug
Find in Files 1
Find in Files 2
Macro
Test
Version Control
レディ
Default1 desktop
Read-write
203/317


C:¥WorkSpace¥YouTube¥M120A_400K_I2C¥M120A_400K_I2C¥Debug
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る
検索
フォルダ
アドレス(D) C:¥WorkSpace¥YouTube¥M120A_400K_I2C¥M120A_400K_I2C¥Debug
移動
フォルダ
Tutorial
YouTube
_Org_R8CM_IOCS_I2C
037_BoardTest
dsd_OLED_128x32
M120A_400K_I2C
M120A_400K_I2C
Debug
Release
PWM_Ctrl_1
simple_1
simple_2
simple_2
timer_1
usb_i2c
YouTube_Work
Audio CD (D:)
DATA (E:)
名前
サイズ
種類
i2c_packet.obj 21 KB Repetier-Host
i2c_tport.lst 252 KB MASM Listing
i2c_tport.obj 26 KB Repetier-Host
IOCS_sio_print.obj 22 KB Repetier-Host
M120A_400K_I2C.abs 54 KB ABS ファイル
M120A_400K_I2C.id 1 KB ID ファイル
M120A_400K_I2C.lib 814 KB Object File Library
M120A_400K_I2C.m16cg 1 KB M16CG ファイル
M120A_400K_I2C.map 4 KB Linker Address Map
M120A_400K_I2C.mot 32 KB FDT4 Data File
M120A_400K_I2C.obj 3 KB Repetier-Host
M120A_400K_I2C.sni 13 KB SNI ファイル
ncrt0.lst 15 KB MASM Listing
ncrt0.obj 9 KB Repetier-Host
R8CM1_IOCS_ADC.lst 141 KB MASM Listing
R8CM1_IOCS_ADC.obj 6 KB Repetier-Host

M120A_400K_I2C.map 内容の一部分

```
data_NEI
                    000080ba   000080c5          c    2
data_NOI
                    000080c6   000080c6          0    1
program
                    000080c6   0000ad3f       2c7a    2
                                              11,386 byte
vector
                    0000fed8   0000ffd7        100    2
fvector
                    0000ffd8   0000ffff         28    1
```

I2C ソフトを、bit転送速度 400Kbpsで、R8C/M120Aに対応させた結果

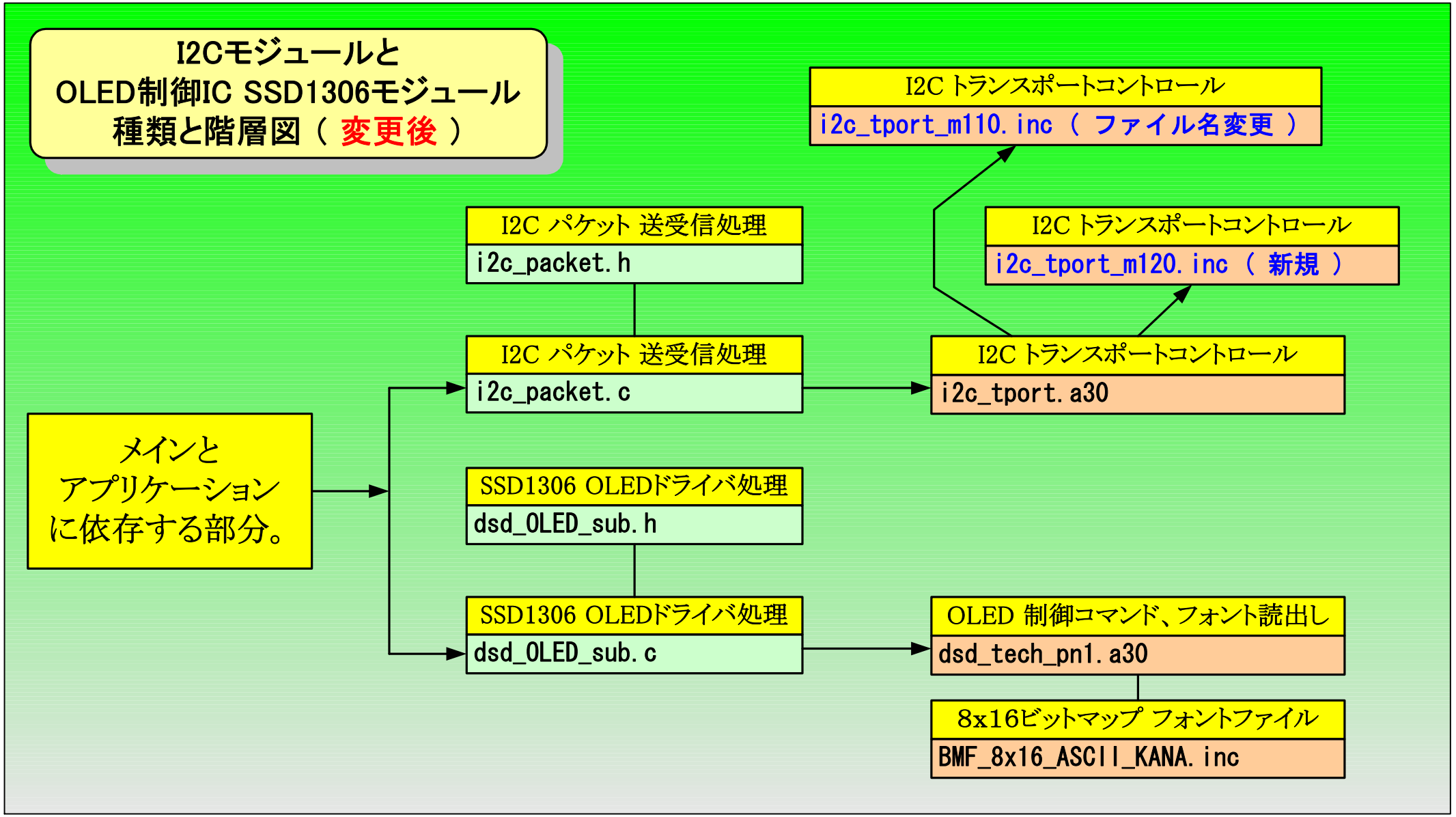
最終的に、うまく行きましたが、だいぶ難儀しました。 ソフト I2C処理 下位層の i2c_tport.inc と i2c_tport.a30 の改修を行いました。

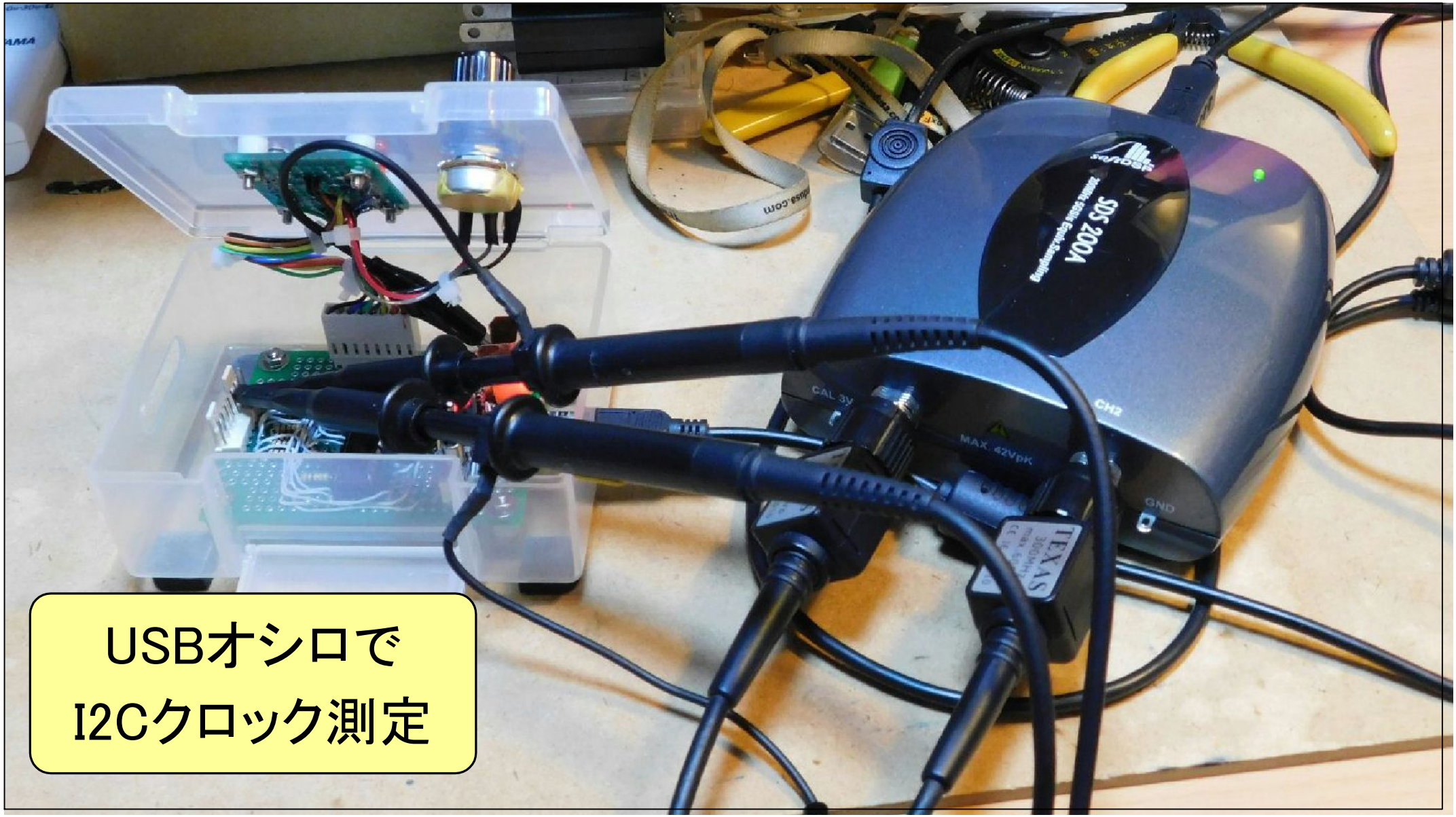
当初、R8C/M110A専用に作成していた I2c_tport.inc を アセンブル制御 .IF ～ .ELSE ～ .ENDIF で R8C/M110A と R8C/M120A の コーディングを 切り分けていましたが、変更箇所が、13箇所発生し、不可解な現象にぶつかりしばらく悩んでいました。

いろいろ考えた末に、i2c_tport.inc を i2c_tport_m110.inc と i2c_tport_m120.inc の2つにファイルを分ける事にしました。
i2c_tport_m120.a30 先頭で、インクルードファイルを、取りこむところで、アセンブル制御で、上記、.inc ファイルの取り込みを スイッチして 取り込みする事にしました。

という事で、i2c_tport.a30 は、先頭の部分変更で上位層から呼び出される関数仕様は、変更前と同じです。
i2c_tport.inc(M110A専用) は i2c_tport_m110.inc に ファイル名を変更し、新たに i2c_tport_m120.inc(M120専用)を 作成しました。

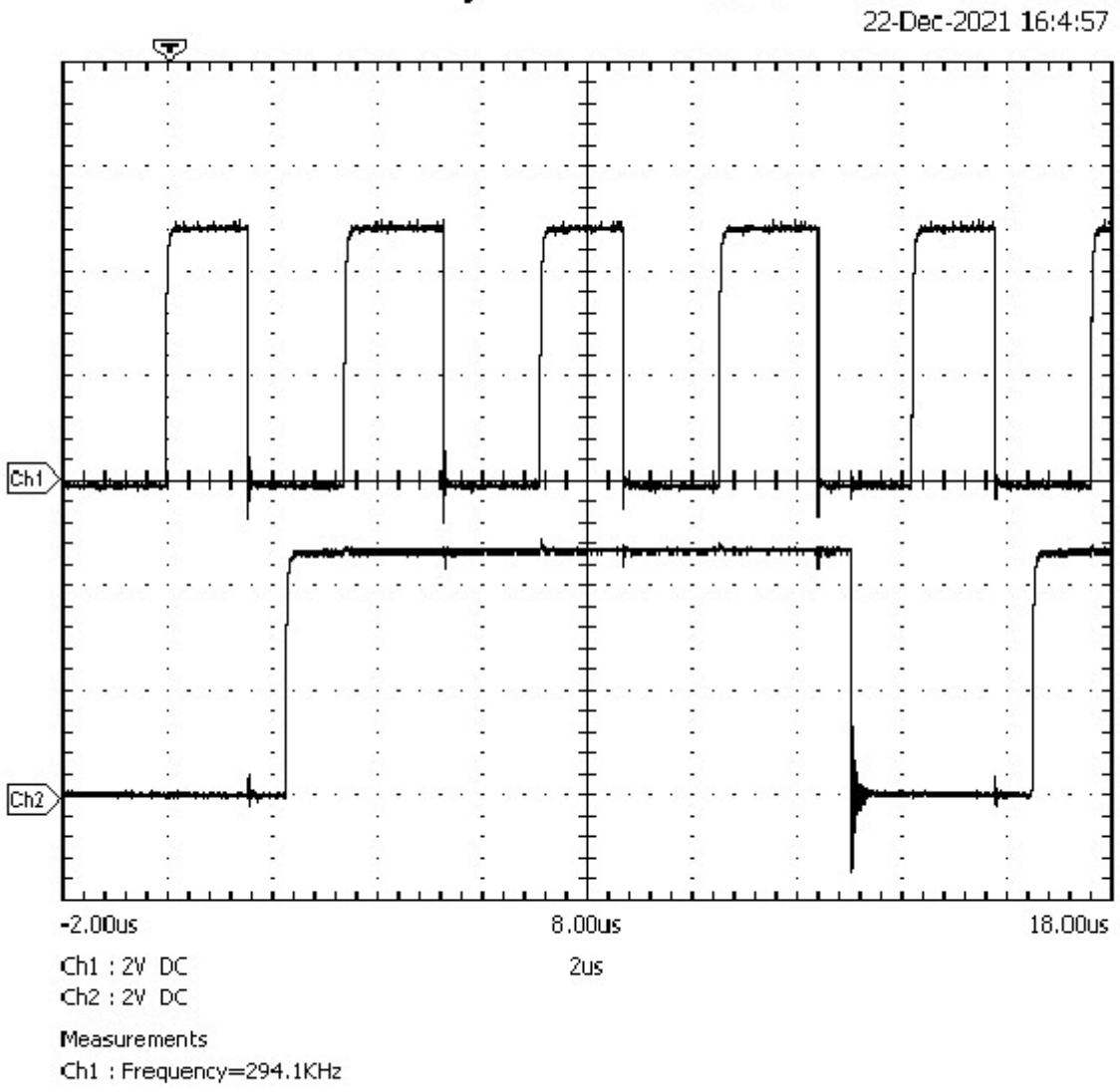
上位層の互換性を維持していたので、OLEDのアクセス処理は、以前の M110Aで動いていたソフトを、そのまま持ってきて、あっさり動きました。


I2Cモジュールと
OLED制御IC SSD1306モジュール
種類と階層図（ 変更後 ）
I2C トランスポートコントロール
i2c_tport_m110.inc （ ファイル名変更 ）
I2C パケット 送受信処理
i2c_packet.h
I2C トランスポートコントロール
i2c_tport_m120.inc （ 新規 ）
I2C パケット 送受信処理
i2c_packet.c
I2C トランスポートコントロール
i2c_tport.a30
メインと
アプリケーション
に依存する部分。
SSD1306 OLEDドライバ処理
dsd_OLED_sub.h
SSD1306 OLEDドライバ処理
dsd_OLED_sub.c
OLED 制御コマンド、フォント読出し
dsd_tech_pn1.a30
8x16ビットマップ フォントファイル
BMF_8x16_ASCII_KANA.inc



USBオシロで
I2Cクロック測定

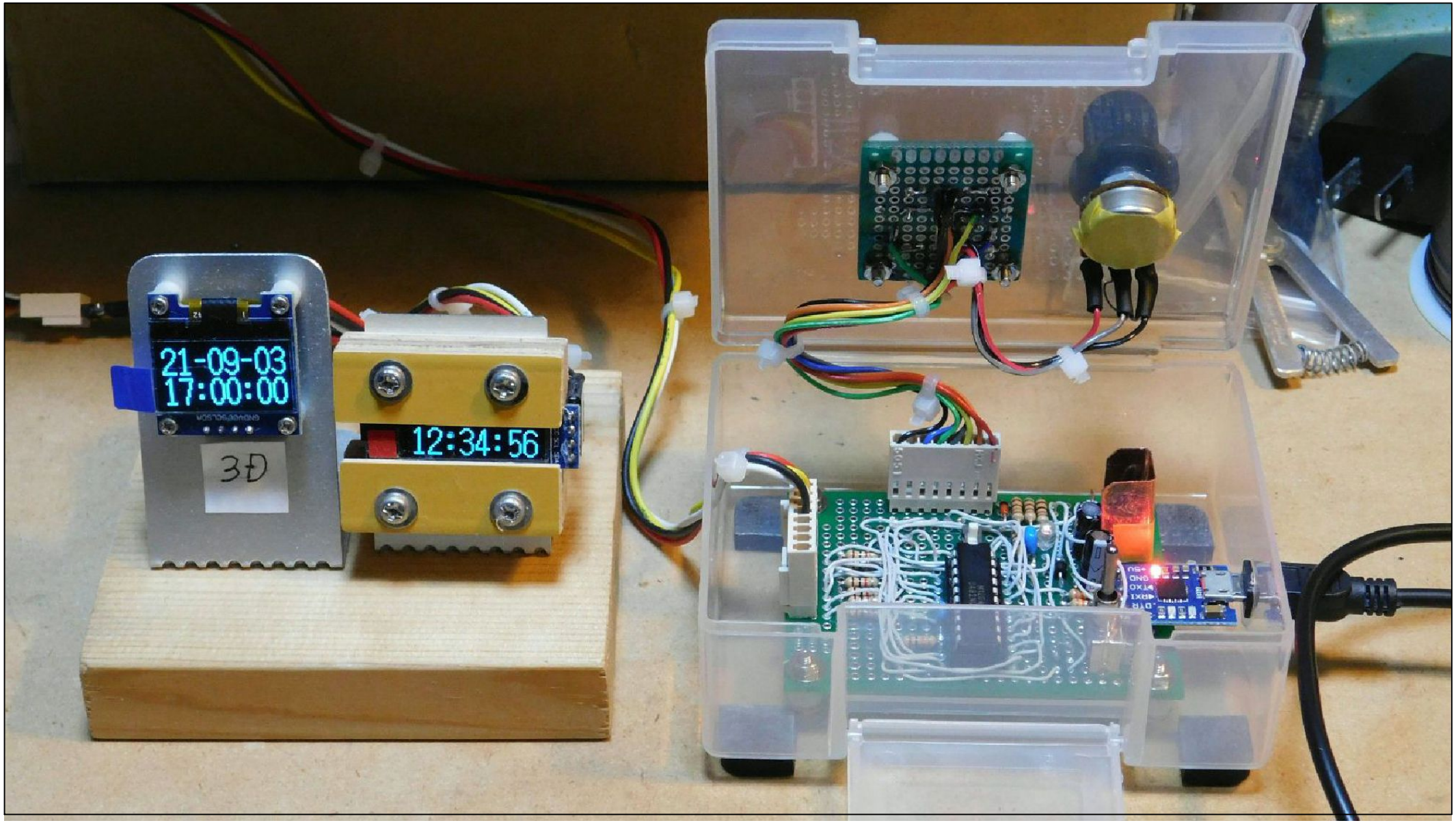
R8C/M120 I2C 信号



調整前

294.1KHz


CH1
CH2
Math
x10
2V
x10
2V
x1
1V
x1
1V
Measurement
Ch1 : Frequency=403.2KHz
Delete
Delete All
Save
Next
2us
500MS/s
10%
CH1 NORMAL Edge 2.32V
for DSTM12
for DSTM12
Check Pr...
PICkit3
Renesas
調整後
403.2KHz


21-09-03
17:00:00
12:34:56
3D