

CMOS 32-BIT SINGLE CHIP MICROCONTROLLER

S1C31 Family
ソフトウェア開発
セットアップガイド

ARM[®]

評価ボード・キット、開発ツールご使用上の注意事項

1. 本評価ボード・キット、開発ツールは、お客様での技術的評価、動作の確認および開発のみに用いられることを想定し設計されています。それらの技術評価・開発等の目的以外には使用しないで下さい。本品は、完成品に対する設計品質に適合していません。
2. 本評価ボード・キット、開発ツールは、電子エンジニア向けであり、消費者向け製品ではありません。お客様において、適切な使用と安全に配慮願います。弊社は、本品を用いることで発生する損害や火災に対し、いかなる責も負いかねます。通常の使用においても、異常がある場合は使用を中止して下さい。
3. 本評価ボード・キット、開発ツールに用いられる部品は、予告無く変更されることがあります。

本資料のご使用につきましては、次の点にご留意願います。

本資料の内容については、予告無く変更することがあります。

1. 本資料の一部、または全部を弊社に無断で転載、または、複製など他の目的に使用することは堅くお断りいたします。
2. 本資料に掲載される応用回路、プログラム、使用方法等はあくまでも参考情報であり、これらに起因する第三者の知的財産権およびその他の権利侵害あるいは損害の発生に対し、弊社はいかなる保証を行うものではありません。また、本資料によって第三者または弊社の知的財産権およびその他の権利の実施権の許諾を行うものではありません。
3. 特性値の数値の大小は、数直線上の大小関係で表しています。
4. 製品および弊社が提供する技術を輸出等するにあたっては「外国為替および外国貿易法」を遵守し、当該法令の定める手続きが必要です。大量破壊兵器の開発等およびその他の軍事用途に使用する目的をもって製品および弊社が提供する技術を費消、再販売または輸出等しないでください。
5. 本資料に掲載されている製品は、生命維持装置その他、きわめて高い信頼性が要求される用途を前提としていません。よって、弊社は本（当該）製品をこれらの用途に用いた場合のいかなる責任についても負いかねます。
6. 本資料に掲載されている会社名、商品名は、各社の商標または登録商標です。
ARM および Cortex は、ARM Limited(またはその子会社)の EU またはその他の国における登録商標です。IAR Systems、IAR Embedded Workbench、C-SPY、I-jet、IAR および IAR システムズのロゴタイプは、IAR Systems AB が所有権を有する商標または登録商標です。All rights reserved.
"Reproduced with permission from ARM Limited. Copyright © ARM Limited"

目 次

1. はじめに	1
2. 開発環境の入手	2
2.1 入手方法	2
2.2 参照ドキュメント	3
3. 開発環境のセットアップ	4
3.1 IDE のインストール	4
3.1.1 IAR EWARM を使用する場合	4
3.1.2 MDK-ARM を使用する場合	5
3.2 ハードウェアの接続	6
3.2.1 I-jet を使用する場合	6
3.2.2 J-Link を使用する場合	7
4. ソフトウェアの開発・評価	9
4.1 サンプルソフトウェアのインストール	9
4.2 ソフトウェアのビルドおよびデバッグ	11
4.2.1 IAR EWARM を使用する場合	11
4.2.2 MDK-ARM を使用する場合	15
改訂履歴表	24

1. はじめに

本ドキュメントは、初めて S1C31 Family^{*1} を使用されるお客様が、速やかにソフトウェア開発および評価を開始できるように、開発環境のセットアップからソフトウェアの開発・評価までのフローについて記載しています。

^{*1}: S1C31 Family は ARM[®] Cortex[®]-M0+プロセッサを搭載したマイクロコントローラでセイコーエプソン株式会社の製品です。

表 1.1 に、S1C31 Family の全体のソフトウェアの開発ステップと各開発ステップの参照ドキュメントを示します。

表 1.1 ソフトウェア開発ステップと参照ドキュメント

開発ステップ	弊社提供の参照ドキュメント ^{*2}
開発環境の入手	・ 本ドキュメント (2 章)
開発環境のセットアップ ・ IDE のインストール ・ ハードウェアの接続	・ 本ドキュメント (3 章) ・ S5U1C31xxxTx マニュアル ・ S1C31 xxx 周辺回路サンプルソフトウェアマニュアル
ソフトウェアの開発・評価 ・ サンプルソフトウェアのセットアップ ・ ソフトウェアのビルド・デバッグ	・ 本ドキュメント (4 章) ・ S5U1C31xxxTx マニュアル ・ S1C31 xxx 周辺回路サンプルソフトウェアマニュアル ・ S1C31xxx テクニカルマニュアル

^{*2}: 弊社マイクロコントローラ Web サイトからダウンロードできます。

2. 開発環境の入手

2. 開発環境の入手

本章では、ソフトウェアの開発・評価で必要となる環境（ソフトウェアおよびハードウェア）について説明します。

2.1 入手方法

S1C31 Family のソフトウェアの開発・評価には、市販の ARM ソフトウェア開発ツール（統合開発環境 (IDE)、デバッグプロンプ）を使用します。

表 2.1.1 に、弊社推奨の開発ツールを示します。これらのツールは、お客様ご自身で各ベンダーまたは各ベンダー販売代理店に問い合わせて入手してください。

表 2.1.1 お客様ご自身でご用意いただくもの

種別	名称	備考
PC	Windows 7/10 搭載 PC	-
統合開発環境 (IDE)	IAR Embedded Workbench for ARM (IAR Systems 社製)	どちらか一方をご用意ください
	MDK-ARM (ARM 社製)	
デバッグプロンプ	I-jet (IAR Systems 社製)	どちらか一方をご用意ください
	J-Link (SEGGER 社製)	

表 2.1.1 に記載される IDE とデバッグプロンプは、どちらか一方をご用意ください。ただし、表 2.1.2 に示すように、IDE とデバッグプロンプには、使用できない組合せがありますのでご注意ください。

表 2.1.2 IDE とデバッグプロンプの使用可能な組合せ

	I-jet	J-Link
IAR Embedded Workbench for ARM	使用可	使用可
MDK-ARM	使用不可	使用可

更に、弊社より提供するサンプルソフトウェアや評価ボードを使用することで、速やかにソフトウェアの開発・評価を始めることができます。

表 2.1.3 に、弊社より提供する開発ツールを示します。

表 2.1.3 弊社よりご提供させていただくもの

種別	名称	備考
サンプルソフトウェア	S1C31xxx 周辺回路サンプルソフトウェア*1	-
評価ボード	S5U1C31xxxTx (S1C31xxx 評価ボード) *2	-

*1: 弊社マイクロコントローラ Web サイトからダウンロードできます。

*2: 弊社半導体製品営業窓口にお問い合わせください。

2.2 参照ドキュメント

各開発ツールの詳細情報については、表 2.2.1 に示すのドキュメントをご参照ください。

表 2.2.1 参照ドキュメント

名称	参照情報	参照ドキュメント
IAR Embedded Workbench for ARM	インストール	・ クイックリファレンス*1
	IDE 操作	・ IAR Embedded Workbench 導入ガイド*1 ・ IDE プロジェクト管理およびビルドガイド*1 ・ C-SPY® デバッグガイド*1
MDK-ARM	インストール	・ Getting Started with MDK*2
	IDE 操作	・ Getting Started with MDK*2 ・ Complete User's Guide Selection*2
I-jet	-	・ IAR デバッグプローブガイド*1
J-Link	-	・ J-Link/J-Trace User Guide*3
S1C31xxx 周辺回路 サンプルソフトウェア	-	・ S1C31xxx 周辺回路サンプルソフトウェアマニュアル*4
S1C31xxx 評価ボード	-	・ S5U1C31xxxTx マニュアル*4

*1: IAR Systems 社 Web サイトからダウンロードできます。

Web サイトの URL は、< <https://www.iar.com/jp/>>です。

*2: ARM 社の ARM KEIL Web サイトからダウンロードできます。

Web サイトの URL は、< <http://www.keil.com/>>です。

*3: SEGGER 社 Web サイトからダウンロードできます。

Web サイトの URL は、< <https://www.segger.com/>>です。

*4: 弊社マイクロコントローラ Web サイトからダウンロードできます。

3. 開発環境のセットアップ

3. 開発環境のセットアップ

本章では、S1C31 Family 用ソフトウェア開発環境のセットアップ手順について説明します。

3.1 IDE のインストール

本節では、IAR EWARM および MDK-ARM のインストール手順について説明します。

3.1.1 IAR EWARM を使用する場合

IAR Embedded Workbench for ARM のインストールは以下の手順で行います。

- (1) IAR システムズ社の Web サイトより、IAR Embedded Workbench for ARM のインストーラを入手します。
- (2) Windows 上で動作している全てのアプリケーションを終了します。
- (3) インストーラをダブルクリックして起動します。起動すると、インストールウィザードが表示されます。(図 3.1.1.1)
- (4) インストールウィザードの[IAR Embedded Workbench®ARM 用インストール]をクリックします。(図 3.1.1.1)

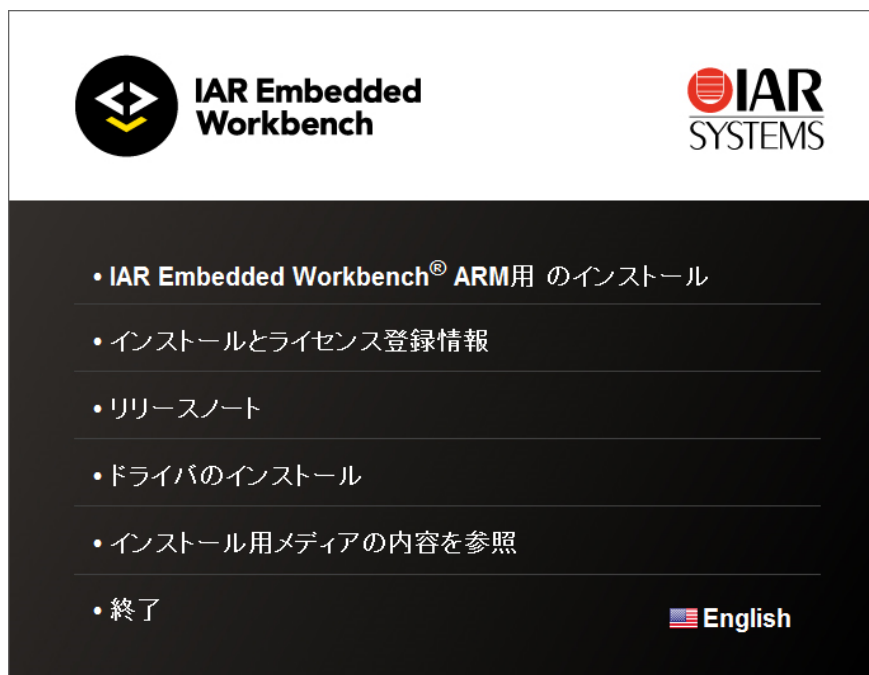


図 3.1.1.1 起動直後の IAR Embedded Workbench インストールウィザード

- (5) インストールウィザードの指示に従ってインストールを行います。推奨する設定は以下のとおりです。
 - ・インストール先： デフォルトのフォルダ
 - ・インストールオプション： 完全インストール

インストール後、初めて IAR Embedded Workbench を起動する場合、ライセンス登録が要求されます。ライセンス登録は、お客様が購入したライセンスの種類に応じて登録処理が異なります。ライセンス登録の手順については、IAR システムズ社が提供する“クイックリファレンス”(表 2.2.1 記載)を参照ください。

3.1.2 MDK-ARM を使用する場合

MDK-ARM のインストールは以下の手順で行います。

- (1) ARM 社の Web サイトより、MDK-ARM のインストーラを入手します。
- (2) Windows 上で動作している全てのアプリケーションを終了します。
- (3) インストーラをダブルクリックして起動します。起動すると、インストールウィザード画面が表示されます。(図 3.1.2.1)
- (4) インストールウィザード画面の[Next>>]ボタンをクリックします。(図 3.1.2.1)

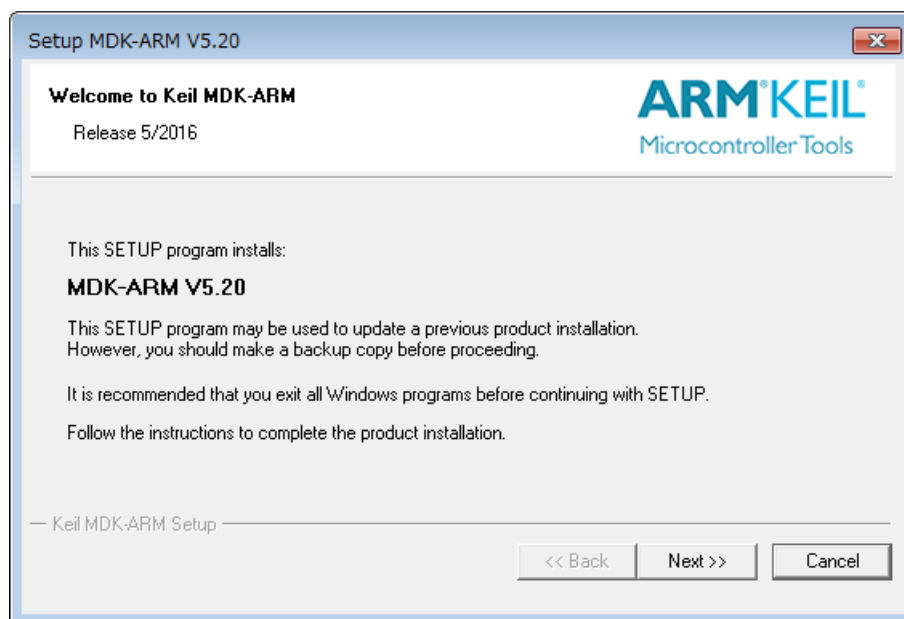


図 3.1.2.1 起動直後の MDK-ARM インストールウィザード画面

- (5) インストールウィザード画面の指示に従って MDK コアのインストールを行います。推奨する設定は以下のとおりです。
 - ・インストール先の選択： デフォルトのフォルダ
- (6) MDK コアのインストール終了後、Pack Installer が自動的に起動しますので、cortex-M0 plus のソフトウェアパックを選択します。

インストール終了後、コードサイズ制限なしで MDK-ARM を使用する場合は、MDK-ARM のライセンス登録が必要です。ライセンス登録は、お客様が購入したライセンスの種類に応じて登録処理が異なります。ライセンス登録の手順については、ARM 社が提供する“Getting Started with MDK”（表 2.2.1 記載）を参照ください。

3. 開発環境のセットアップ

3.2 ハードウェアの接続

本節では、ソフトウェア開発環境のハードウェア接続について説明します。S1C31 Family では、デバッグプローブとして I-jet または J-Link の使用を推奨しています。

3.2.1 I-jet を使用する場合

- (1) I-jet 付属の USB マイクロケーブルを使用して、IAR Embedded Workbench をインストール済みの PC と I-jet を接続します。接続すると、デバイスドライバが自動で設定されます。
- (2) I-jet 付属の MIPI-20 ケーブルと ARM-20 ピッチ変換アダプタを使用して I-jet と評価ボード (S5U1C31xxxTx) を接続します (図 3.2.1.1 参照)。

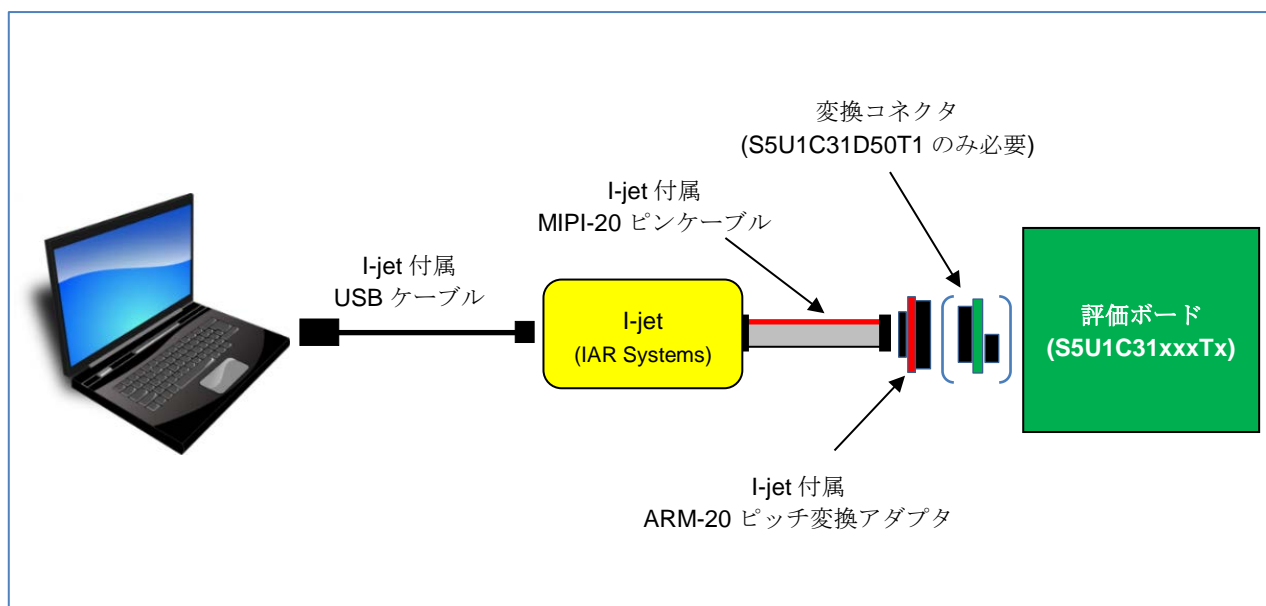


図 3.2.1.1 I-jet を使用する場合のハードウェアの接続例*1

*1: 図 3.2.1.1 は、評価ボード (S5U1C31xxxTx) をデバッグプローブ (I-jet) から供給される 5V 電源で動作させた場合の例です。

I-jet 使用時のハードウェアの接続に関する詳細については、“IAR デバッグプローブガイド” および “S5U1C31xxxTx マニュアル” (表 2.2.1 記載) を参照ください。

3.2.2 J-Link を使用する場合

- (1) J-Link 付属の USB ケーブルを使用して、IAR Embedded Workbench または MDK-ARM をインストール済みの PC と J-Link を接続します。接続すると、デバイスドライバが自動で設定されます。
- (2) J-Link 付属の 20 ピンリボンケーブルを使用して J-Link と評価ボード (S5U1C31xxxTx) を接続します (図 3.2.2.1 参照)。
- (3) J-Link から評価ボード (S5U1C31xxxTx) に 5V 電源を供給するために、以下の手順で J-Link の設定を行います。評価ボードに外部電源を供給する場合は、以下の手順は不要です。
 - (3.1) SEGGER 社の Web サイトより、Ver.6.xx 以降の J-Link Software and Documentation Pack をダウンロードします。
 - (3.2) ダウンロードした J-Link Software and Documentation Pack (*.exe) をダブルクリックしてインストールします。
 - (3.3) Windows のスタートメニューから [SEGGER - J-Link V6.xx > J-Link Commander V6.xx] を起動します。コマンドライン入力のコンソール画面が表示されます (図 3.2.2.2 参照)。
 - (3.4) コンソール画面に、以下のコマンドを入力します。

```
J-Link> power on perm  
J-Link> exit
```

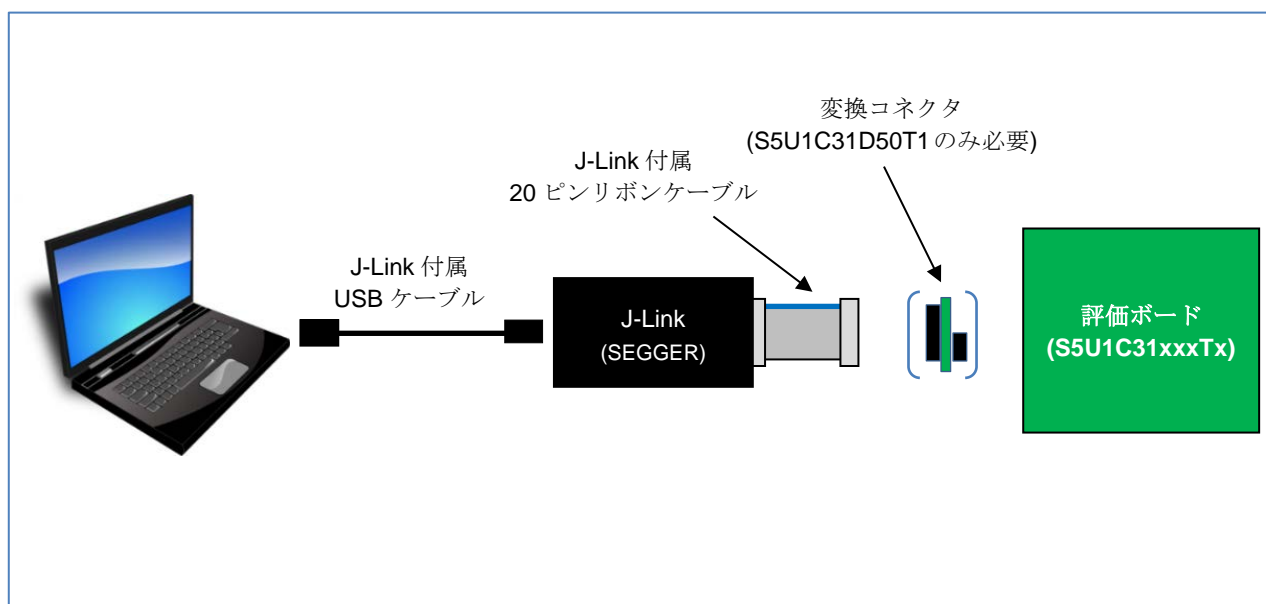


図 3.2.2.1 J-Link を使用する場合のハードウェアの接続例^{*1}

*1: 図 3.2.2.1 は、評価ボード (S5U1C31xxxTx) をデバッグプローブ (J-Link) から供給される 5V 電源で動作させた場合の例です。

3. 開発環境のセットアップ

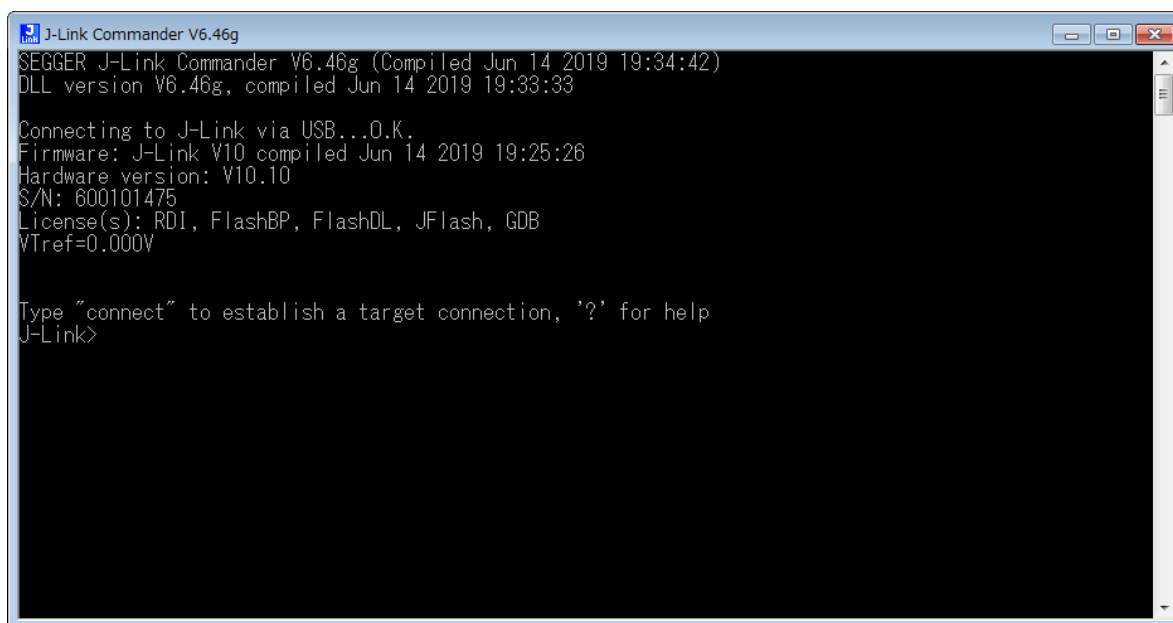


図 3.2.2.2 J-Link Commander の起動

J-Link 使用時のハードウェアの接続に関する詳細については、“J-Link/J-Trace User Guide” および “S5U1C31xxxTx マニュアル”（表 2.2.1 記載）を参照ください。

4. ソフトウェアの開発・評価

本章では、弊社より提供するサンプルソフトウェアと評価ボードを使用したソフトウェアのビルド及びデバッグの手順について説明します。

4.1 サンプルソフトウェアのインストール

本節では、サンプルソフトウェアのインストールの手順について説明します。新規に S1C31 Family のソフトウェアを開発する際は、事前にサンプルソフトウェアのインストールを実行してください。

- (1) S1C31xxx 周辺回路サンプルソフトウェアパッケージ (.zip) を弊社マイクロコントローラ Web サイトからダウンロードして任意のフォルダに展開します。
- (2) 展開先のフォルダから “s1c31xxxsp_verx_xx_xxxxxx.exe” を選択し、右クリックメニューで [管理者として実行] を選択します。
- (3) インストーラ起動後、インストーラの指示に従ってインストールを実行します。
 1. インストール内容を確認
 2. ソフトウェア使用許諾契約の条項の確認
 3. インストールフォルダの選択、及び、インストール実行

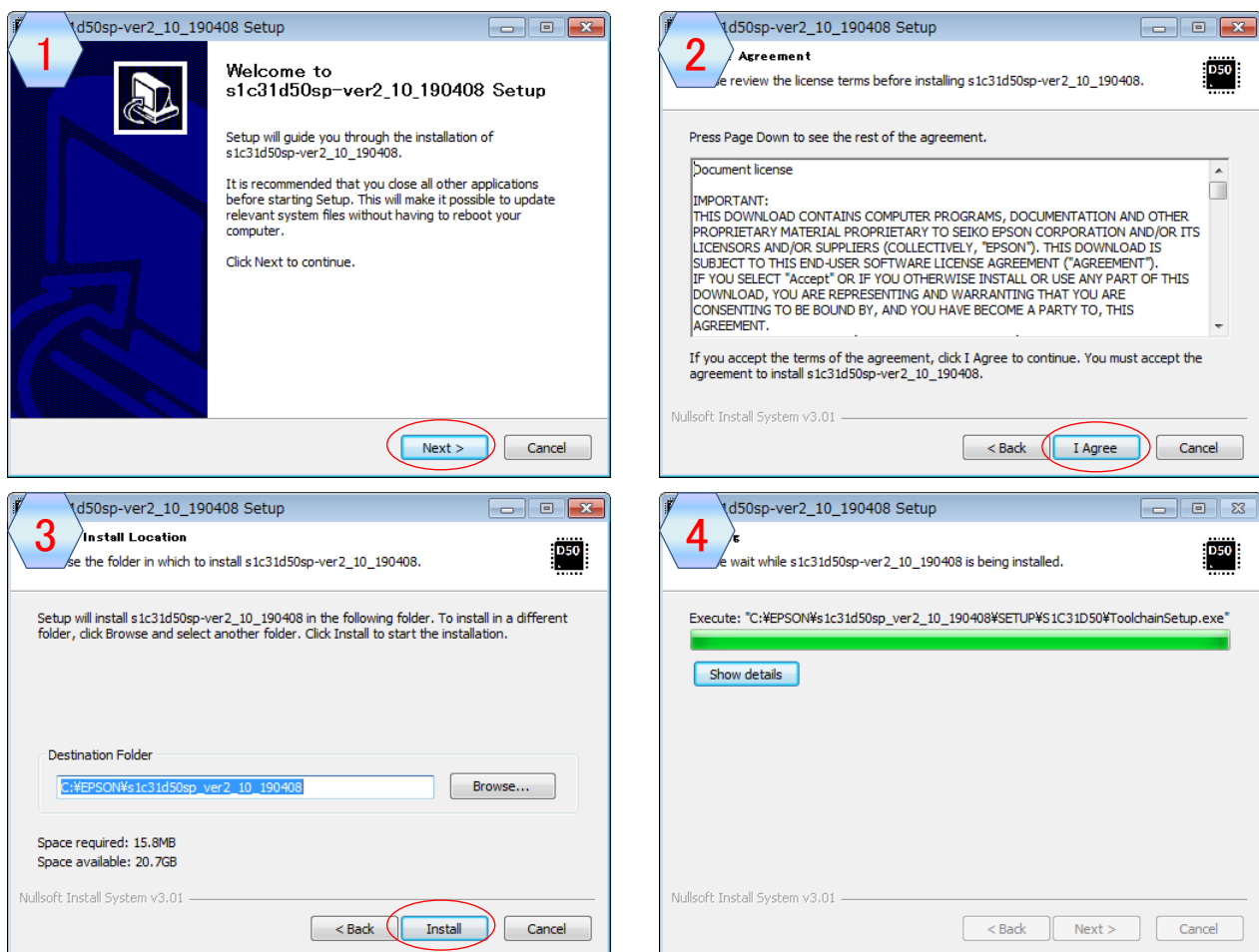


図 4.1.1 インストーラ画面

4. ソフトウェアの開発・評価

(4) インストールが終了すると、セットアップツール（S1C31xxx Toolchain integration utility）が自動で起動します。本ツールの指示に従って、以下に示すステップでセットアップを行います。

1. セットアップ内容の確認
2. セットアップの対象となる IDE の選択
3. デバッグプロブの選択（MDK-ARM を選択した場合のみ）

注: セットアップツールは、IDE 上でソフトウェアを開発するために必要な各種設定ファイルや定義ファイル、フラッシュローダ等を IDE のインストールフォルダへコピーします。

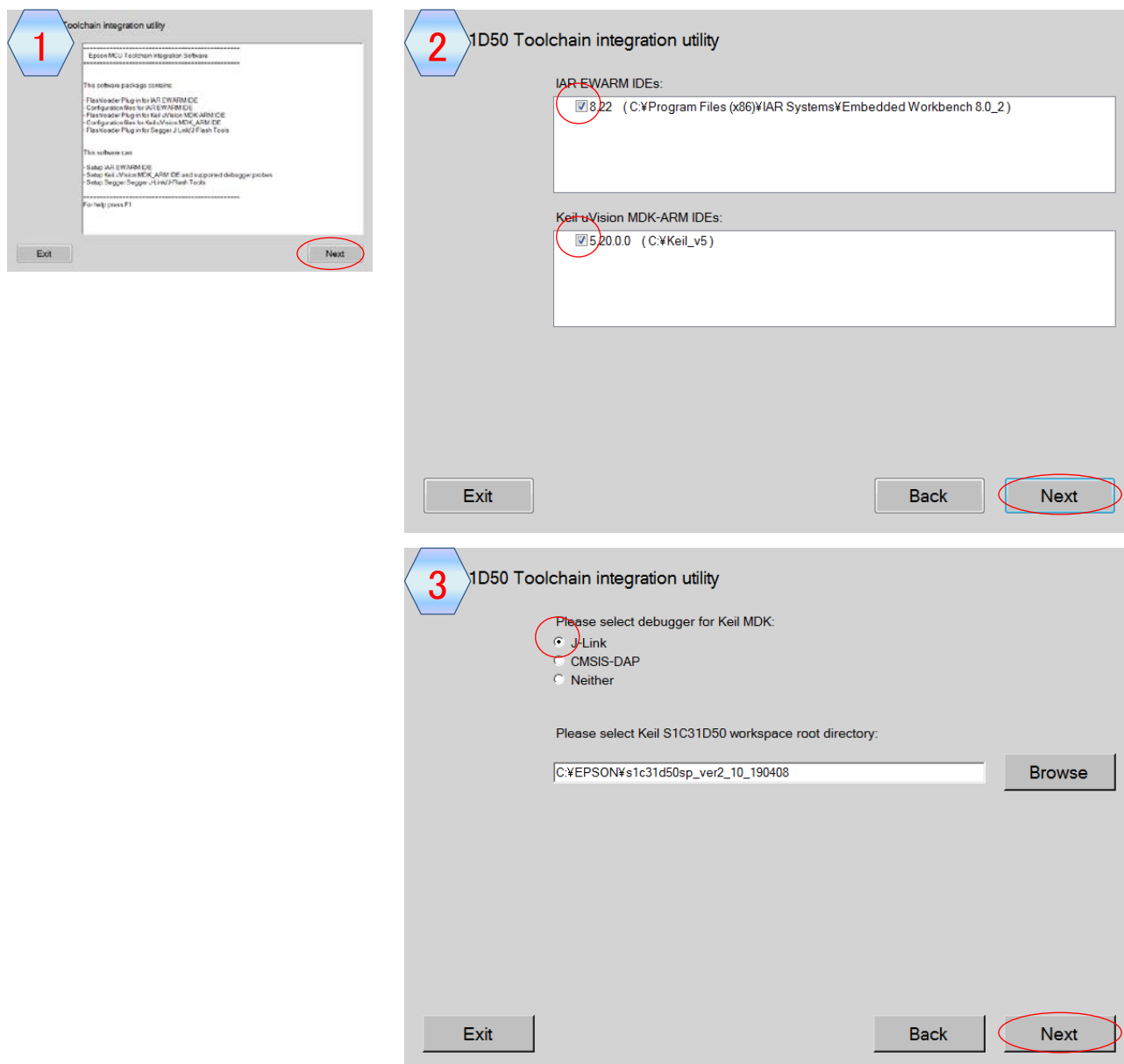


図 4.1.2 セットアップツール画面

(6) 最後に、セットアップツールおよびインストーラを終了します。

サンプルソフトウェアの詳細については、“S1C31xxx 周辺回路サンプルソフトウェアマニュアル”（表 2.2.1 記載）を参照ください。

4.2 ソフトウェアのビルドおよびデバッグ

本節では、ソフトウェアのビルドおよびデバッグの方法について説明します。

4.2.1 IAR EWARM を使用する場合

(1) IAR EWARM の起動

Windows のスタートメニューから [IAR Systems > IAR Embedded Workbench for ARM 8.xx > IAR Embedded Workbench] をクリックして IAR EWARM を起動します。

(2) ワークスペースのオープン

メニュー [ファイル > ワークスペースを開く] を選択して、以下のワークスペースをオープンします。

“C:\¥EPSON¥s1c31xxxsp_verx_xx_xxxxxx¥Examples¥WORKSPACE¥S5U1C31xxxTx¥IARExamples.eww”

(3) アクティブプロジェクトの選択

[ワークスペース] ウィンドウ上部のドロップダウンリストから、ビルド対象のサンプルソフトウェアプロジェクトを選択します (図 4.2.1.1)。

サンプルソフトウェアの各プロジェクトには、以下に示す 3 種類のビルド構成が用意されています。

- Debug - 内蔵 RAM 上で実行されるプログラム ROM を生成します。
最適化レベルは“低”に設定されています。
- DebugFlash - 内蔵 Flash メモリ上で実行されるプログラム ROM を生成します。
最適化レベルは“低”に設定されています。
- ReleaseFlash - 内蔵 Flash メモリ上で実行されるプログラム ROM を生成します。
最適化レベルは“高”に設定されています。

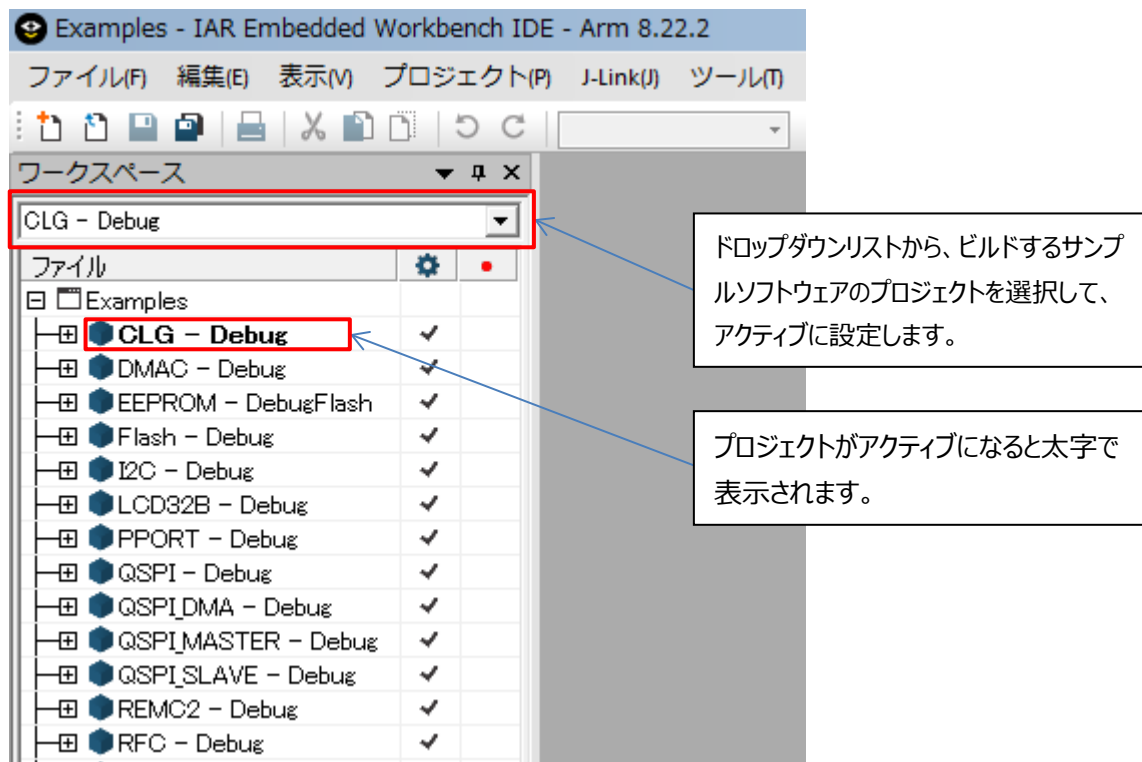


図 4.2.1.1 アクティブプロジェクトの選択

4. ソフトウェアの開発・評価

(4) デバッグプローブの設定

使用するデバッグプローブの種類に応じて、以下の手順でドライバを設定します。

1. メニュー [プロジェクト > オプション] を選択します。
2. 表示されたダイアログの [カテゴリ] リストから [デバッグ] を選択します (図 4.2.1.2)。
3. [設定] タブを選択し、[ドライバ] のドリップダウンリストから使用するデバッグプローブの種類を選択します (図 4.2.1.2)。
 - ・ I-jet を使用する場合は “I-jet/JTAGjet” を選択してください。
 - ・ J-Link を使用する場合は “J-Link/J-Trace” を選択してください。

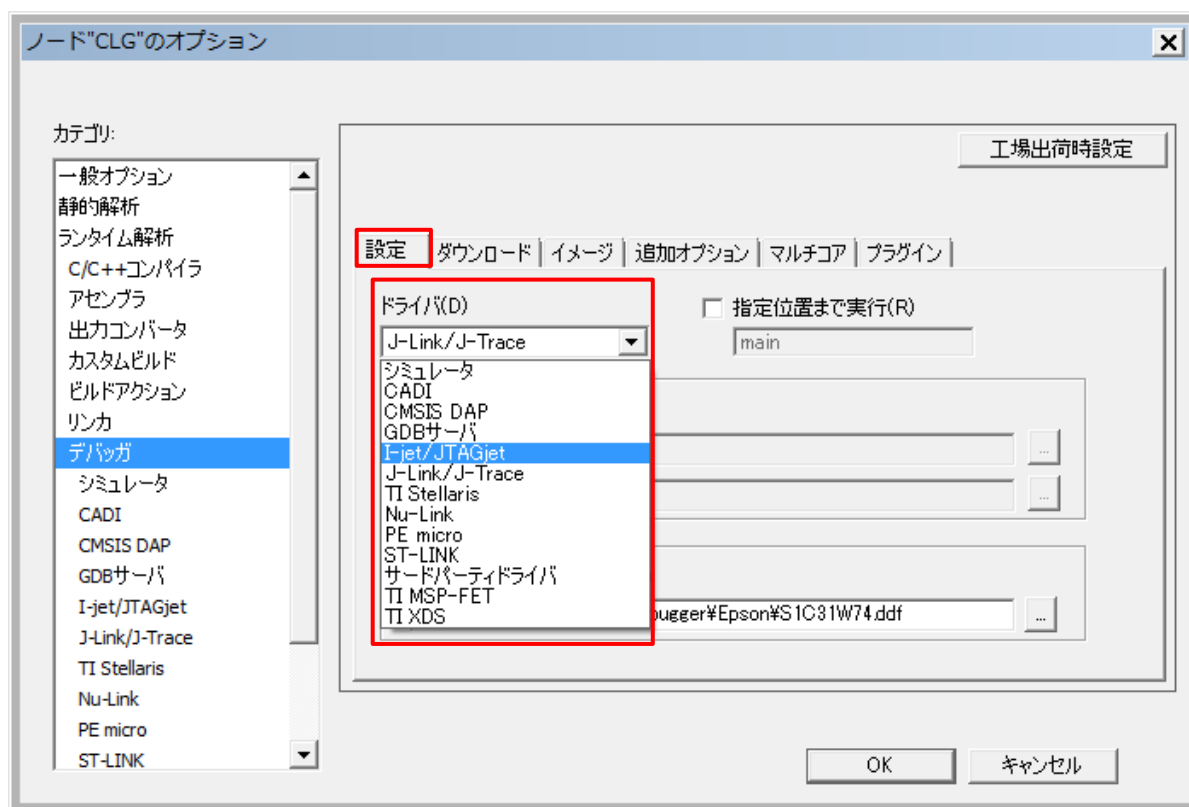


図 4.2.1.2 デバッグプローブの設定

(5) フラッシュローダの設定

アクティブプロジェクトのビルド構成が DebugFlash または ReleaseFlash の場合、内蔵 Flash メモリにプログラムをロード (Flash プログラミング) するためのフラッシュローダを設定する必要があります。

フラッシュローダはデフォルトで設定されています。フラッシュローダが設定されていない場合は、以下の手順に従って行ってください。

1. メニュー [プロジェクト > オプション] を選択します。
2. 表示されたダイアログの [カテゴリ] リストから [デバッグ] を選択します (図 4.2.1.3)。
3. [ダウンロード] タブを選択します (図 4.2.1.3)。
4. [フラッシュローダを使用する(U)] のチェックを有効にします (図 4.2.1.3)。
5. [デフォルトの.board ファイルのオーバーライド] のチェックを有効にします (図 4.2.1.3)。
6. [...] ボタンをクリックし、“S1C31xxx_int.board” を選択します (図 4.2.1.3)。

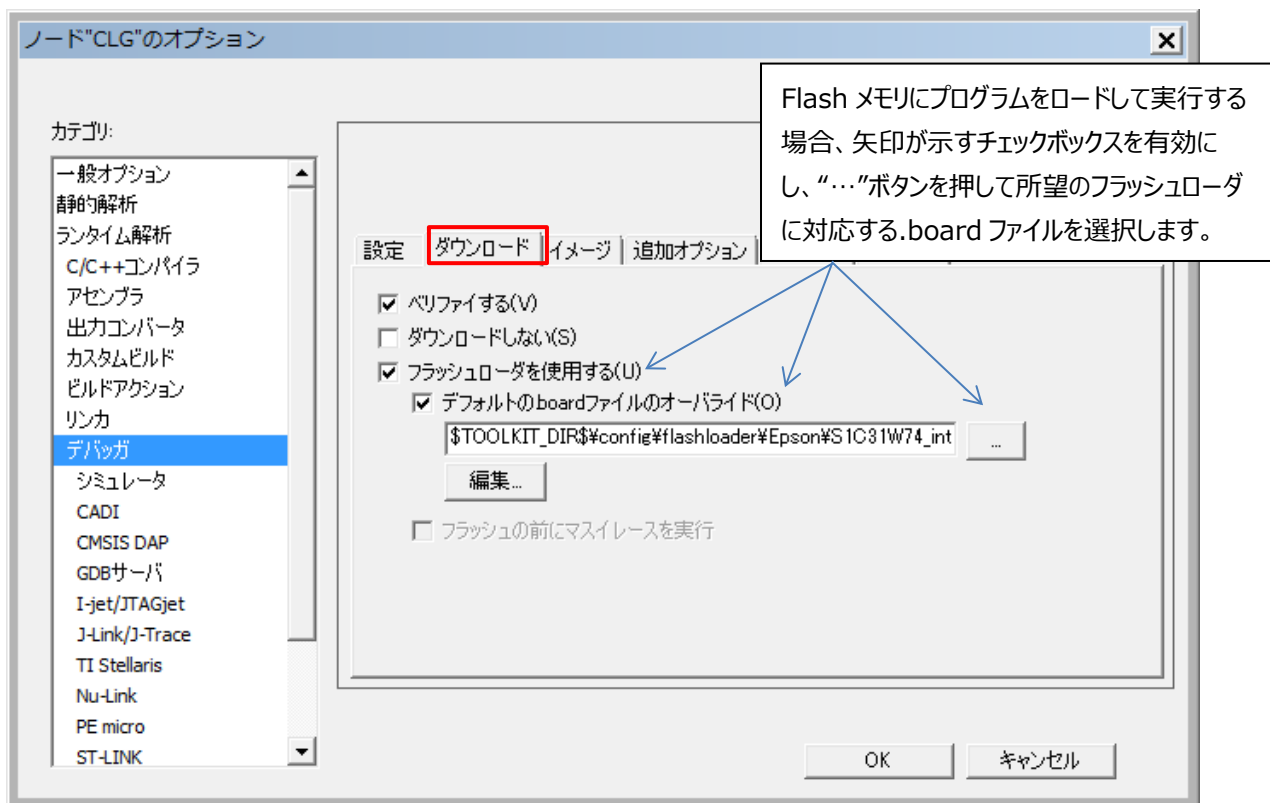


図 4.2.1.3 フラッシュローダの設定


4. ソフトウェアの開発・評価

(6) アクティブプロジェクトのビルド

メニュー [プロジェクト > メイク] を選択して、アクティブプロジェクトをビルドします (図 4.2.1.4)。

注：ビルド中にリンカエラーが発生する場合、sePeriphLibrary または seGraphicsLibrary のライブラリがビルドされていない可能性があります。これらのライブラリをビルドしてから、再度、ターゲットプロジェクトをビルドしてください。このとき、ライブラリはターゲットプロジェクトと同じビルド構成を選択してください。例えば、“LCD32B-Debug” をビルドする場合は、“sePeriphLibrary - Debug”、“seGraphicsLibrary-Debug” を選択してビルドしてください。

(7) アクティブプロジェクトのダウンロードおよびデバッグ

メニュー [プロジェクト > ダウンロードしてデバッグ] を選択、または、ツールバー上の  (ダウンロードしてデバッグ) ボタンをクリックして、デバッグを開始します (図 4.2.1.4)。

ビルド操作に関する詳細については、“IDE プロジェクト管理およびビルドガイド” (表 2.2.1 記載) を参照ください。また、デバッグ操作に関する詳細については、“C-SPY® デバッグガイド” (表 2.2.1 記載) を参照ください。

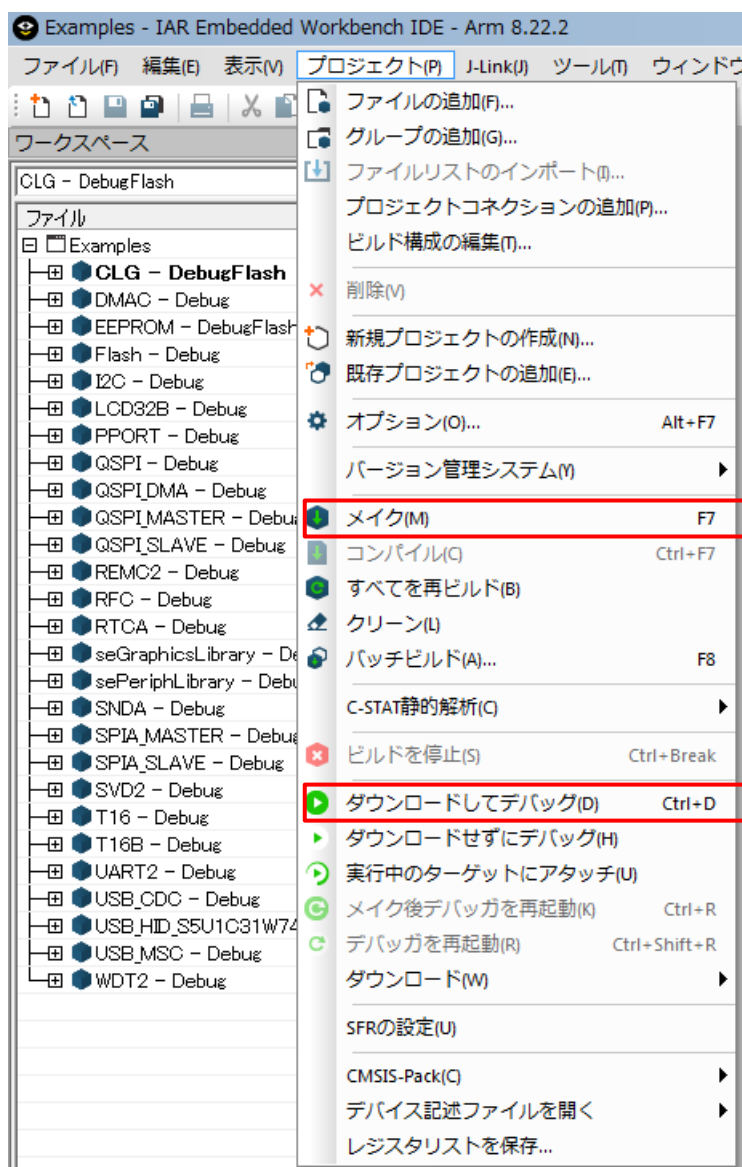


図 4.2.1.4 アクティブプロジェクトのビルドおよびデバッグ

4.2.2 MDK-ARM を使用する場合

(1) MDK-ARM (uVision) の起動

Windows のデスクトップ上の [Keil uVision5] のショートカットアイコンをクリックして uVision を起動します。

(2) ワークスペースのオープン

メニュー [Project > Open Project...] をクリックして、以下のワークスペースをオープンします。

“C:\¥EPSON¥s1c31xxxsp_verx_xx_xxxxxx¥Examples¥WORKSPACE¥S5U1C31xxxTx¥ARM¥Examples.uvmpw”

(3) アクティブプロジェクトの選択

[Project] ウィンドウからビルド対象のプロジェクトを選択し、右クリックメニューの [Set as Active Project] を選択します (図 4.2.2.1)。次に、ツールバー上のプルダウンメニューからビルド構成を選択します (図 4.2.2.2)。

サンプルソフトウェアの各プロジェクトには、以下に示す 2 種類のビルド構成が用意されています。

- Debug - 内蔵 RAM 上で実行されるプログラム ROM を生成します。
- DebugFlash - 内蔵 Flash メモリ上で実行されるプログラム ROM を生成します。

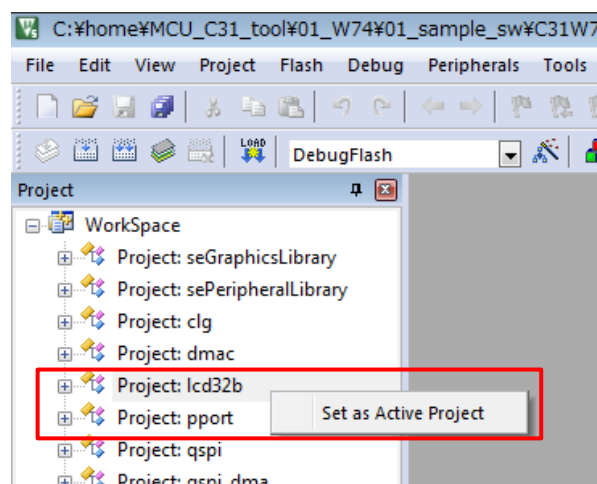
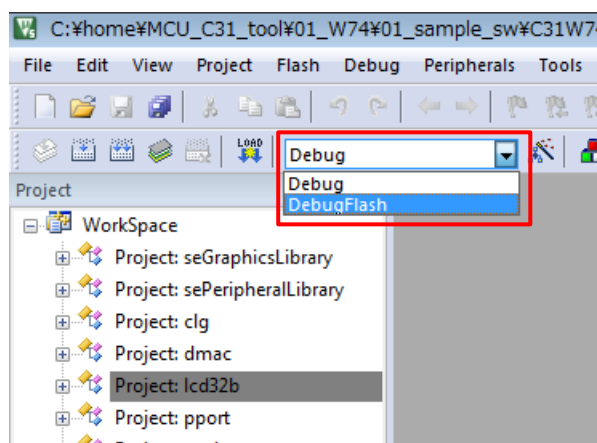


図 4.2.2.1 アクティブプロジェクトの選択



4. ソフトウェアの開発・評価

図 4.2.2.2 ビルド構成の選択

(4) デバッグプロープの設定

使用するデバッグプロープの種類に応じて、以下の手順でドライバを設定します。

1. メニュー [Project > Options for {project} - Target '{build configuration}'] を選択します。
2. [Options for Target '{build configuration}'] ダイアログの [Debug] タブを選択します。
3. [Use:] チェックボックスの右横にあるドロップダウンリストから [J-Link/J-TRACE Cortex] を選択します (図 4.2.2.3)。
4. 上記ドロップダウンリストの右横にある [Settings] ボタンを押下します (図 4.2.2.3)。
5. [Cortex JLink/JTrace Target Driver Setup] ダイアログの [Port] ドロップダウンリストから [SW] を選択します (図 4.2.2.4)。
6. 最後に[OK]ボタンを押下して、すべてのダイアログを閉じます。

注：デバックプロープの設定は、デバックプロープ (J-Link) を PC に接続した状態で行ってください。

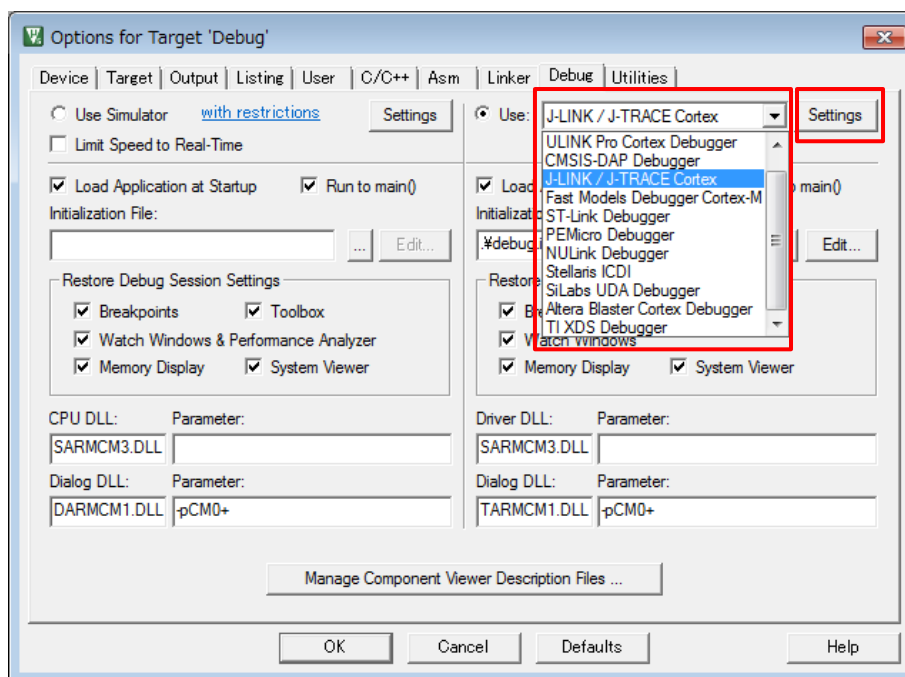


図 4.2.2.3 デバックプロープの選択

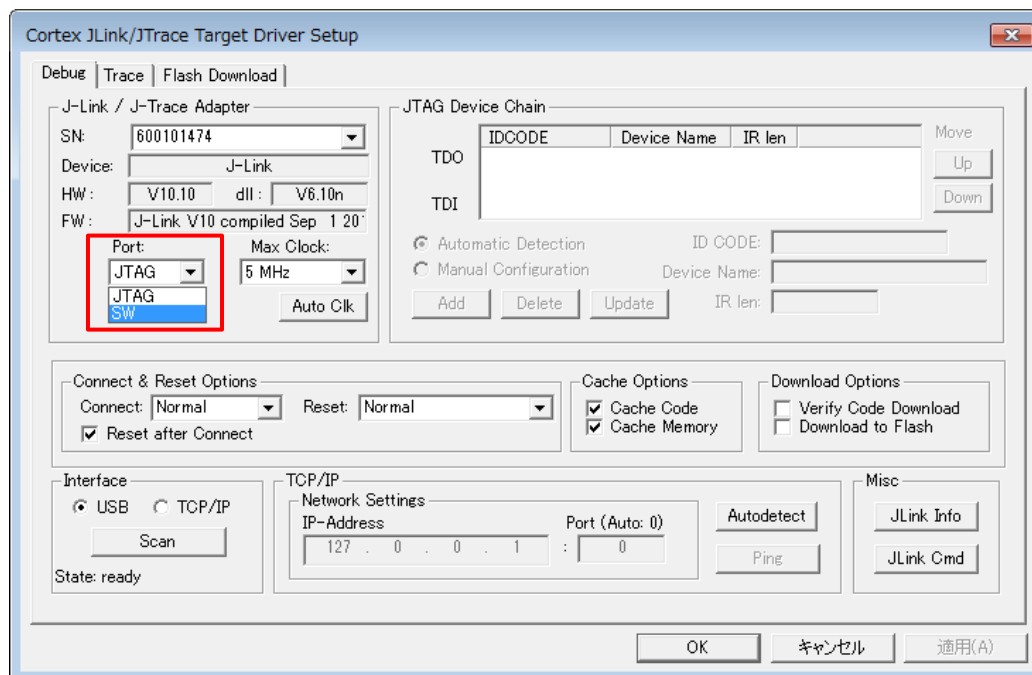


図 4.2.2.4 デバッグプローブの通信設定（J-Link）

4. ソフトウェアの開発・評価

(5) debug.ini ファイルの設定

アクティブプロジェクトのビルド構成が Debug の場合、以下の手順で debug.ini ファイルを設定します。ビルド構成が DebugFlash の場合は、debug.ini ファイルを設定しないでください。

1. メニュー [Project > Options for {project} - Target '{build configuration}'] を選択します。
2. [Options for Target '{build configuration}'] ダイアログの [Debug] タブを選択します。
3. [Initialize File] エディットボックス右横の [...] ボタンを押下し、debug.ini ファイルを指定します。(図 4.2.2.5)。
4. 最後に[OK]ボタンを押下して、すべてのダイアログを閉じます。

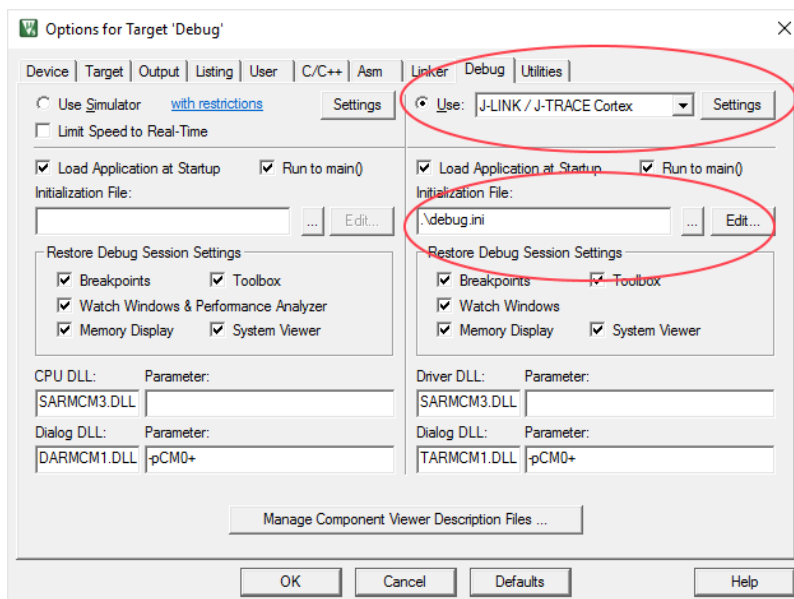


図 4.2.2.5 debug.ini ファイルの設定

(6) フラッシュローダの設定

アクティブプロジェクトのビルド構成が DebugFlash の場合、内蔵 Flash メモリにプログラムをロード (Flash プログラミング) するためのフラッシュローダを設定する必要があります。

フラッシュローダはデフォルトで設定されています。フラッシュローダが設定されていない場合は、以下の手順に従って行ってください。

1. メニュー [Project > Options for {project} - Target '{build configuration}'] を選択します。
2. [Options for Target '{build configuration}'] ダイアログの [Utilities] タブを選択し、[Configure Flash Menu Command] 枠内の [Settings] ボタンを押下します (図 4.2.2.6)。
3. [Cortex JLink/JTrace Target Driver Setup] ダイアログの [Flash Download] タブを選択し、[Remove] ボタンを押下して [Programming Algorithm] リスト上のフラッシュローダを全て削除します (図 4.2.2.7)。次に [Add] ボタンを押下します (図 4.2.2.8)。
4. [Add Flash Programming Algorithm] ダイアログからフラッシュローダ “S1C31xxxint ***kB Flash” を選択し、[Add] ボタンを押下します (図 4.2.2.9)。
5. [Download Function] 枠内の [Erase Sectors]、[Program]、[Verify] のチェックを有効にし、それ以外のチェックを無効にします (図 4.2.2.10)。
6. 最後に [OK] ボタンを押下して、すべてのダイアログを閉じます。

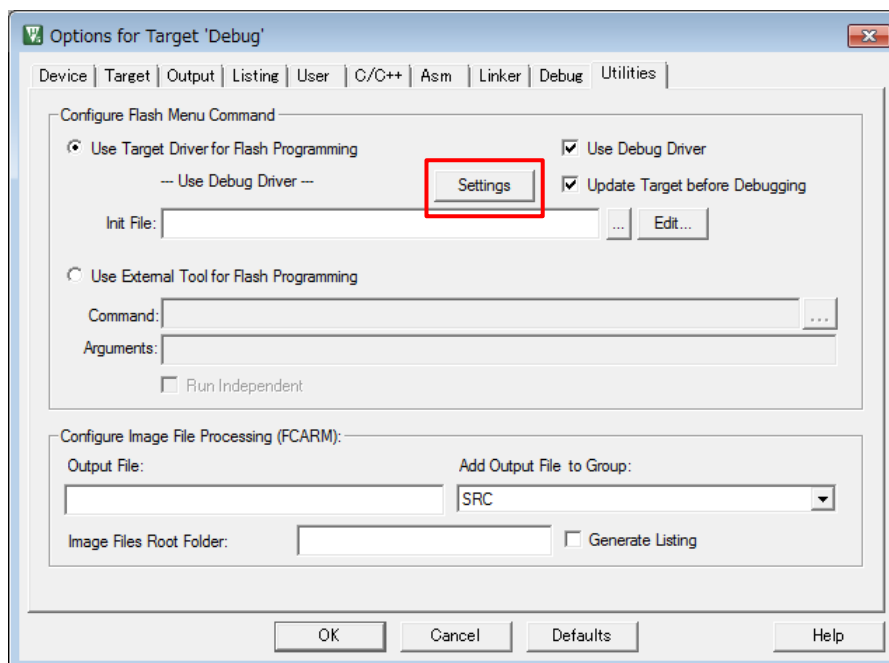


図 4.2.2.6 フラッシュローダの設定

4. ソフトウェアの開発・評価

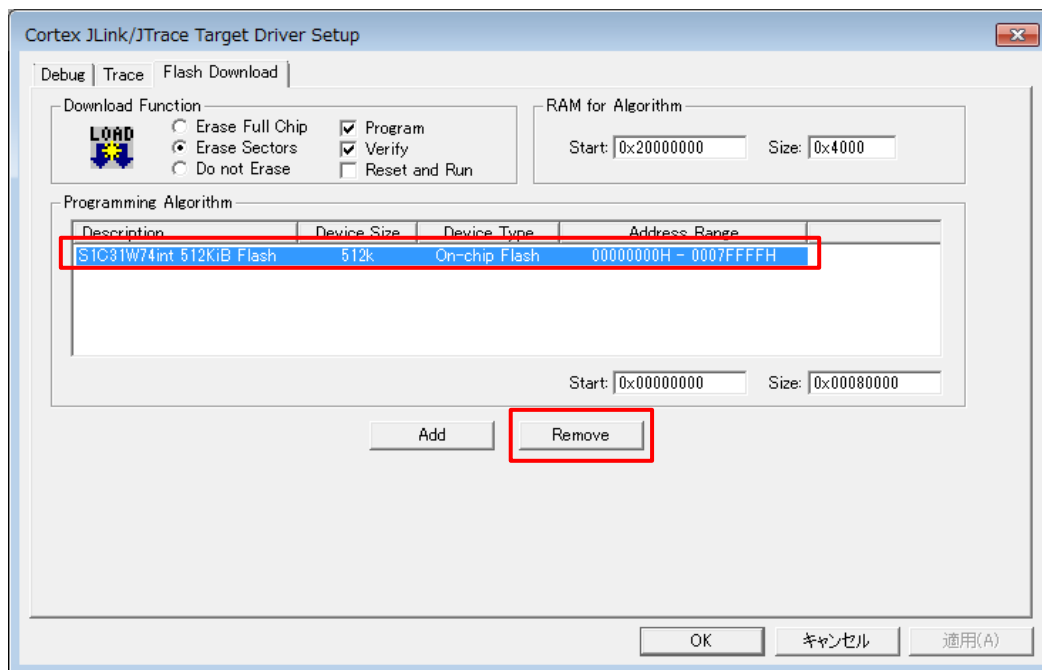


図 4.2.2.7 フラッシュローダの削除

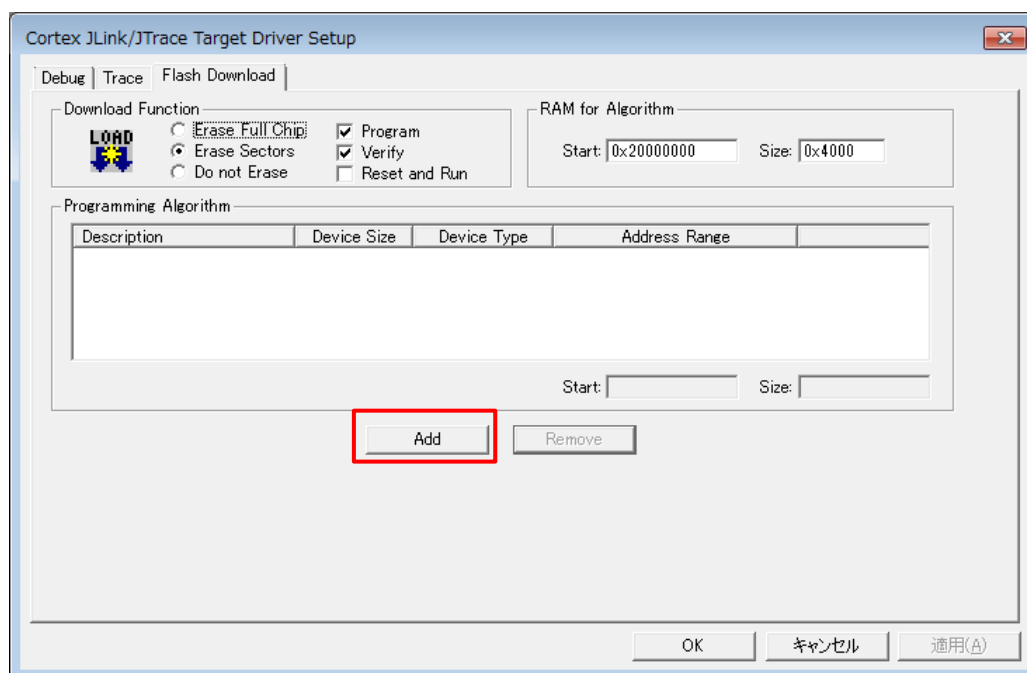


図 4.2.2.8 フラッシュローダの追加

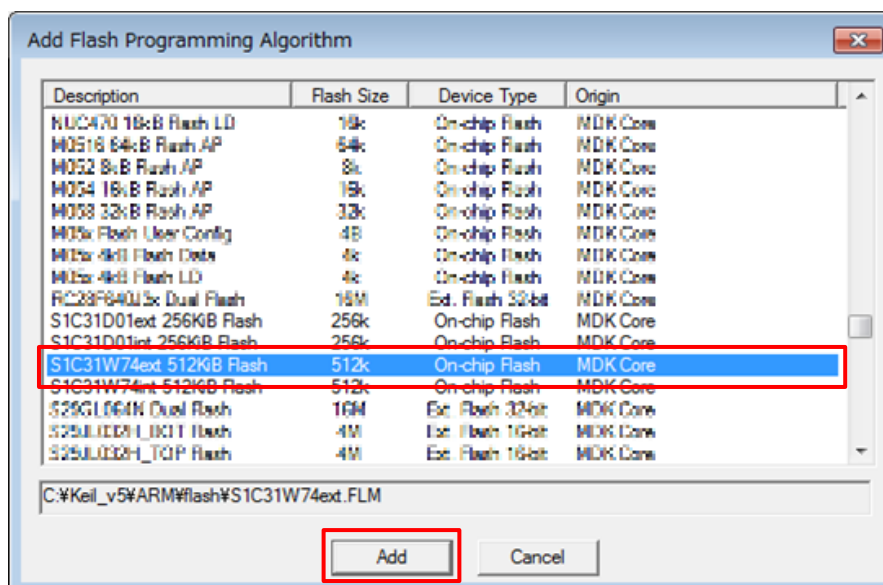


図 4.2.2.9 フラッシュローダの選択

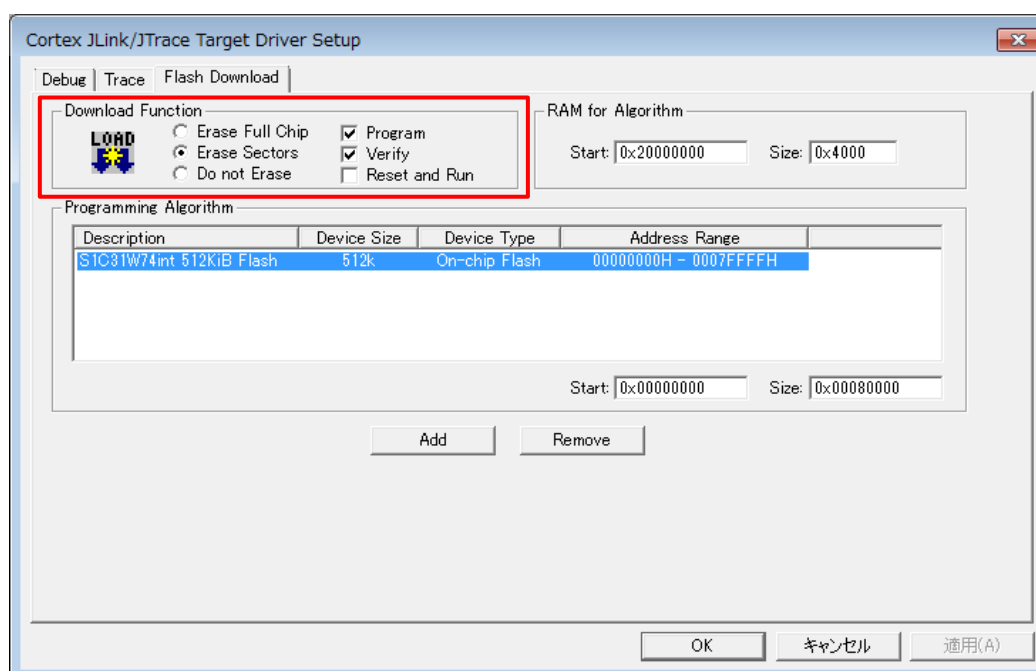


図 4.2.2.10 ダウンロード機能の設定

4. ソフトウェアの開発・評価

(7) アクティブプロジェクトのビルド

メニュー [Project > Build '{project name}' ({build configuration})] を選択して、アクティブプロジェクトをビルドします (図 4.2.2.11)。

注：ビルド中にリンカエラーが発生する場合は、sePeriphLibrary と seGraphicsLibrary のライブラリプロジェクトがビルドされていない可能性があります。これらのライブラリプロジェクトをビルドしてから、再度、アクティブプロジェクトをビルドしてください。

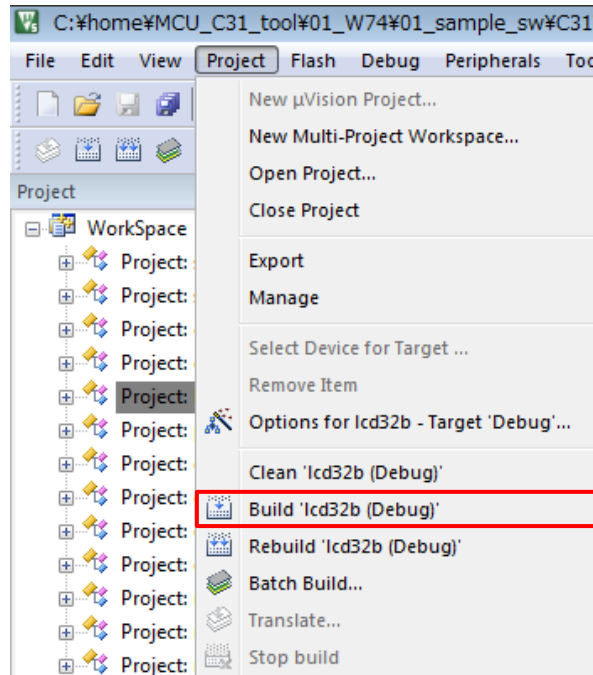


図 4.2.2.11 アクティブプロジェクトのビルド

(8) アクティブプロジェクトのダウンロードおよびデバッグ

メニュー [Flash > Download] を選択してアクティブプロジェクトのプログラムをターゲットボードにロードします (図 4.2.2.12)。続いて、メニュー [Debug > Start/Stop Debug Session] を選択してデバッグを開始します (図 4.2.2.13)。

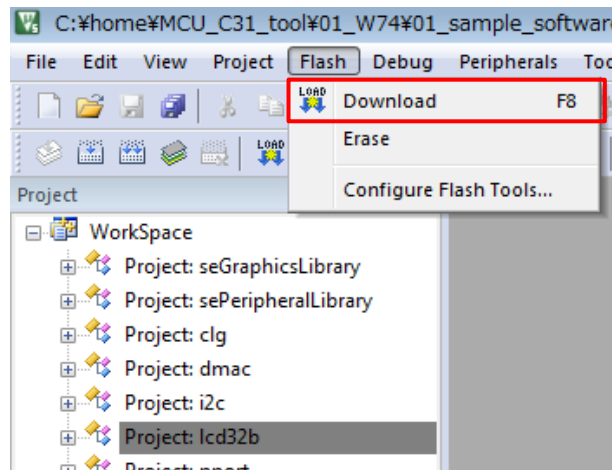


図 4.2.2.12 アクティブプロジェクトのダウンロード

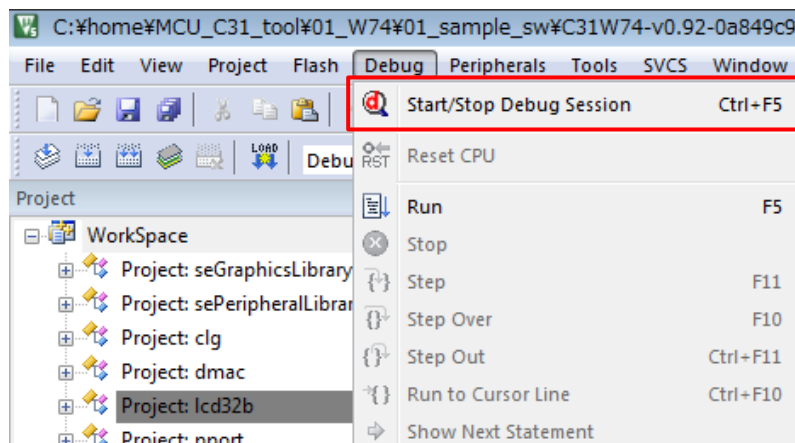


図 4.2.2.13 アクティブプロジェクトのデバッグ

ビルド・デバッグ操作に関する詳細については、“Complete User’s Guide Selection” (表 2.2.1 記載) を参照ください。

改訂履歴表

[illegible]

セイコーエプソン株式会社

営業本部 デバイス営業部

東京 〒191-8501 東京都日野市日野 421-8
TEL (042) 587-5313 (直通) FAX (042) 587-5116

大阪 〒530-6122 大阪市北区中之島 3-3-23 中之島ダイビル 22F
TEL (06) 7711-6770 (代表) FAX (06) 7711-6771

ドキュメントコード : 413445902
2017 年 4 月 作成
2019 年 7 月 改訂