



ML63Q2500

ソフトウェア開発環境

チュートリアル

ご注意

ローム製品取扱い上の注意事項

- 1) 本製品をご使用の際は、最新の製品情報をご確認の上、絶対最大定格^(*)、動作条件その他の指定条件の範囲内でお使いください。指定条件の範囲を超えて使用された場合や、使用上の注意を守ることなく使用された場合、その後に発生した故障、誤動作等の不具合、事故、損害等については、ローム株式会社（以下、「当社」といいます）はいかなる責任も負いません。また、指定条件の範囲内のご使用であっても、半導体製品は種々の要因で故障・誤作動する可能性があります。万が一本製品が故障・誤作動した場合でも、その影響により人身事故、火災損害等が起こらないよう、お客様の責任において、ディレーティング、冗長設計、延焼防止、バックアップ、フェイルセーフ等お客様の機器・システムとしての安全確保を行ってください。
(*)絶対最大定格：瞬時たりとも超過してはならない限界値となります。
- 2) 本資料に掲載されております製品は、耐放射線設計がなされておられません。
- 3) 本資料に記載されております応用回路例やその定数、ソフトウェア等の情報は、半導体製品の標準的な動作例や応用例を説明するものです。お客様の機器やシステムの設計においてこれらの情報を使用する場合には、お客様の責任において行ってください。また、量産設計をされる場合には、外部諸条件を考慮してください。これらのご使用に起因して生じた損害等に関し、当社は一切その責任を負いません。
- 4) 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の技術情報は、それをもって当該技術情報に関する当社または第三者の知的財産権その他の権利を許諾するものではありません。したがって、当該技術情報を使用されたことによる第三者の知的財産権に対する侵害またはこれらに関する紛争について、当社は何ら責任を負うものではありません。
- 5) 当社は、本資料に明示した用途で本製品が使用されることを意図しています。本資料に明示した用途以外への使用を検討される場合は、必ず営業窓口までお問い合わせください。また、本製品を、医療機器分類クラスⅢ、Ⅳに該当する用途に使用される際は、必ず当社へご連絡の上、書面にて承諾を得てください。
本製品を、直接生命・身体に危害を及ぼす可能性のある機器・システム、極めて高い信頼性を要求される機器（航空宇宙機器、原子力制御機器、海底中継機器等）に使用することはできません。当社の事前の書面による承諾なく、当社の意図していない用途に製品を使用したことにより生じた損害等に関し、当社は一切その責任を負いません。
- 6) 本製品は、RoHS 指令など適用される環境関連法令を遵守の上ご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関して、当社は一切の責任を負いません。
- 7) 本製品および本資料に記載の技術を輸出または国外へ提供する際には、「外国為替及び外国貿易法」、「米国輸出管理規則」など適用される輸出関連法令を遵守し、それらの定めにしたがって必要な手続を行ってください。
- 8) 本資料に記載されている内容または本製品についてご不明な点がございましたら営業窓口までお問い合わせください。
- 9) 本資料の一部または全部を当社の許可なく、転載・複写することを堅くお断りします。

その他の注意事項

- 1) 本資料に記載の内容は、改良などのため予告なく変更することがあります。本製品のご使用、ご購入に際しては、必ず事前に営業窓口で最新の情報をご確認ください。
- 2) 本資料に記載されております情報は、正確を期すため慎重に作成したものです。万が一、当該情報の誤り・誤植に起因して、お客様に損害が生じた場合においても、当社はその責任を負うものではありません。

TSZ72037・01・001

目次

1. 概要	1
1.1 準備するもの	1
1.2 関連するマニュアル	2
2. インストール	3
2.1 LAPIS Development Tools LEXIDE-Ω のインストール.....	3
2.2 ML63Q2500 グループ リファレンスソフトウェアのインストール.....	3
2.3 機種情報ファイルのインストール	3
3. プログラム作成・デバッグ	5
3.1 プログラム作成手順.....	5
3.2 LEXIDE-Ω の起動.....	6
3.3 LEXIDE-Ω プロジェクト新規作成.....	7
3.4 ソースファイルのプロジェクトへの登録.....	10
3.5 プログラム作成・編集.....	13
3.6 ビルドオプションの設定	14
3.6.1 マクロの設定.....	15
3.6.2 インクルードパスの設定	16
3.6.3 最適化オプションの設定.....	17
3.7 ビルド	17
3.8 デバッグ	18
3.8.1 事前準備	18
3.8.2 デバッグコンフィグレーションの作成	19
3.8.3 デバッグの開始.....	22
3.8.4 プログラムの実行開始・再開	23
3.8.5 プログラムの実行停止	23
3.8.6 ブレークポイントの設定.....	24
3.8.7 デバッグの終了	24
4. 【参考】 Solist-AI™ のファームウェア開発について	25
4.1 サンプルプログラムの制御フロー	25
4.2 サンプルプログラムの読み込み(インポート)方法.....	27
4.3 LEXIDE Scope による波形表示	28
4.3.1 デバッグコンフィグレーション	28
4.3.2 LEXIDE Scope の起動.....	29
4.3.3 LEXIDE Scope の使い方	30
改版履歴	1

「Solist-AI™」は、ローム株式会社の商標または登録商標です。

「Arm™」、「Cortex™」は、Arm Limited (またはその子会社または関連会社)の商標または登録商標です。

1. 概要

本書は、Solist-AI™ マイコン用ソフトウェアを開発するための統合開発環境 LEXIDE-Ω のチュートリアルです。本書にしたがって読み進めていくことで、開発環境の各ツールの使い方を習得できます。なお、本書では例として ML63Q2557 を題材に、実際に簡単なプログラムを作成し、実機上で動作させていきながら理解を深めていきます。ML63Q2557 以外の機種をお使いの場合は、適宜読み替えてください。

【注意】

本書に掲載されている画面は作成時点の例であり、バージョンアップ等により実際の画面と異なる場合があります。本書に記載の内容は、開発ツールソフトウェア等の各マニュアル類の補足的な資料となります。使い方の詳細につきましては、お使いの開発ツールソフトウェア等のユーザーズマニュアルをご参照ください。

1.1 準備するもの

本書を読み進める前に、以下を準備してください。

名称	説明
<u>LAPIS Development Tools</u> <u>LEXIDE-Ω (2.1.0 以降)</u>	Arm™ Cortex™-M コアを搭載した弊社製マイクロコントローラ用アプリケーションプログラム開発環境です ^(*) 。
<u>ML63Q2500 Software Pack</u>	ML63Q2500 グループの機種情報ファイルです ^(*) 。
<u>CMSIS-Core(M)</u>	Arm™ Cortex™-M コア向けの機種情報ファイルです ^(*) 。
<u>Solist-AI™ Sim 異常検知対応版</u> (V1.00.04)	Solist-AI™ の機能をパソコン上で確認することができるシミュレーションツールです ^(*) 。異常検知向け(教師なし学習)です。まとまった単位の数値データを使用して学習し(正常状態)、学習した状態と異なる状態(異常状態)の検知が確認できます。
<u>Solist-AI™ Sim 教師あり学習対応版</u> (SLV1.00.04)	Solist-AI™ の機能をパソコン上で確認することができるシミュレーションツールです ^(*) 。異常検知向け(教師なし学習)とは別に、センサデータによる物体識別や、バッテリーの劣化予測など予測/パラメータ推定(教師あり学習)での活用を検討いただく場合はこちらをご使用ください。
<u>ML63Q2500 グループ</u> <u>リファレンスソフトウェア</u>	ML63Q2500 グループ用のペリフェラルドライバと、リファレンスボード上で動作するサンプルプログラムです ^(*) 。
<u>Solist-AI™ Scope</u>	Solist-AI™ の効果を確認するリアルタイム波形表示ツールです ^(*) 。本ツールを使用する場合、USB-SPI 変換モジュールが必要です。※Solist-AI™ Scope の代わりに LEXIDE-Ω 付属の LEXIDE Scope も使用可能です。その場合、USB-SPI 変換モジュールは不要です。詳細は「4.3 LEXIDE Scope による波形表示」をご参照ください。
リファレンスボード	マイクロコントローラのサンプルを搭載したボードです。お客様の目的に応じて必要な部品等を搭載していただくことによって、ML63Q25xx のマイクロコントローラの動作をお試しいただくことができます。
デバッグアダプタ	ソフトウェアのデバッグ、Flash へのプログラム書き込みに使用します。J-Link または CMSIS-DAP 準拠のデバッグアダプタをお使いください。本チュートリアルでは J-Link を使用します。
USB-SPI 変換モジュール (オプション)	USB to SPI(FT232H)を介して PC と接続することで、AI をモニタすることができます。FTDI 製 FT232H を搭載した MM-FT232H または MM-FT232HC(以下: MM-FT232H,MM-FT232HC, Sunhayato Corp.)の使用を推奨します。※Solist-AI™ Scope を使用しない場合は不要です。

*1: ローム WEB サイトの下記ページよりダウンロードいただけます。

<https://www.rohm.co.jp/products/micon/solist-ai/ml63q2500-group>

*2: 以下よりダウンロードいただけます。

<https://www.keil.arm.com/packs/cmsis-arm/versions/>

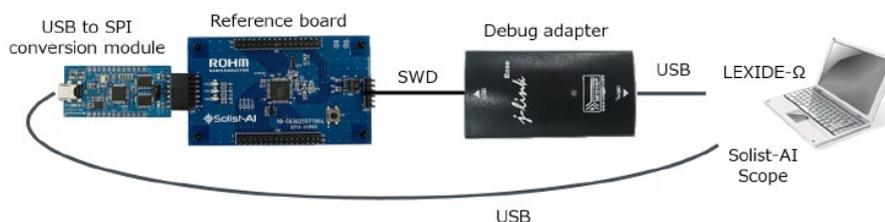


図 1-1. ソフトウェア開発環境の構成

1.2 関連するマニュアル

本書に関連するマニュアル類を以下に示します。併せて参照してください。

名称	説明
<u>ML63Q2500 グループ ユーザーズマニュアル</u>	ML63Q2500 グループのマイクロコントローラのハードウェアマニュアルです ^(*1) 。
LEXIDE-Ω ARM コア搭載 LSI 向け ユーザーズマニュアル	Arm コア搭載 LSI 用プログラム開発のための統合開発環境 LEXIDE-Ω の使用方法を記載したマニュアルです ^(*2) 。 LEXIDE-Ω のプロジェクト作成やデバッグ時の設定など使用方法を確認したい場合に参照ください。
ARM コア向け LEXIDE C コンパイラ ユーザーズマニュアル	LCC-ARM ツールチェーン(C コンパイラ, アセンブラ, リンカ等)の使用方法について記載したマニュアルです ^(*3) 。
ML63Q2500 リファレンスソフトウェア スタートアップマニュアル	ML63Q2500 グループ用のペリフェラルドライバとリファレンスボード上で動作するサンプルプログラムについて記載したマニュアルです ^(*4) 。
ペリフェラルドライバ API マニュアル	ML63Q2500 グループ用のペリフェラルドライバ API について記載した HTML ヘルプ形式のドキュメントです ^(*4) 。
Solist-AI™ Scope Users Guide	メモリの情報をリアルタイムで可視化(グラフ表示)できる デバッグ用 Windows アプリケーション Solist-AI™ Scope の使用方法について記載したマニュアルです ^(*5) 。

*1 : ローム WEB サイトの下記ページよりダウンロードいただけます。

<https://www.rohm.co.jp/products/micon/solist-ai/ml63q2500-group>

*2 : Windows スタートメニューの[LAPIS LEXIDE Software Tools ドキュメント]よりご参照いただけます。

*3 : Windows スタートメニューの[LAPIS LEXIDE Software Toolchain Ver.xxxxxx]よりご参照いただけます。

*4 : ML63Q2500 リファレンスソフトウェアの Document フォルダより参照いただけます。

*5 : Solist-AI™ Scope の Document フォルダより参照いただけます。

2. インストール

開発環境を使うにあたり、以下のツールをお使いの PC にインストールします（ダウンロードの参照先は「1.1 準備するもの」を参照してください）。

- LAPIS Development Tools LEXIDE-Ω
- ML63Q2500 グループ リファレンスソフトウェア
- 機種情報ファイル（ML63Q2500 Software Pack, CMSIS-Core(M)）

2.1 LAPIS Development Tools LEXIDE-Ω のインストール

ダウンロードした LAPIS_LEXIDE_Vxxxx.zip（Vxxxx はバージョンを示す文字列）を任意のフォルダに展開し、フォルダ内の LexideInstaller_yyyymmdd.exe（yyyymmdd はリリースバージョンを示す文字列）をダブルクリックします。インストールはインストーラの指示にしたがってインストールしてください。

2.2 ML63Q2500 グループ リファレンスソフトウェアのインストール

ダウンロードした ML63Q2500_ReferenceSoftware_vxxx.zip（vxxx はリリースバージョンを示す文字列）を任意のフォルダに展開してください。プログラム作成時に、リファレンスソフトウェアのサンプルプログラムを流用する場合は、コピーしてお使いください。

2.3 機種情報ファイルのインストール

機種情報ファイル（ML63Q2500 Software Pack, CMSIS-Core(M)）のインストールは、LEXIDE-Ω より行います。最初に LEXIDE-Ω を起動します。

“Windows スタート > すべて > LAPIS LEXIDE Tools > LEXIDE-Ω” を選択します。しばらくすると、[Select a directory as workspace]ダイアログが表示されます。ここでは、「C:\lexide\workspace_omega_v2_arm」を設定し、[Launch]ボタンをクリックします。

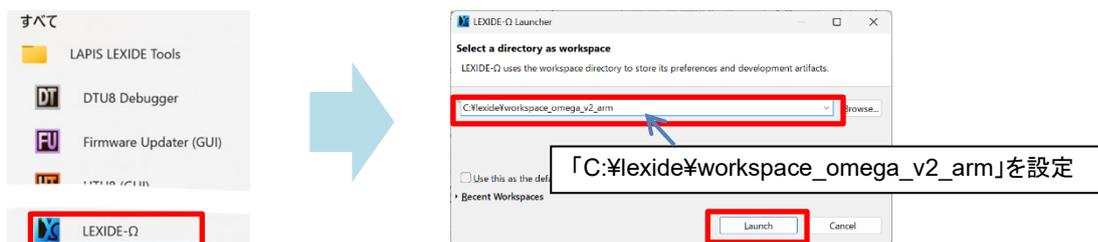


図 2-1. LEXIDE-Ω の起動

LEXIDE-Ω が起動したら、LEXIDE-Ω の CMSIS-Pack Manager を表示します。CMSIS-Pack Manager は、LEXIDE-Ω 右上の  をクリックして表示される [Open Perspective] ダイアログから [CMSIS-Pack Manager] を選択して [Open] ボタンをクリックすると表示されます。

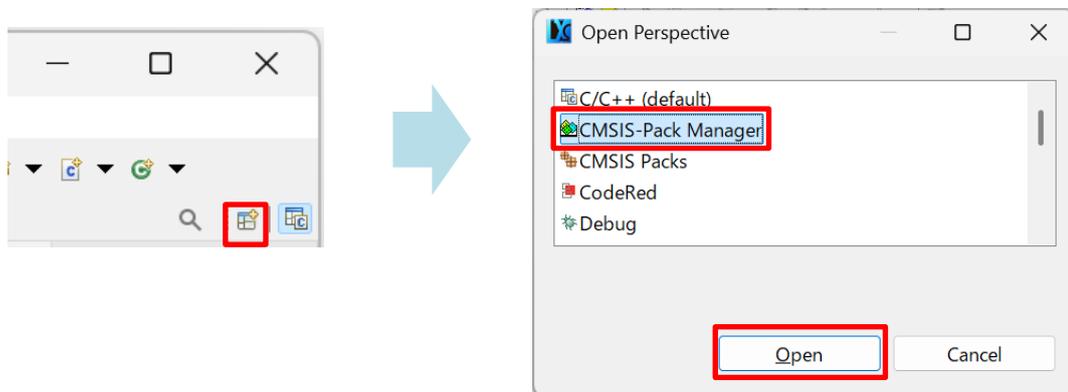


図 2-2. CMSIS-Pack Manager の表示方法

表示された[CMSIS-Pack Manager]パースペクティブ^(*)の[Import existing packs...]をクリックし、表示された[Import Packs]ダイアログにてダウンロードした機種情報ファイル（ここでは ROHM.ML63Q25x7_DFP1.01.pack）を指定して[開く]ボタンをクリックします。

*1：パースペクティブとは、作業ごとによく使うメニューバーやツールバー等を含むビューのレイアウトを定義したものです。

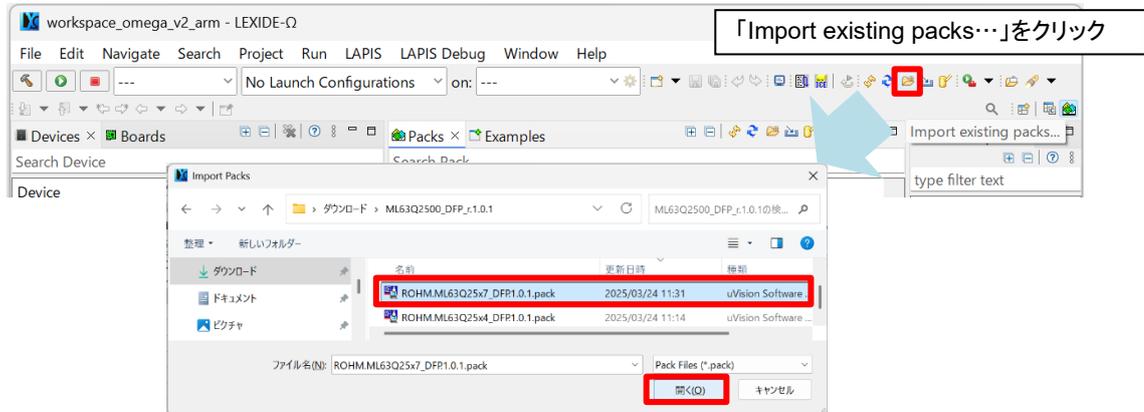


図 2-3. ML63Q2500 Software Pack(ML63Q25x7_DFP)のインストール

インストールが完了すると、以下のように、インストールされた機種情報ファイルが表示され、LEXIDE-Ω上で機種情報ファイルを利用できるようになります。

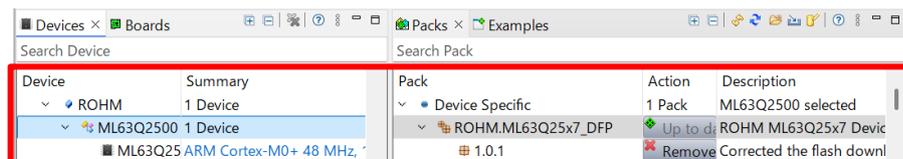


図 2-4. インストールされた機種情報ファイルの表示

同様に,[Import Packs]ダイアログにて CMSIS-Core(M)を指定してインストールします。

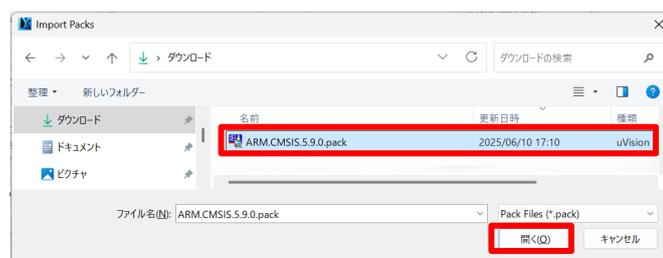


図 2-5. CMSIS-Core(M)のインストール

3. プログラム作成・デバッグ

ここでは、ML63Q2500 リファレンスソフトウェアに含まれるサンプルプログラムを使用します。使用するプログラムはBlinkLEDで、リファレンスボード上のLEDを点滅するプログラムです。点滅のタイミングは低速タイムベースカウンタの割込みにより制御します。

3.1 プログラム作成手順

本チュートリアルでは、以下の手順でプログラムを作成していきます。

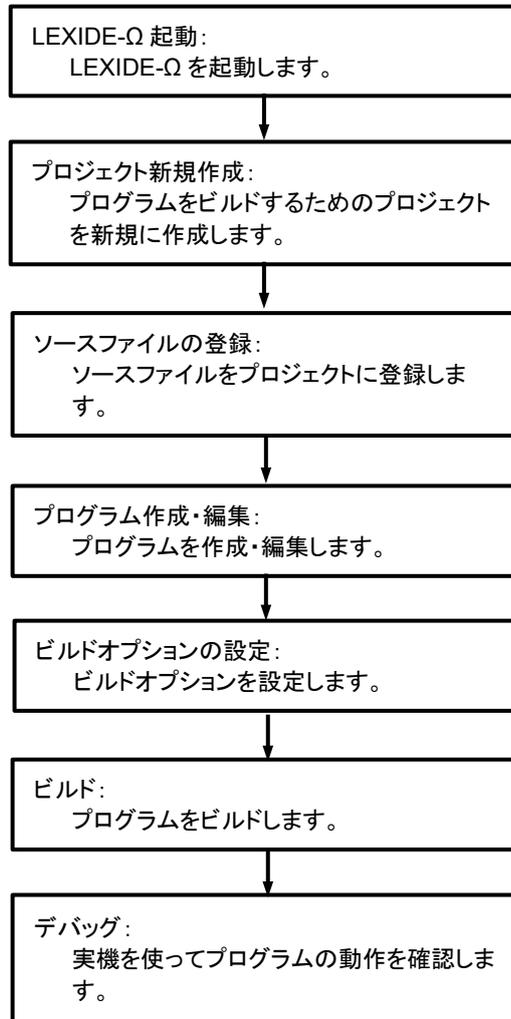


図 3-1. プログラム作成手順

3.2 LEXIDE-Ωの起動

“Windows スタート > すべて > LAPIS LEXIDE Tools > LEXIDE-Ω” を選択します。
 しばらくすると、[Select a directory as workspace]ダイアログが表示されます。ここでは、
 「C:¥lexide¥workspace_omega_v2_arm」を設定し、[Launch]ボタンをクリックします。

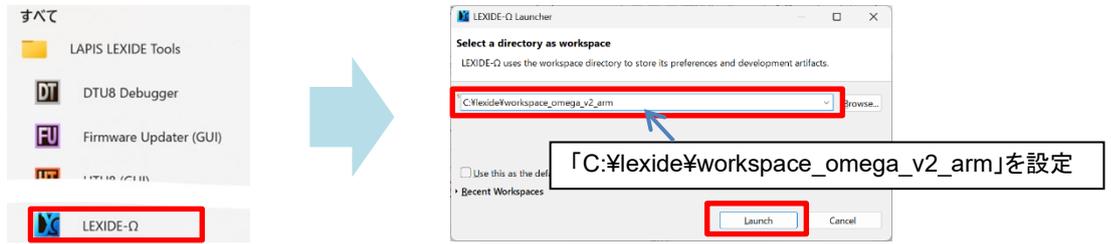


図 3-2. LEXIDE-Ωの起動

しばらくすると、LEXIDE-Ωが起動します。
 LEXIDE-Ωを起動したときの画面は以下のようにになっています。

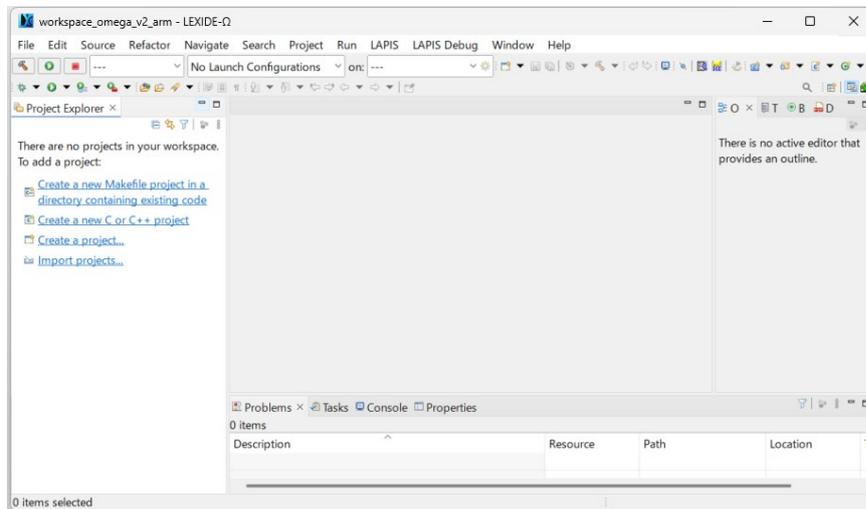


図 3-3. LEXIDE-Ω 起動時の画面

画面が図 3-3 のようになっていない場合は、 をクリックして[C/C++]パースペクティブに切り替えてください。

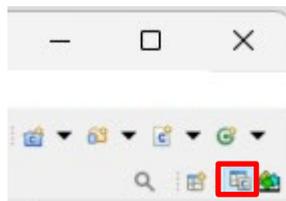


図 3-4. [C/C++]パースペクティブへの切り替え

LEXIDE-Ω を起動すると、下図のように、ワークスペース (C:\lexide\workspace_omega_v2_arm) の下に .metadata フォルダが作成されます。

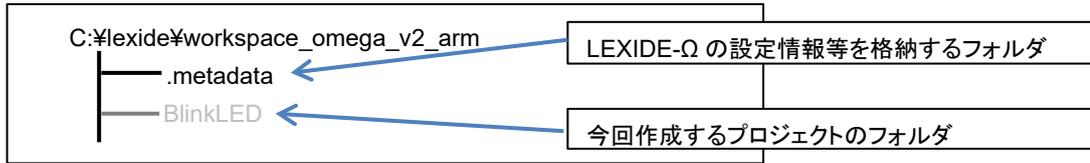


図 3-5. LEXIDE-Ω のワークスペース内のフォルダ構成

ワークスペースとは、LEXIDE-Ω の設定情報やプロジェクトなどを格納するための作業フォルダのことです。ワークスペースには、複数のプロジェクトを作成することができます。本チュートリアルでは、このワークスペース (C:\lexide\workspace_omega_v2_arm) の下に、プロジェクトおよびソースファイルのフォルダを作成していきます。

3.3 LEXIDE-Ω プロジェクト新規作成

プロジェクトを新規に作成します。

[File] > [New] > [Project] を選択し、[New Project] ダイアログを開きます。

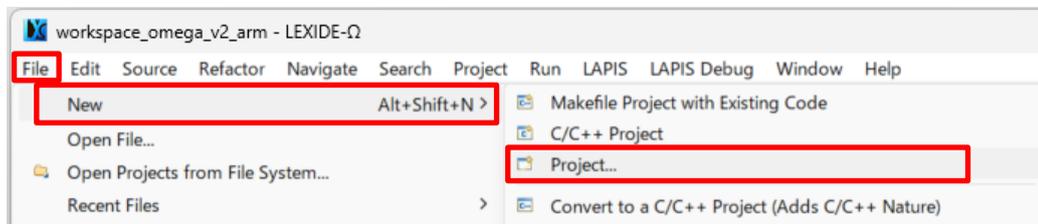


図 3-6. [New Project] ダイアログの表示方法

[New Project] ダイアログで、[C/C++] > [C Project] を選択し、[Next] ボタンをクリックします。

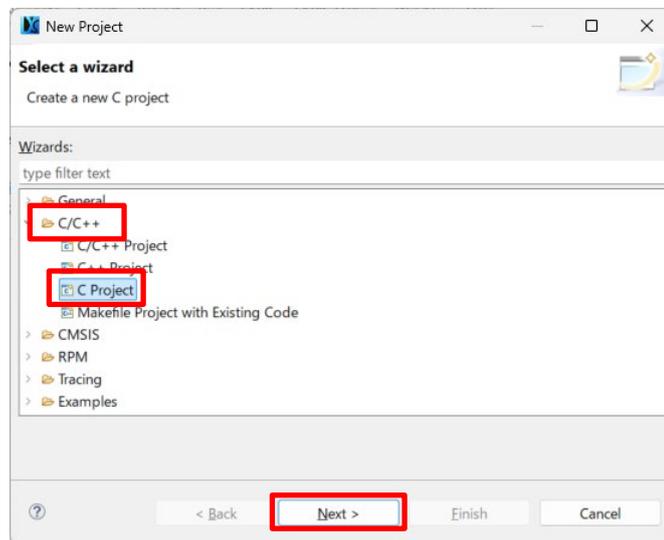


図 3-7. プロジェクトの選択

[C Project]ページにて、プロジェクト名、プロジェクトのタイプ、ツールチェーンなどを設定し、[Next]ボタンをクリックします。本チュートリアルでは、プロジェクト名を BlinkLED として新規に作成します。

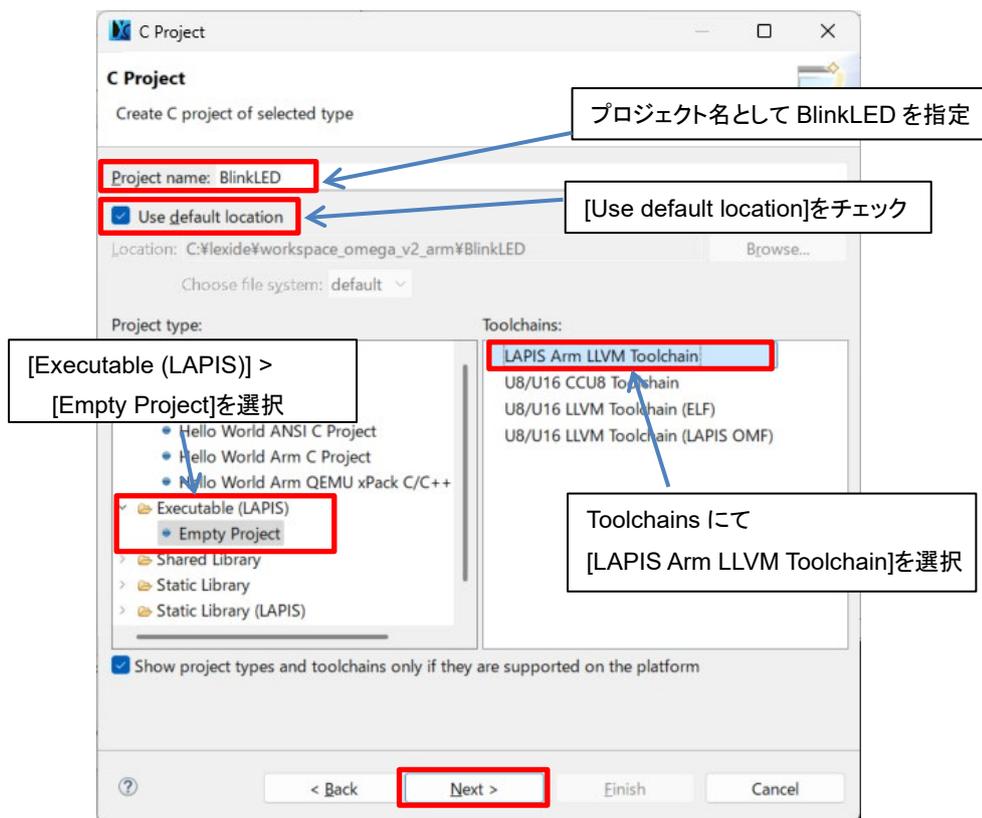


図 3-8. プロジェクトの選択

[Select Configurations]ページにて、ビルドコンフィグレーションを選択し、[Next]ボタンをクリックします。本チュートリアルでは、[Debug]を選択します。

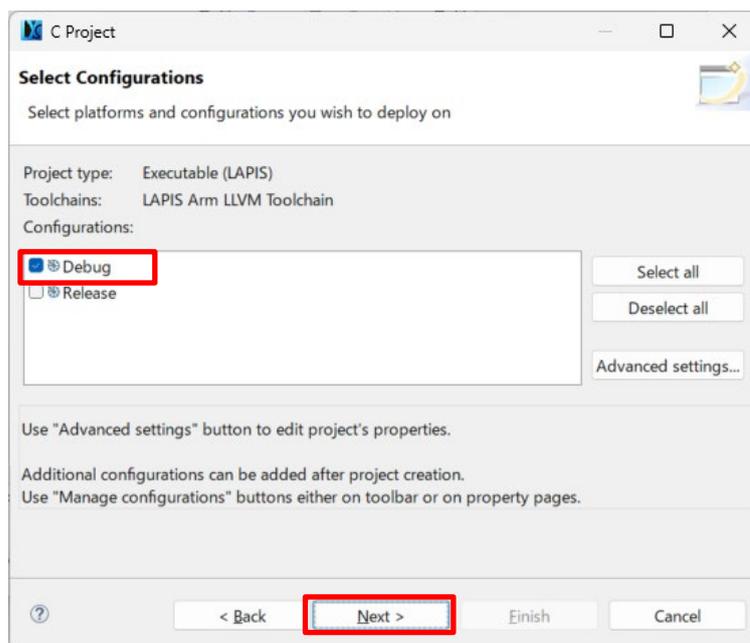


図 3-9. ビルドコンフィグレーション設定

[Select LAPIS Arm Device]ページで、ターゲット MCU の選択をします。
 本チュートリアルでは、[Family]で[ML63Q2500]、[Device]で ML63Q25x7 を選択します。

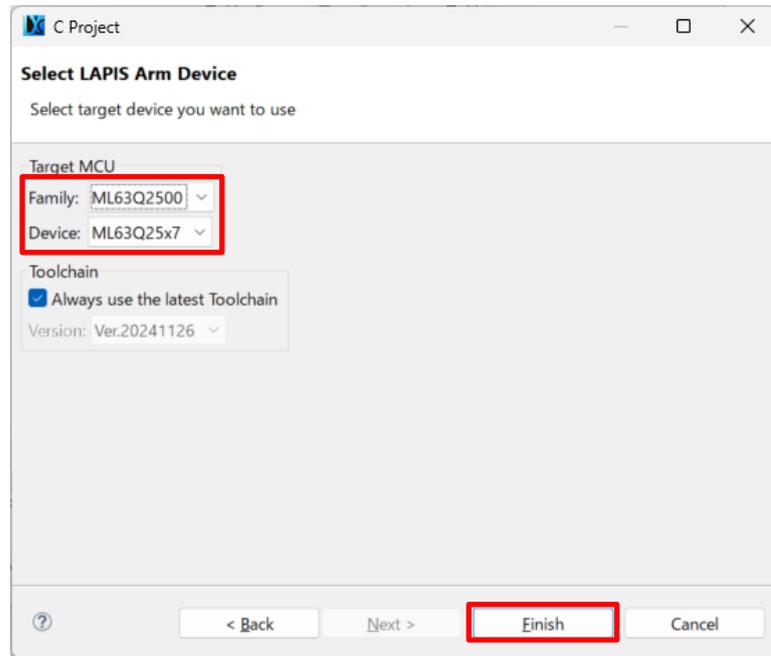


図 3-10. ターゲット MCU の選択

[Finish]ボタンをクリックすると、プロジェクトウィザードが終了し、プロジェクトが作成されます。

LEXIDE-Ωの[Project Explorer]上には、新規に作成したプロジェクト「BlinkLED」が表示され、ターゲット MCU のペリフェラルのレジスタを定義したヘッダファイル ML63Q25x7.h やスタートアップファイル startup_ML63Q25x7.c、リンク用のスクリプトファイルなどが登録された状態となっています。

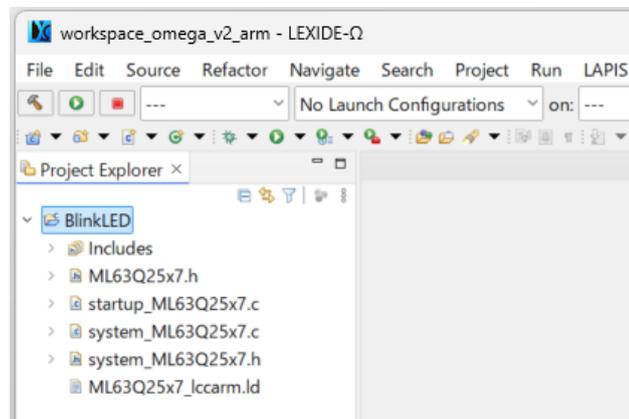


図 3-11. プロジェクト新規作成直後の様子

3.4 ソースファイルのプロジェクトへの登録

リファレンスソフトウェアに含まれている LED 点滅プログラム (BlinkLED) のソースファイルを LEXIDE-Ω に登録します。

リファレンスソフトウェアの各サンプルプログラムには、LEXIDE-Ω 用のプロジェクトが含まれていますが、バーチャルフォルダを使用しており、ファイルやフォルダの構成変更がしづらいというデメリットがあります。

そこで、本チュートリアルでは、サンプルプログラムのソースファイルを手動でプロジェクトへ登録していきます。

サンプルプログラムのプロジェクトは、ファイル構成の確認用として参照するのみとします。

リファレンスソフトウェアの BlinkLED サンプルプログラムのファイル構成は、以下のようになっています。

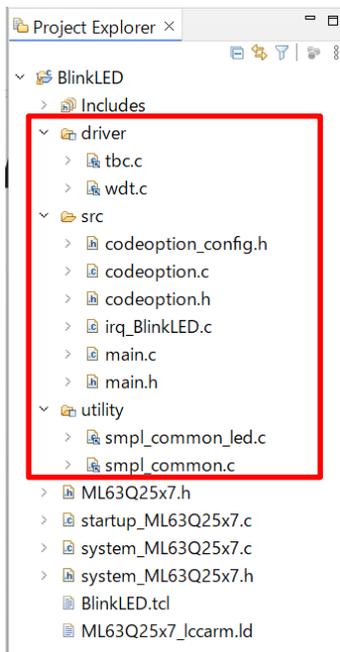


図 3-12. リファレンスソフトウェアの BlinkLED サンプルプログラムのファイル構成

本チュートリアルでは、上記のファイル構成を参考に、赤枠で囲まれた BlinkLED サンプルプログラムのソースファイルをプロジェクトに登録していきます。

リファレンスソフトウェアの SourceCode > samples > Common > BlinkLED の下の src フォルダを選択し、[Project Explorer]上のプロジェクト BlinkLED にドラッグ&ドロップします。

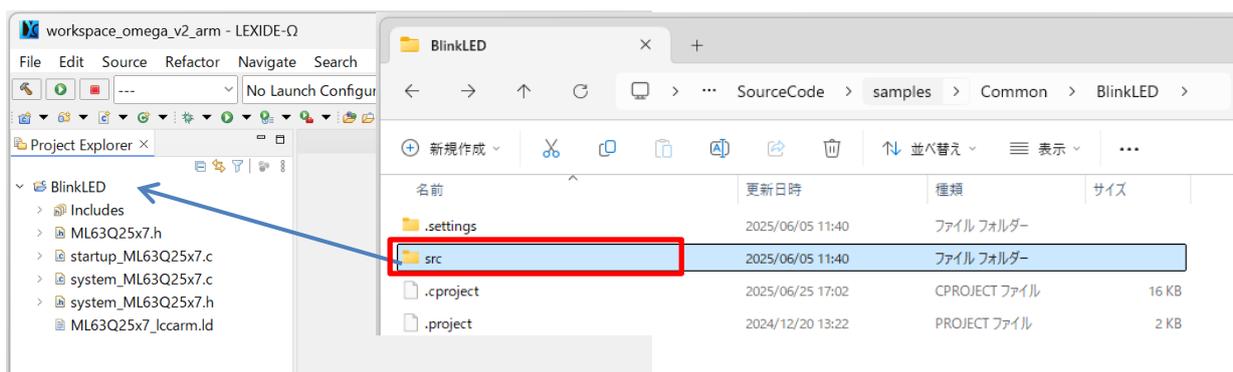


図 3-13. BlinkLED サンプルプログラムの src フォルダをプロジェクトへドラッグ&ドロップ

下記ダイアログが表示されたら、[Copy files and folders]を選択して[OK]ボタンをクリックします。

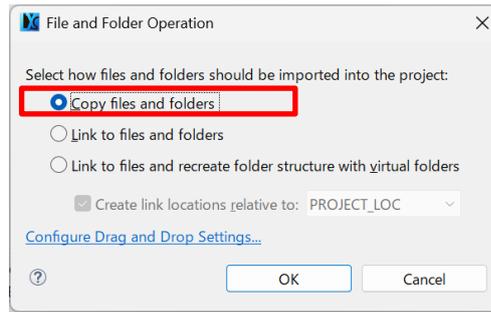


図 3-14. ファイルとフォルダの操作に関する確認ダイアログ

ドラッグ&ドロップされたフォルダおよびソースファイルがワークスペースのフォルダにコピーされ、[Project Explorer]のプロジェクトに表示されます。

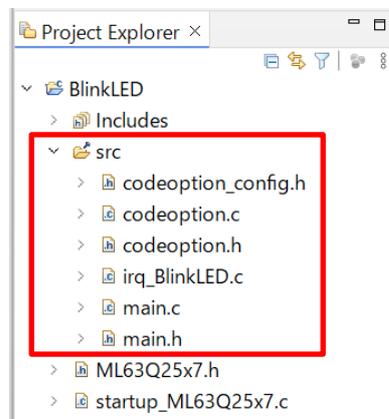


図 3-15. プロジェクトに登録されたフォルダおよびソースファイル

同様に、リファレンスソフトウェアの SourceCode 下の、driver フォルダ、および utility フォルダを選択して、[Project Explorer]上のプロジェクトにドラッグ&ドロップします。

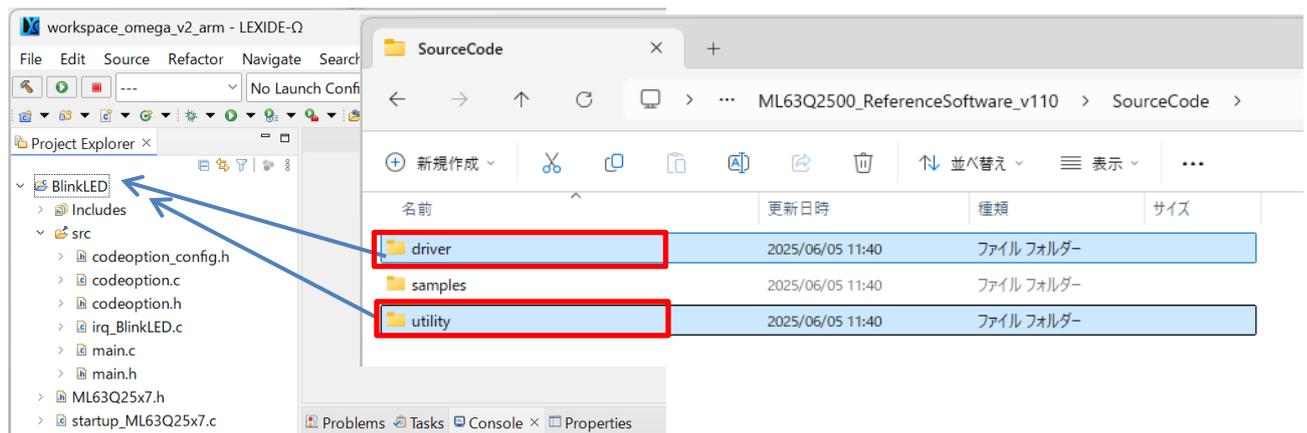


図 3-16. driver フォルダと utility フォルダをプロジェクトへドラッグ&ドロップ

下記ダイアログが表示されたら、[Copy files and folders]を選択して[OK]ボタンをクリックします。

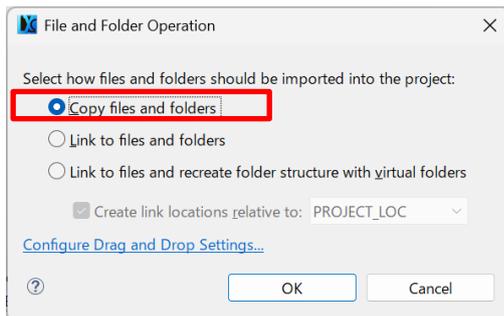


図 3-17. ファイルとフォルダの操作に関する確認ダイアログ

続いて、BlinkLED サンプルプログラムに不要なファイルをプロジェクトのビルド対象から外していきます。

「図 3-12. リファレンスソフトウェアの BlinkLED サンプルプログラムのファイル構成」を確認すると、使用されている driver フォルダ内のソースファイルは tbc.c と wdt.c, utility フォルダ内のソースファイルは smpl_common.c と smpl_common_led.c であることが分かります。

したがって、driver¥src フォルダ内の、tbc.c と wdt.c 以外のファイルをビルド対象から除外します。

driver¥lib フォルダはライブラリファイルを格納したフォルダで、プログラムから呼び出されない限りリンクされることはないため、そのままとします。

driver¥src フォルダ内の tbc.c と wdt.c 以外のファイルを、Ctrl キーを押しながら選択し、右クリックで表示されるポップアップメニューから [Resource Configurations] > [Exclude from Build...]を選択します。

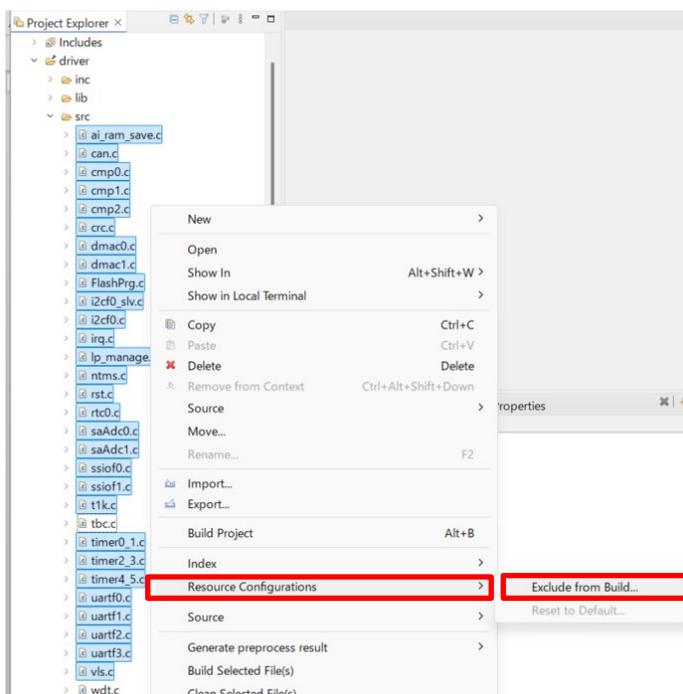


図 3-18. 不要なファイルをビルド対象から除外する

下記ダイアログが表示されたら、[Debug]をチェックして[OK]ボタンをクリックします。

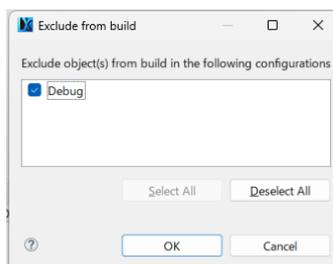


図 3-19.ビルド対象から除外する際の確認ダイアログ

以上により、選択したファイルがビルド対象から除外されます。
 ビルド対象から除外されたファイルは、グレースアウトされた状態で表示されます。

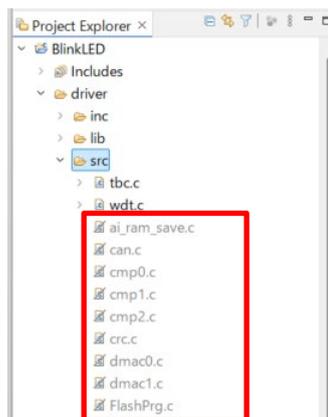


図 3-20.ビルド対象から除外されたファイル

以上で、プロジェクトへのファイル登録は完了です。

3.5 プログラム作成・編集

本チュートリアルで使用するサンプルプログラム（BlinkLED）は、一定時間ごとにリファレンスボード上の LED（P50, P51, P52）の点灯パターンを切り替えていく処理を繰り返します。
 点灯パターンの切り替えタイミングは、低速タイムベースカウンタ（TBC）割込みにより制御しています。
 本チュートリアルでは、処理を変更せずにそのまま使用します。

ソースファイルを編集する場合は、[Project Explorer]上のプロジェクトに表示されているソースファイル名をダブルクリックするとソースファイルが表示され、編集できるようになります。

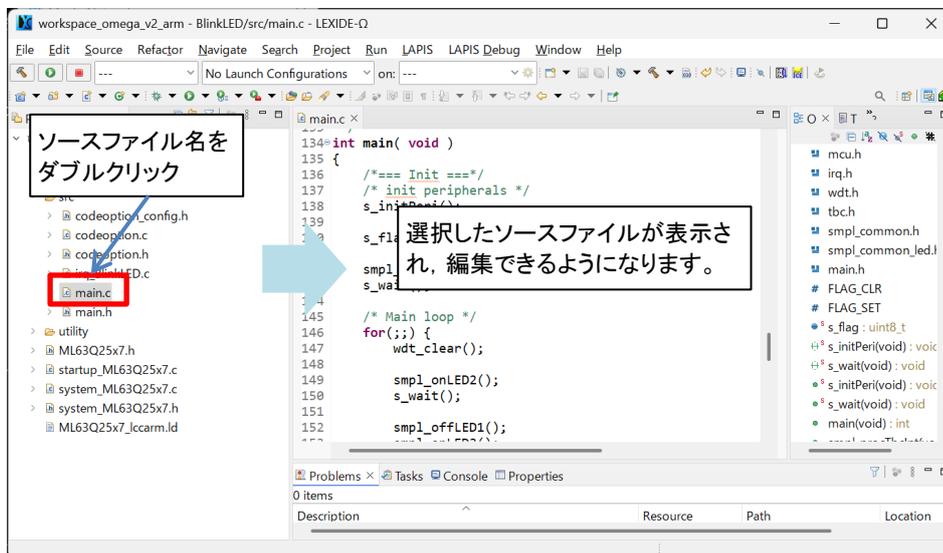


図 3-21.ソースファイルの表示方法

3.6 ビルドオプションの設定

インクルードパスやマクロ、最適化オプションなどのビルドオプションを設定します。
 ビルドオプションの設定は、[Properties]ダイアログ上で行います。
 [Properties]ダイアログは、[Project Explorer]のBlinkLED プロジェクトを選択後、[Project] > [Properties]を選択することで表示されます。

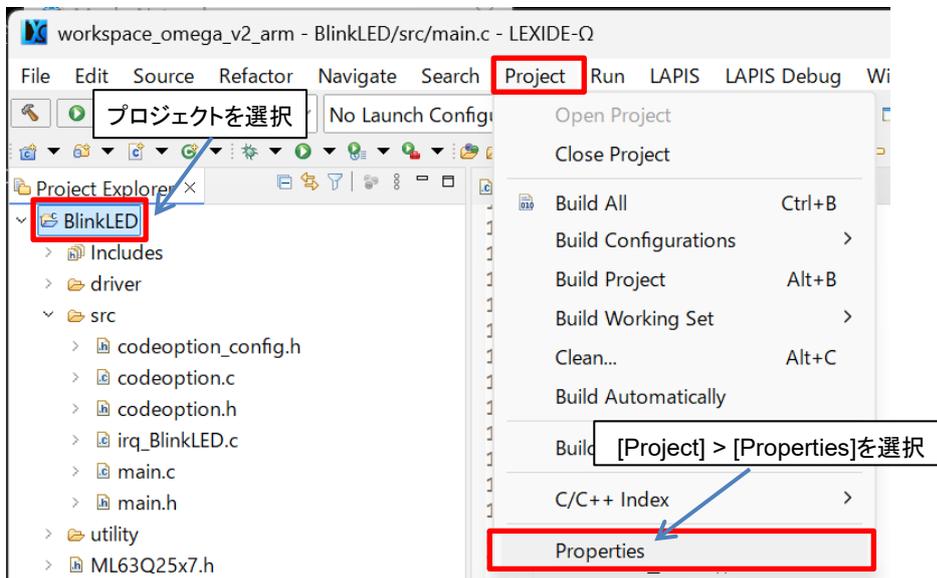


図 3-22.[Properties]ダイアログの表示方法

ビルドオプションは、下記[Properties]ダイアログの[C/C++ Build] > [Setting]を選択して設定します。

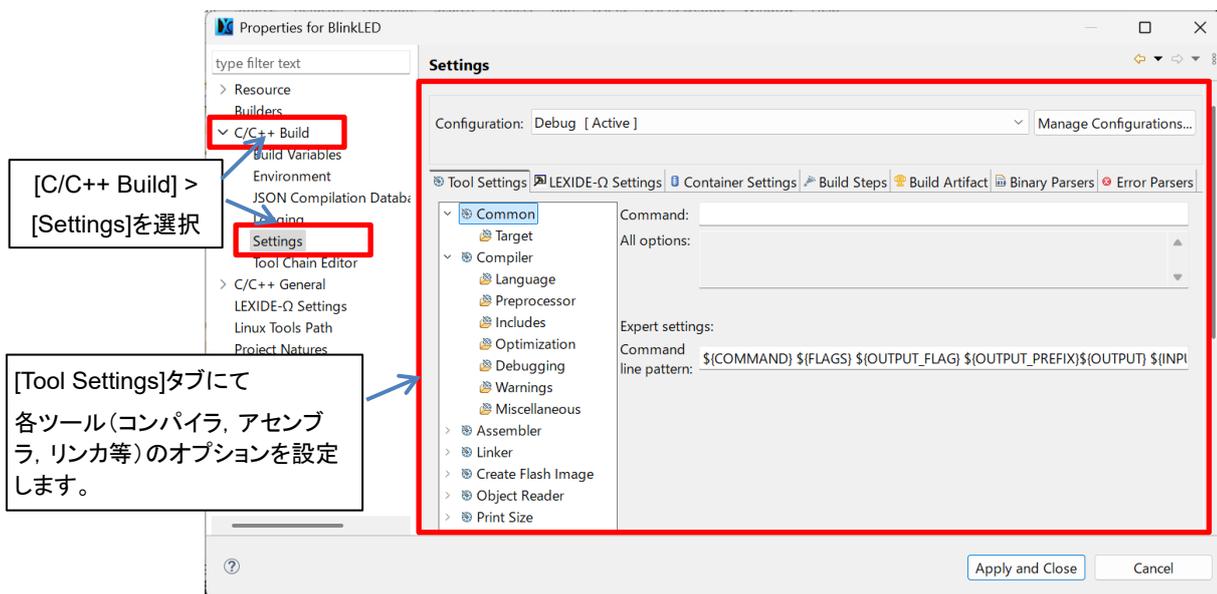


図 3-23.[Properties]ダイアログ ビルドオプションの設定画面

本チュートリアルでは、以下の設定を行います。

- マクロの設定
- インクルードパスの設定
- 最適化オプションの設定

3.6.1 マクロの設定

リファレンスソフトウェアには、マクロによって機種に応じて設定を切り替える部分が含まれています。このため、使用するマイコンに対応するマクロを設定します。

[Compiler] > [Preprocessor]を選択して[Add]ボタンをクリックし、表示されたダイアログにて、使用するマイコンに該当するマクロを指定します。本チュートリアルでは ML63Q2557 を指定しています。

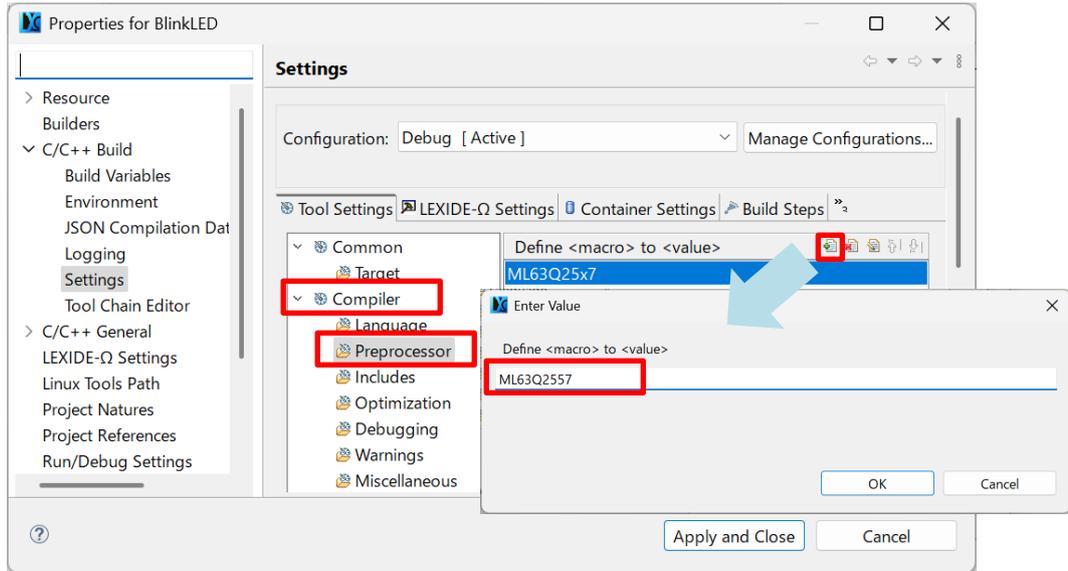


図 3-24.マクロの設定

3.6.2 インクルードパスの設定

インクルードファイル (*.h) は、BlinkLED の直下、driver\inc、src、utility\board フォルダに格納されています。
 [Compiler] > [Includes]を選択して[Add]ボタン  をクリックし、表示された[Add directory path]ダイアログの
 [Workspace]ボタンをクリックします。

表示された[Folder selection]ダイアログにて、「Ctrl」キーを押しながら、BlinkLED および driver > inc、src、utility > board を選択し、[OK]ボタンをクリックします。

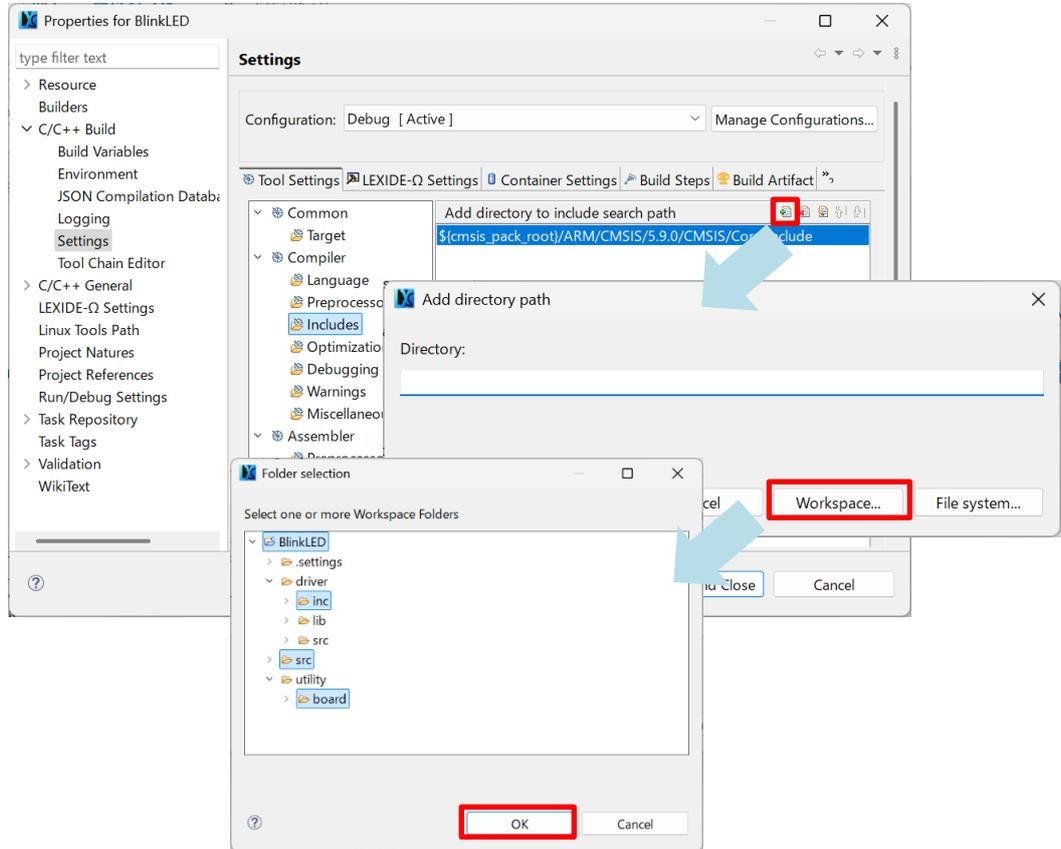


図 3-25.インクルードパスの設定

3.6.3 最適化オプションの設定

[Compiler] > [Optimization]にて、最適化オプションを設定します。
 本チュートリアルでは、デバッグに最適なコードを生成するため、None を選択します。
 すべての設定が終わったら、[Apply and Close]ボタンをクリックして設定を反映させます。

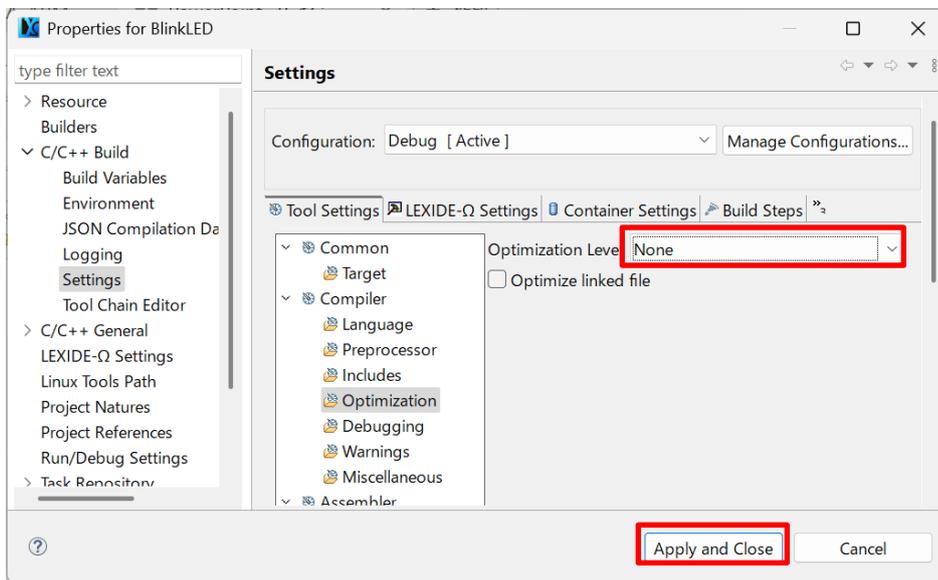


図 3-26.最適化オプションの設定

3.7 ビルド

「BlinkLED」プロジェクトを以下の手順でビルドします。

1. [Project Explorer]でプロジェクト（BlinkLED）を選択します。
2. ツールバーの  ボタンをクリックすると、ビルドが開始します。

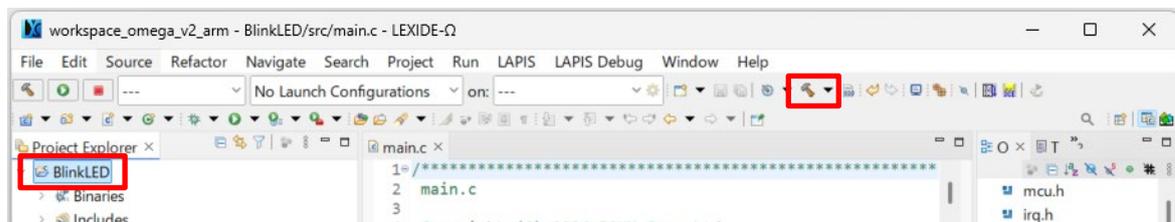


図 3-27.ビルド

ビルドが完了すると、[Console]タブに「Build Finished」と表示されます。

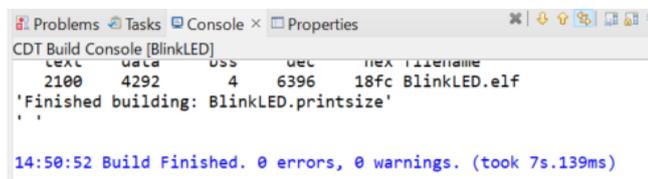


図 3-28.ビルド完了時のメッセージ

3.8 デバッグ

ここでは、LEXIDE-Ω上で操作するデバッガの使い方について説明します。
 デバッガの詳細については、『LEXIDE-Ω ARM コア搭載 LSI 向けユーザーズマニュアル』の「6章 Arm デバッグ機能」を参照してください。

3.8.1 事前準備

本チュートリアルでは、デバッグアダプタとして SEGGER J-Link を使用します。
 SEGGER J-Link を使用する場合は、あらかじめ J-Link Software and Documentation pack V7.62 以降をインストールしてください。
 インストール後、ROHM 製 LSI 用のフラッシュローダを J-Link へ認識させる必要があります。
 以下に手順を示します。

- ① 使用するフラッシュローダモジュールファイルを準備します。「2.3 機種情報ファイルのインストール」にて機種情報ファイルをインストールすると、通常はユーザーアカウントの以下のフォルダにフラッシュローダが格納されているのを確認することができます。
 C:\%Users%\<ユーザー名>\%AppData%\Local%\Arm%\Packs%\<ベンダ名>\<ROHM 製 LSI 名>\<バージョン>\%Flash%\<ROHM 製 LSI フラッシュローダファイル名>.FLM



図 3-29.ローム製 LSI(ML63Q25x7)フラッシュローダの格納場所

- ② 以下のフォルダを作成し、①のフラッシュローダモジュールファイルをコピーします。
 C:\%Users%\<ユーザー名>\%AppData%\Roaming%\SEGGER%\JlinkDevices%\Rohm%\ML63Q25x7%
- ③ 同じフォルダに、追加したデバイスを J-Link が認識するためのファイル（本チュートリアルでは ML63Q25x7.xml）を作成します。

```

¥---JLinkDevices
  +---Rohm
    +---ML63Q25x7
      |           ML63Q25x7.FLM
      |           ML63Q25x7.xml
      ...
    
```

- ④ 上記③で作成した xml ファイルに以下の内容を記載します。ML63Q25x7 以外の場合は、お使いの LSI の仕様に合わせて変更してください。
 ※赤枠で囲った行は改行を入れずに 1 行で記載してください。

```

<DataBase>
<!-- -->
<!-- Rohm -->
<!-- -->
<Device>
<ChipInfo Vendor="ROHM" Name="ML63Q25x7" Core="JLINK_CORE_CORTEX_M0"
  WorkRAMAddr="0x20000000" WorkRAMSize="0x02000"/>
<FlashBankInfo Name="Internal Flash" BaseAddr="0x10000000" MaxSize="0x00040000"
  Loader="ML63Q25x7.FLM" LoaderType="FLASH_ALGO_TYPE_CMSIS" AlwaysPresent="1"/>
</Device>
</DataBase>
    
```

3.8.2 デバッグコンフィグレーションの作成

デバッグコンフィグレーションに、GDB SEGGER J-Link Debugging を選択してください。
 [Project Explorer]上でプロジェクトを選択して右クリックで表示されるポップアップメニューから[Debug As] > [Debug Configurations...]を選択し、 [Debug Configurations]ダイアログを表示します。

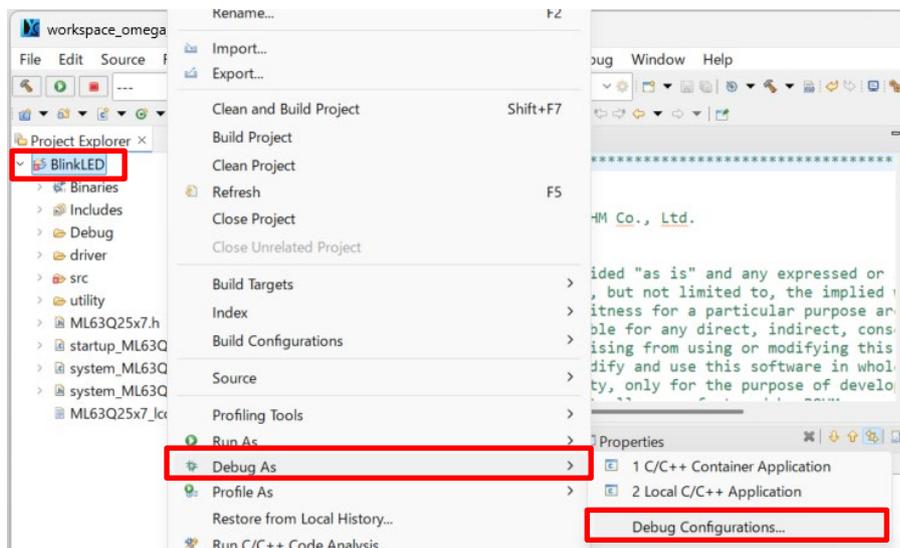


図 3-30.[Debug Configuration]ダイアログの表示方法

表示された[Debug Configuration]ダイアログにて、 [GDB SEGGER J-Link Debugging]を選択し、 [New launch configuration]ボタン  をクリックすると、 デバッグコンフィグレーションが作成されます。
 デバッグコンフィグレーションの名前は、 デフォルトでは<プロジェクト名>+Debug となりますが、 変更可能です。
 本チュートリアルでは J-Link 用のデバッグコンフィグレーションであることを明確にするため、 BlinkLED Debug J-Link としています。

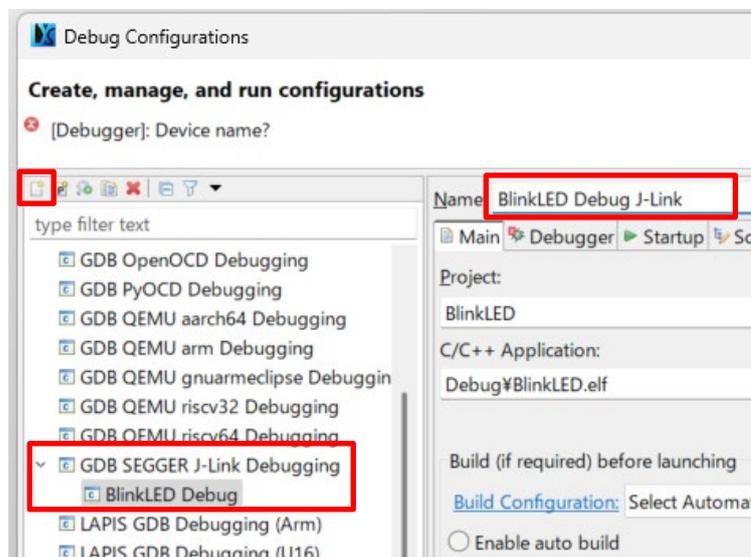


図 3-31.デバッグコンフィグレーションの新規作成

デバッグコンフィグレーションを作成した後、追加で設定する必要がある項目について、以下に説明します。

[Debugger]タブの[Device name]にデバイス名 ML63Q25x7 を設定します。

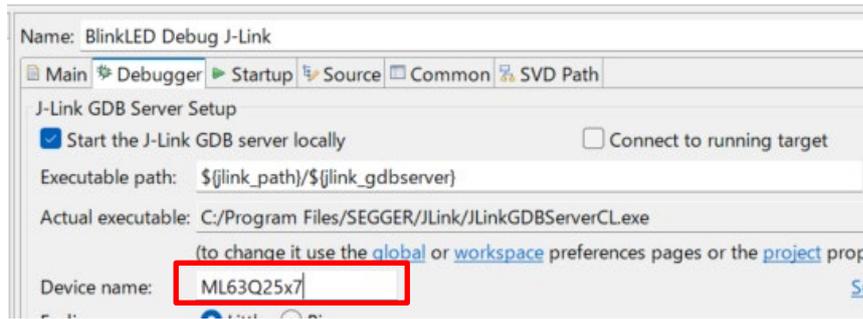


図 3-32.デバイス名の設定

[Debugger]タブの[Executable name]に GDB クライアントの実行ファイル名を設定します。[Browse]ボタンをクリックして表示される[Select GDB Client Binary]ダイアログにて、LAPIS > LEXIDE > gdb の下にある arm-none-eabi-gdb.exe を選択し、[開く]ボタンをクリックします。

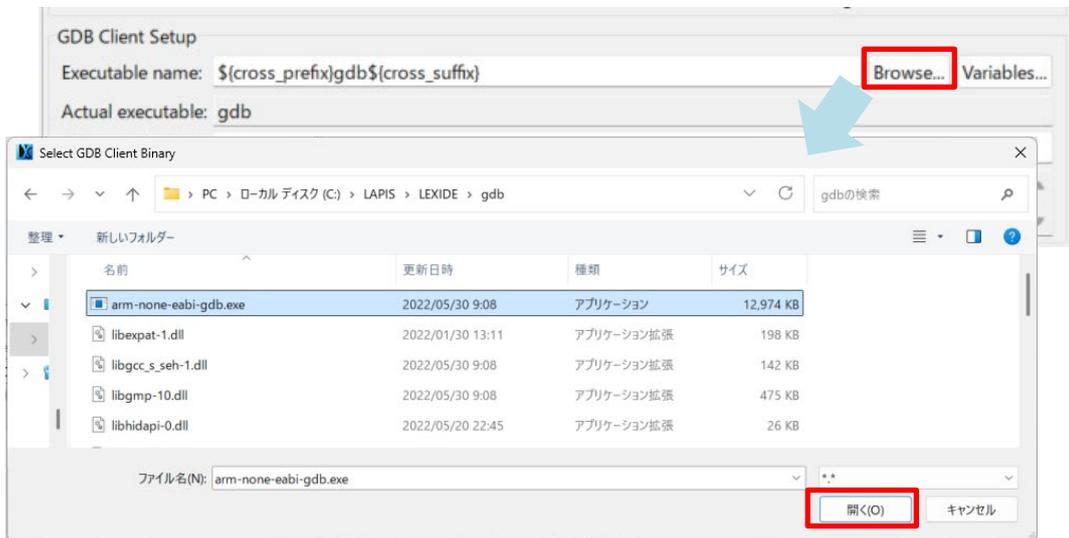


図 3-33.GDB クライアントの実行ファイル名の設定

[SVD Path]タブの[File path]に SVD ファイルのパスを設定します。SVD ファイルのパスを設定することで、デバッグ上でペリフェラルの各レジスタを参照できるようになります。
 [Browse]ボタンをクリックして表示される[Select SVD File]ダイアログにて、機種情報ファイルがインストールされているフォルダの下にある SVD ファイル（本チュートリアルでは ML63Q25x7.svd）を選択し、[開く]ボタンをクリックします。

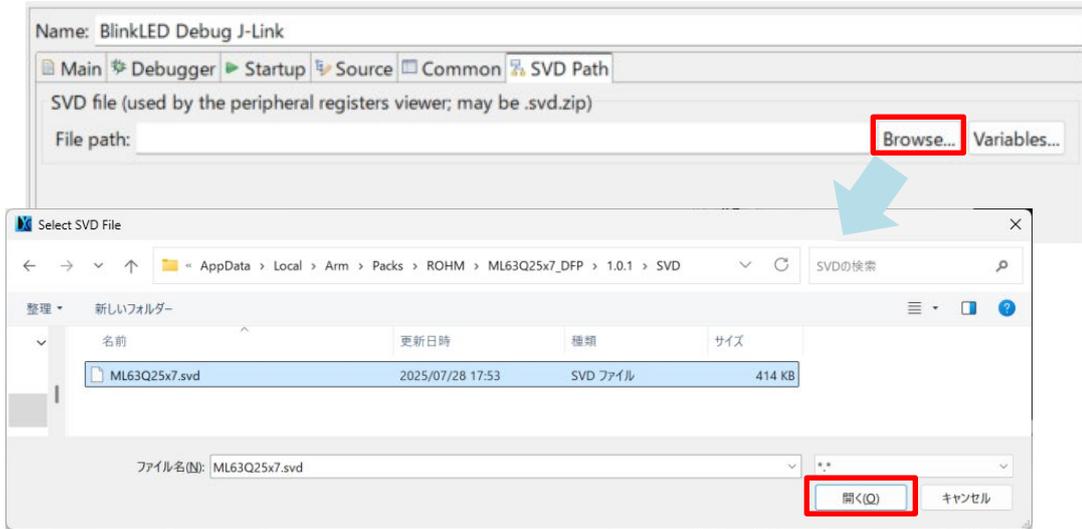


図 3-34.SVD ファイルのパス設定

他の設定項目は基本的にデフォルトの設定で問題ありません。
 [Close]ボタンをクリックして[Debug Configurations]ダイアログを閉じます。

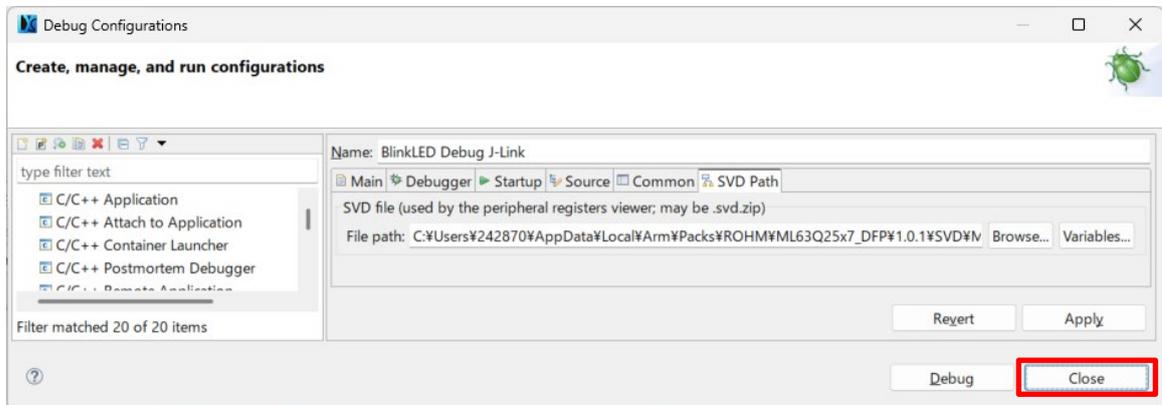


図 3-35.[Debug Configuration]ダイアログのクローズ

以下のダイアログが表示されたら[Save]ボタンをクリックして設定内容を保存します。



図 3-36.[Debug Configuration]クローズ時の確認ダイアログ

3.8.3 デバッグの開始

以下のように PC と J-Link, リファレンスボード, USB to SPI 変換モジュールを接続します。

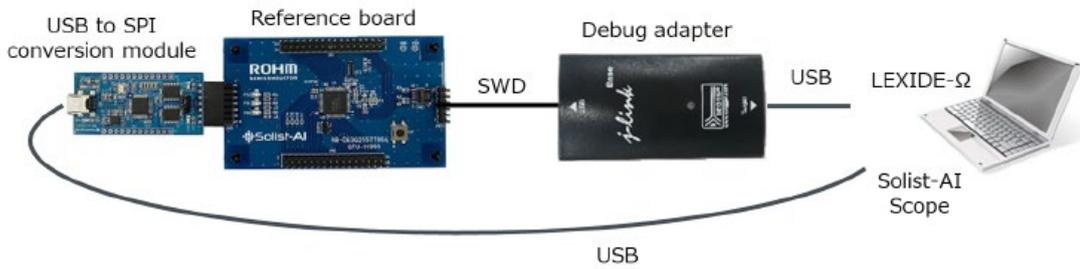


図 3-37.デバッグ時の接続

デバッグ開始時には, [Launch Configuration]でデバッグコンフィグレーションを選択, [Launch Mode]で[Debug]を選択し, [Launch in 'Debug' mode]ボタン  をクリックします。

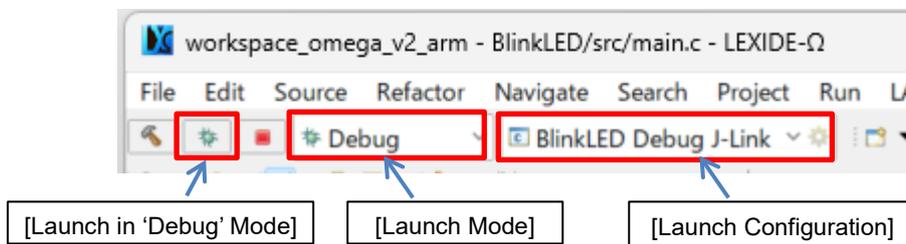


図 3-38.デバッグの開始

パースペクティブの切り替えを確認するダイアログが表示されたら, [Switch]ボタンをクリックします。

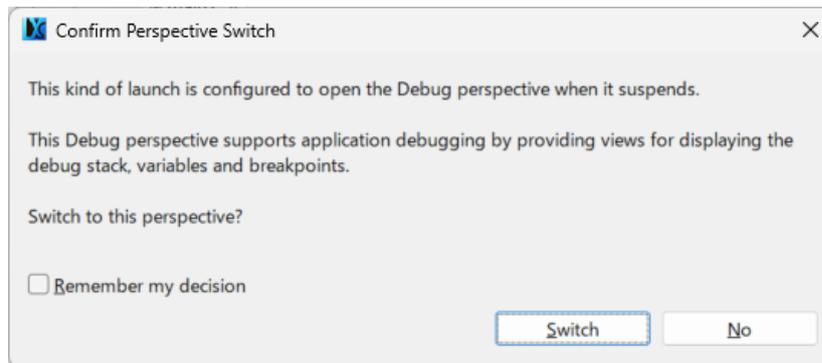


図 3-39.パースペクティブ切り替えの確認ダイアログ

LEXIDE-Ωの画面がデバッグ用のパースペクティブ（以降[Debug]パースペクティブ）に切り替わります。プログラムは main 関数の先頭まで実行した状態でブレークします。

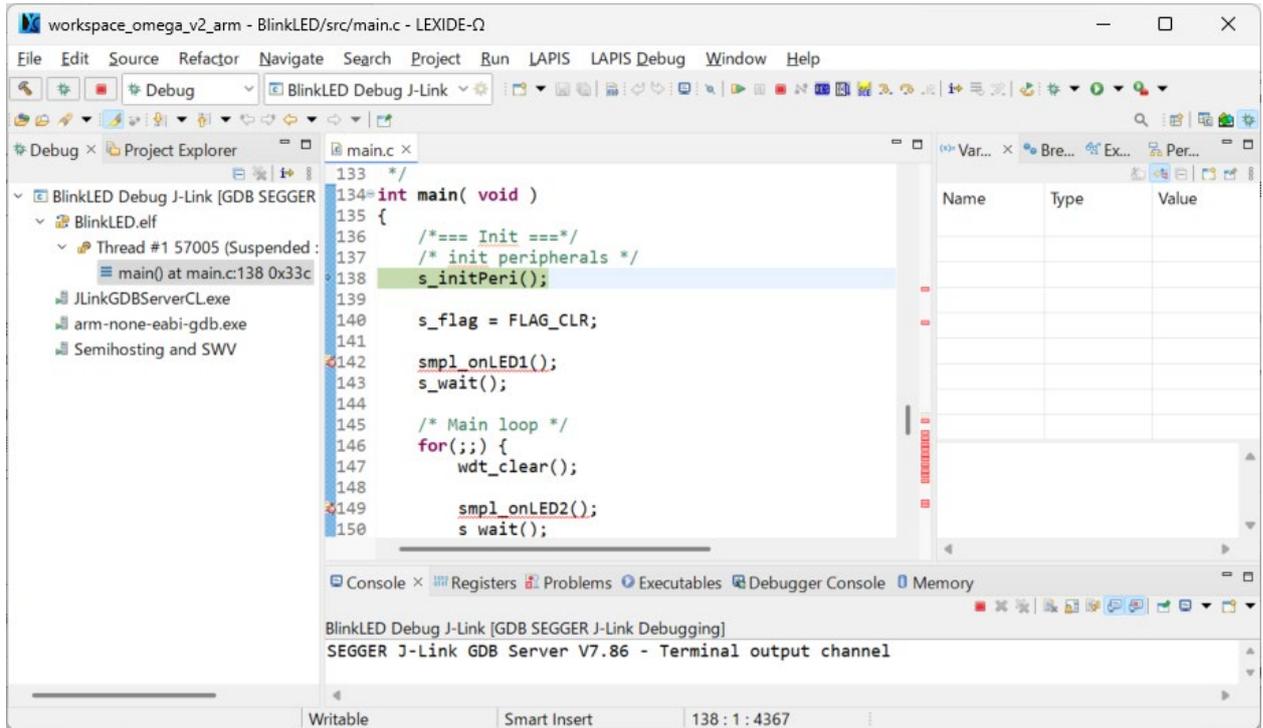


図 3-40.デバッグ時の画面

3.8.4 プログラムの実行開始・再開

ツールバーの[Resume (F8)]ボタンをクリックします。プログラムが実行され、リファレンスボード上のLED（P50, P51, P52）の点灯パターンが切り替わっていくのを確認できます。



図 3-41.プログラムの実行開始・再開

3.8.5 プログラムの実行停止

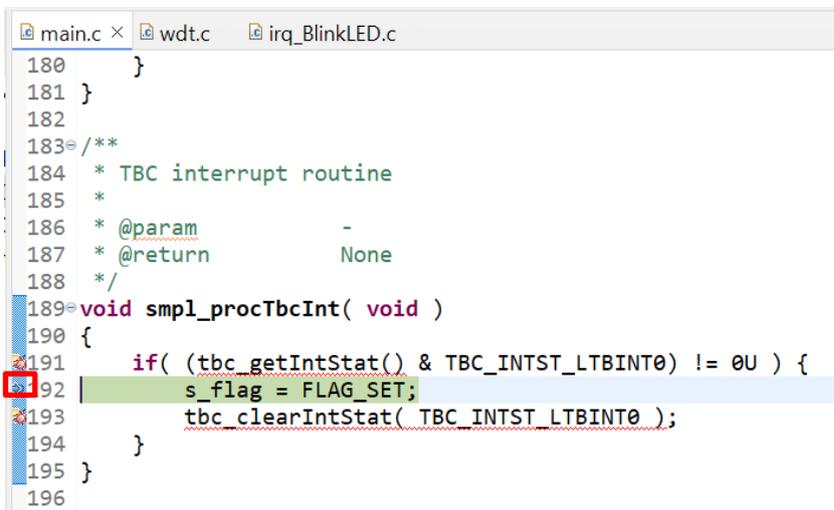
ツールバーの[Suspend]ボタンをクリックすると、プログラムの実行が停止します。



図 3-42.プログラムの実行停止

3.8.6 ブレークポイントの設定

プログラムを特定の場所で止めたいとき、ブレークポイントを設定します。
 ソース行の左端をダブルクリックするたびにブレークポイントの設定／解除ができます。
 ブレークポイントが設定されると、ソース行の左端にチェック付きの小さな丸いアイコン  が表示されます。
 ここでは、main.c の TBC 割込み関数 `smp1_procTbcInt()` にブレークポイントを設定します。
 [Resume (F8)] ボタンをクリックするたびに、ブレークポイントを設定した行でブレークし、LED の点灯パターンが変わることを確認できます。



```

main.c × wdt.c irq_BlinkLED.c
180     }
181 }
182
183 /**
184  * TBC interrupt routine
185  *
186  * @param      -
187  * @return     None
188  */
189 void smp1_procTbcInt( void )
190 {
191     if ( tbc_getIntStat() & TBC_INTST_LTBINT0 ) != 0U ) {
192         s_flag = FLAG_SET;
193         tbc_clearIntStat( TBC_INTST_LTBINT0 );
194     }
195 }
196
    
```

図 3-43.ブレークポイントの設定

3.8.7 デバッグの終了

デバッグを終了させるには、ツールバー上の[Stop]ボタン  をクリックします。

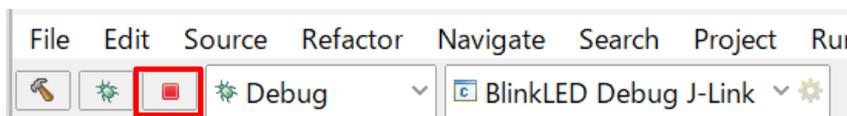


図 3-44.デバッグの終了

デバッグ前のレイアウトに戻す場合には、ツールバー上の右側にある[C/C++]パースペクティブボタン  をクリックして、表示を切り替えてください。

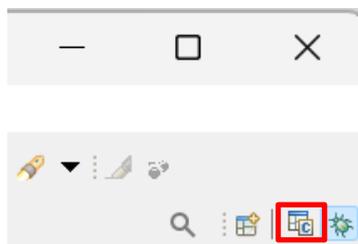


図 3-45. [C/C++]パースペクティブへの切り替え

4. 【参考】Solist-AI™のファームウェア開発について

ローム WEB サイトの Solist-AI™の各製品ページに、アプリケーションノート『加速度センサを使った異常検知と AI ライブラリ使用方法』とサンプルプログラム『加速度センサを使った異常検知の AI ライブラリ』、および『ML63Q2500 シリーズ リファレンスソフトウェア』を掲載しています^(*)。

Solist-AI™のファームウェア開発時の参考としてください。

*1 : ローム WEB サイトの下記ページよりダウンロード可能です。
<https://www.rohm.co.jp/products/micon/solist-ai/ml63q2500-group>

アプリケーションノート

Title	Revision	Description	Last Updated
NEW 加速度センサを使った異常検知とAIライブラリ使用方法	1	本ドキュメントは、加速度センサを使った異常検知のAIライブラリの使用方法について記載しています。	2025/04/18

サンプルプログラム

Title	Revision	Description	Last Updated
NEW ML63Q2500シリーズ リファレンスソフトウェア	1.2	ML63Q2500グループ用のペリフェラルドライバと、リファレンスボード上で動作するサンプルプログラムです。LEXIDE-Qで使用できます。	2025/10/28
NEW Solist-AI™ Scope (リアルタイム波形表示ツール)	1.2.11	Solist-AI™の効果を確認するリアルタイム波形表示ツールです。使用する場合はSolist-AI™マイコンとSolist-AI™ Scopeを接続するアダプタ (MM-FT232HCなど) が必要です。お客様にてご用意ください。	2025/10/28
NEW 加速度センサを使った異常検知のAIライブラリ	1	加速度センサを使った異常検知のAIライブラリの使用方法の添付データです。	2025/04/18

図 4-1. Solist-AI™の各製品ページ内のアプリケーションノートとサンプルプログラム (抜粋)

4.1 サンプルプログラムの制御フロー

『加速度センサを使った異常検知と AI ライブラリ使用方法』(以降、アプリケーションノート)は、加速度センサを使った異常検知の AI ライブラリの使用方法について記載しています。このアプリケーションノートに対応するサンプルプログラムが『加速度センサを使った異常検知と AI ライブラリ』(以降、AnomalyDetectionDemo サンプルプログラム)に含まれています。AnomalyDetectionDemo サンプルプログラムの制御フローは以下のようになっています。

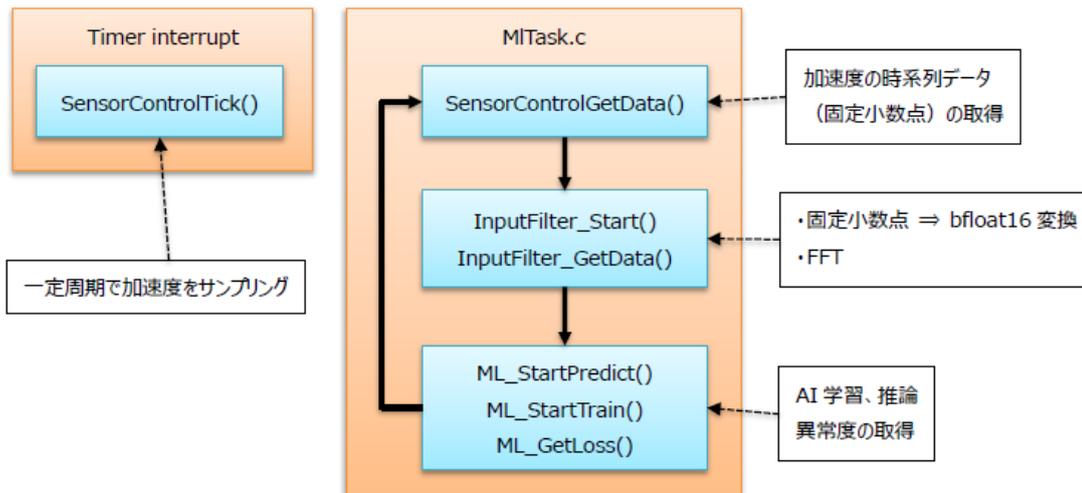


図 4-2. AnomalyDetectionDemo サンプルプログラムの制御フロー (アプリケーションノートからの抜粋)

AnomalyDetectionDemo サンプルプログラムには、初期学習、推論、および逐次学習が実装されています。それぞれのフローは以下のようになっています。

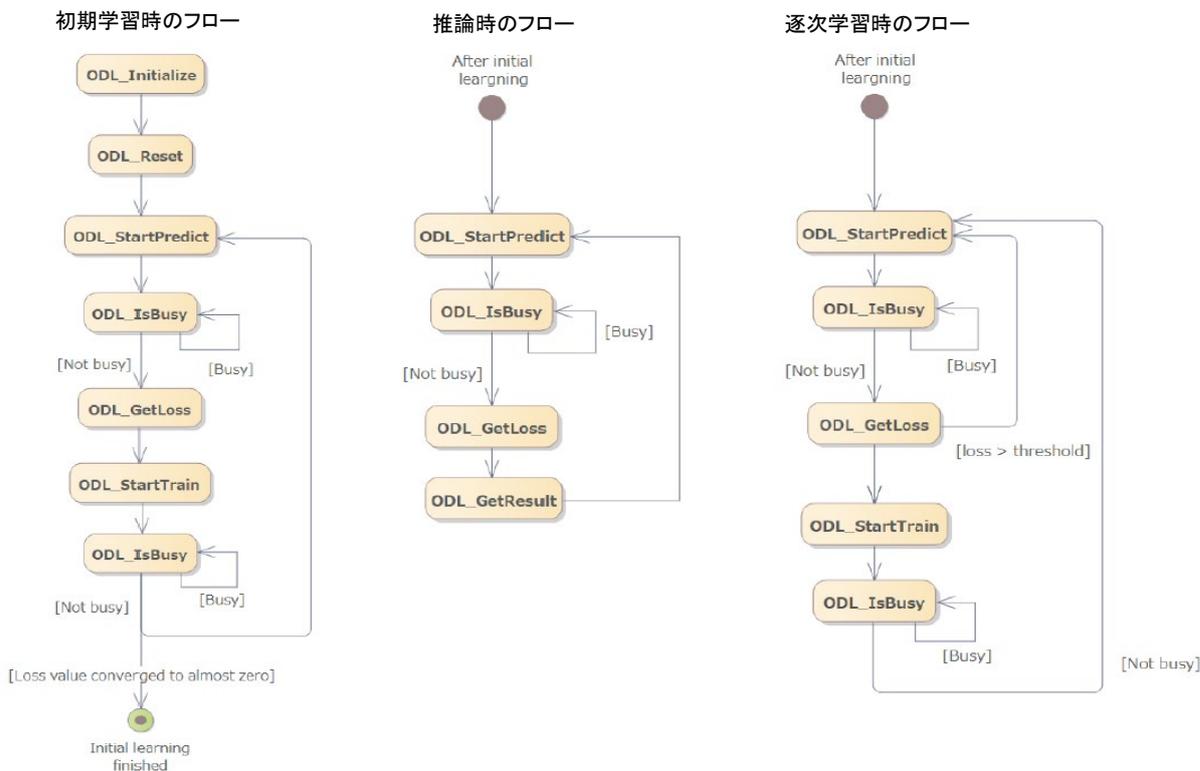


図 4-3. 初期学習、推論、逐次学習の制御フロー（アプリケーションノートからの抜粋）

AnomalyDetectionDemo サンプルプログラムで使用している Solist-AI™のライブラリの API につきましては、アプリケーションノートの「3. Solist-AI™ library」を参照してください。併せて、『ML63Q2500 シリーズ リファレンスソフトウェア』（以降、リファレンスソフトウェア）のスタートアップマニュアル（FJXT63Q2500_REFSOFT_STARTUP-xx.pdf）（以降、スタートアップマニュアル）の「2.1. API 一覧」、およびリファレンスソフトウェアの Document フォルダにある FEX63Q2500_REFSOFT_DRIVER_API-xx.chm を参照してください。

スタートアップマニュアルの「3.2. Solist-AI サンプルプログラム」および SourceCode¥Samples¥Solist-AI 内の該当フォルダのソースコードも併せて参照してください。

4.2 サンプルプログラムの読み込み（インポート）方法

『加速度センサを使った異常検知の AI ライブラリ』および『ML63Q2500 シリーズ リファレンスソフトウェア』のサンプルプログラムには、LEXIDE-Ω用のプロジェクトファイルが含まれており、LEXIDE-Ωで読み込むことが可能です。

『加速度センサを使った異常検知の AI ライブラリ』のサンプルプログラムを例として、以下にプロジェクトの読み込み手順を示します。

- ① LEXIDE-Ωの[File] > [Import]より表示されるダイアログにて、[General] > [Existing Projects into Workspace]を選択し、[Next]ボタンをクリックします。
- ② [Select root directory]でサンプルプログラムのプロジェクトファイルが格納されているフォルダ（Solist-AI_FW_AnomalyDetectionSample¥samples¥Solist-AI¥AnomalyDetectionDemo）を指定して[Finish]ボタンをクリックすることで、プロジェクトを読み込むことができます。

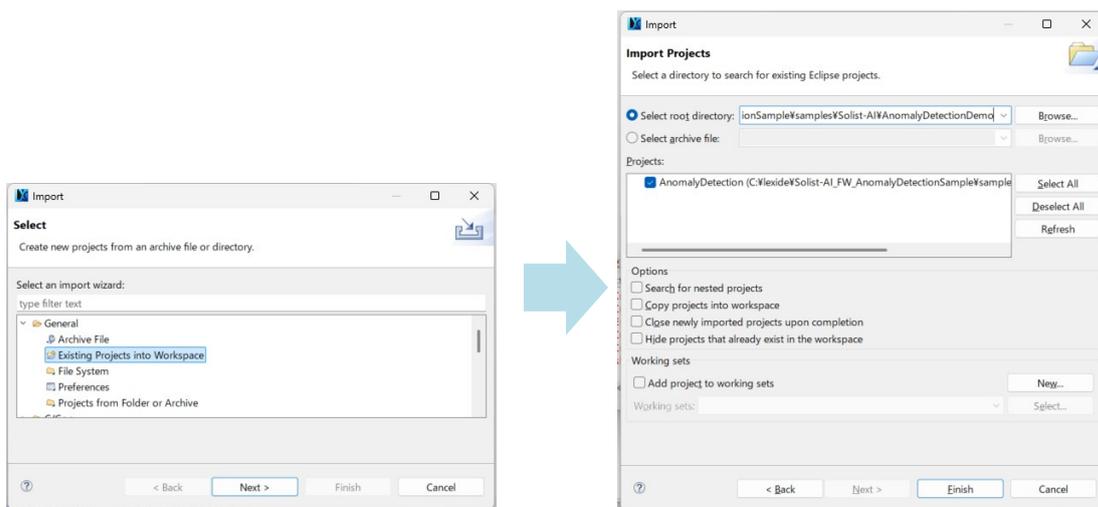


図 4-4. サンプルプログラムのプロジェクト読み込み（インポート）

【注意】

サンプルプログラムは、加速度センサ KXTJ3-1057 を対象として作成しています。KXTJ3-1057 以外のセンサを使用している場合、または、センサを接続していない状態の場合、Solist-AI™ Scope が起動できません。Solist-AI™ Scope は、センサから正常にデータを受信できる状態で起動してください。

KXTJ3-1057 以外のセンサを使用する場合は、お使いのセンサに合わせてプログラムを変更していただく必要があります。センサ関連のソースファイルは Solist-AI_FW_AnomalyDetectionSample¥sensor に格納されています。これらを参考に実装してください。

KXTJ3-1057 用のソースファイルは、kxtj3.c です。ソースファイルに定義されている各関数の概要を以下に示します。

関数名	説明
int8_t Kxtj3Init(bool addr_pin)	センサデバイスを初期化する
int8_t Kxtj3Start(void)	センサデバイスの動作を開始する
int8_t Kxtj3Stop(void)	センサデバイスの動作を停止する
int8_t Kxtj3RawGetVal(int16_t *raw)	センサからデータを取得する
int8_t Kxtj3GetVal(int16_t *acc)	センサからの取得データをスケールリングする
static int8_t InitCheckVal(void)	WHO_AM_I レジスタの内容を取得しデバイス ID が正しいことをチェックする
static int8_t InitSettingIc(uint8_t *checkval)	センサデバイスを設定する
static int8_t InitSettingGval(uint8_t checkval, int16_t *gval)	センサデバイスの加速度レンジを設定する
static int8_t Write(uint8_t reg, uint8_t *data, uint16_t size)	センサデバイスへの書き込み
static int8_t Read(uint8_t reg, uint8_t *data, uint16_t size)	センサデバイスからの読み出し

4.3 LEXIDE Scope による波形表示

CMSIS-DAP 準拠のデバッグアダプタをお使いの場合、LEXIDE Scope を使ってプログラム実行中に SWD 経由で RAM や SFR の値を読み出し、リアルタイムに値やグラフを表示することが可能となっており、LEXIDE Scope を Solist-AI™ Scope の代用のツールとしてお使いいただくことができます。

なお、LEXIDE Scope は MM-FT232H のような USB SPI 変換モジュールを用いることなく使用することが可能です。LEXIDE Scope を使用する場合の開発環境の構成を以下に示します。

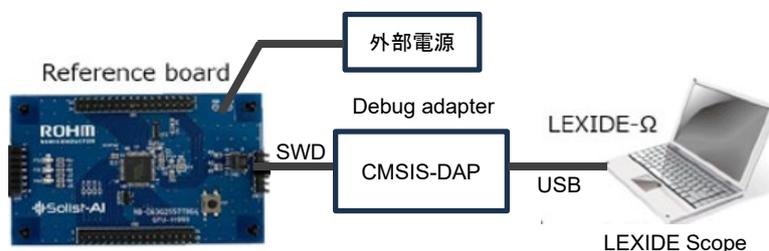


図 4-5. LEXIDE Scope 使用時の開発環境の構成

LEXIDE Scope の使い方の詳細については、『LEXIDE-Ω ARM コア搭載 LSI 向け ユーザーズマニュアル』の「6.9.1 LEXIDE Scope」を参照してください。

4.3.1 デバッグコンフィグレーション

CMSIS-DAP デバッグアダプタを使用する場合、LEXIDE-Ω のデバッグコンフィグレーションに[LAPIS GDB Debugging (Arm)]を選択する必要があります。

[Debug Configuration]ダイアログにて、[LAPIS GDB Debugging (Arm)]を選択し、[New launch configuration]ボタンをクリックすると、デバッグコンフィグレーションが作成されます。

デバッグコンフィグレーションの名前は、デフォルトでは<プロジェクト名>+Debug となりますが、変更可能です。本チュートリアルでは CMSIS-DAP 用のデバッグコンフィグレーションであることを明確にするため、AnomalyDetection Debug CMSIS-DAP としています。

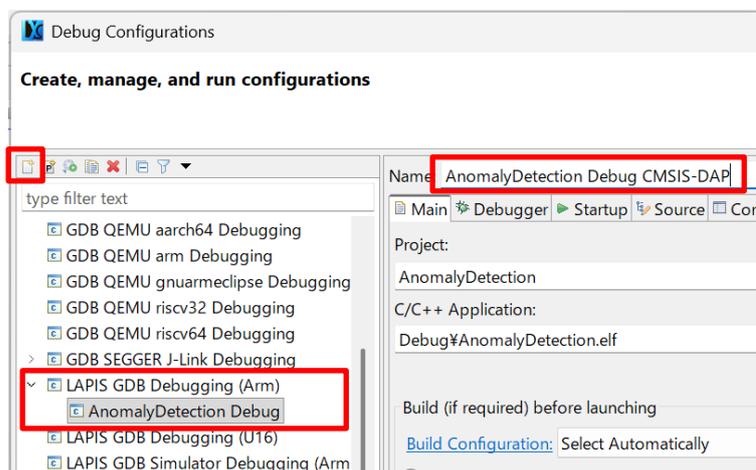


図 4-6. CMSIS-DAP 用デバッグコンフィグレーションの新規作成

[Debugger]タブの[Target ICE]が[CMSIS-DAP]になっていることを確認し、下方の[LEXIDE Scope Setup]の[Launch LEXIDE Scope]をチェックします。
他の設定については、基本的にデフォルトの設定で問題ありません。

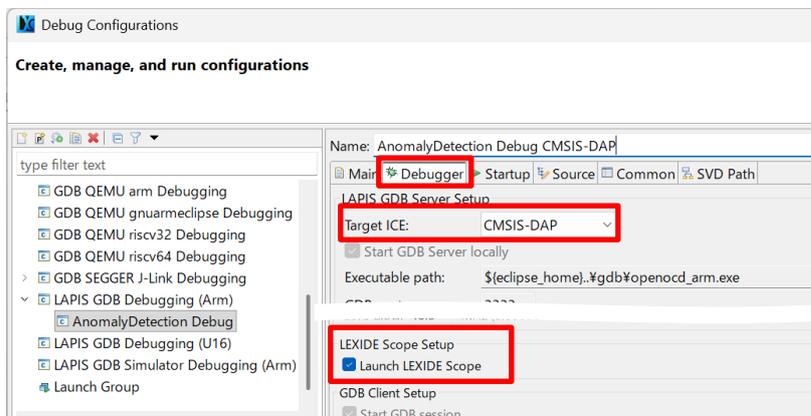


図 4-7.LEXIDE Scope の起動設定

4.3.2 LEXIDE Scope の起動

LEXIDE Scope は LEXIDE-Ω から起動します。
最初に、デバッグを開始します。
[Launch Configuration]でデバッグコンフィグレーション（本チュートリアルでは AnomalyDetection Debug CMSIS-DAP）を選択し、[Launch Mode]で[Debug]を選択し、[Launch in 'Debug' mode]ボタン  をクリックします。

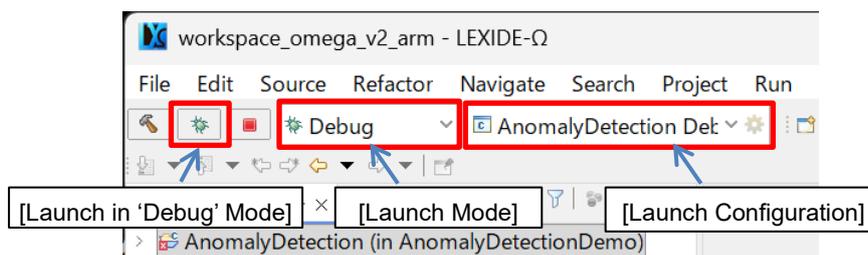


図 4-8.デバッグの開始

デバッグが開始しブレイクしたら、LEXIDE-Ωの[LAPIS Debug] > [Launch LEXIDE Scope]を選択すると LEXIDE Scope が起動します。

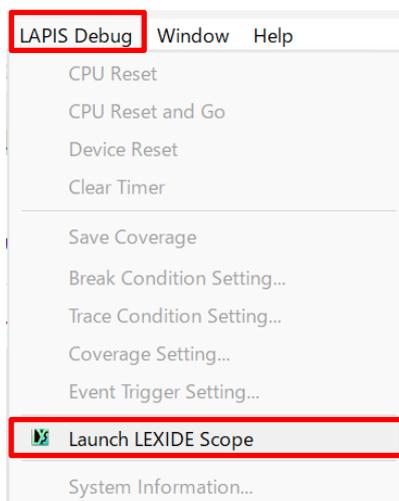


図 4-9.LEXIDE Scope の起動



図 4-10.LEXIDE Scope 起動時の画面(メインウィンドウ)

4.3.3 LEXIDE Scope の使い方

LEXIDE Scope の基本的な使い方の手順を以下に示します。

手順 1 : モードの設定

CMSIS-DAP を使う場合, LEXIDE Scope の[Settings] > [Mode Settings]を選択し, モードを[Universal Mode]に設定します。

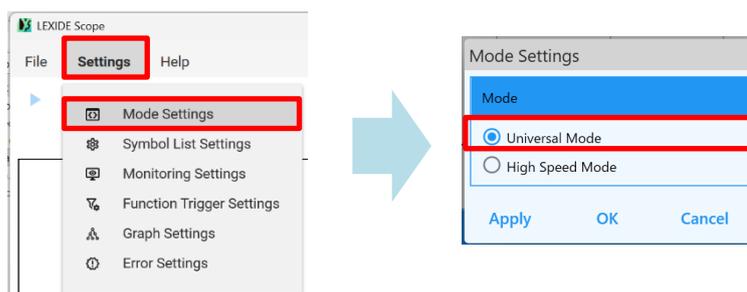


図 4-11.LEXIDE Scope のモード設定

手順 2 : 測定ポイントの追加

メインウィンドウの[Measurement Point]の[Add]をクリックし, 表示された[Add Measurement Point]ダイアログにて, 測定ポイントを選択し, [Add]をクリックして追加します。

本チュートリアルでは, i2AccOutX_, i2AccOutY_, i2AccOutZ_, output_loss を追加します。追加が完了したら[Close]をクリックしてダイアログを閉じます。

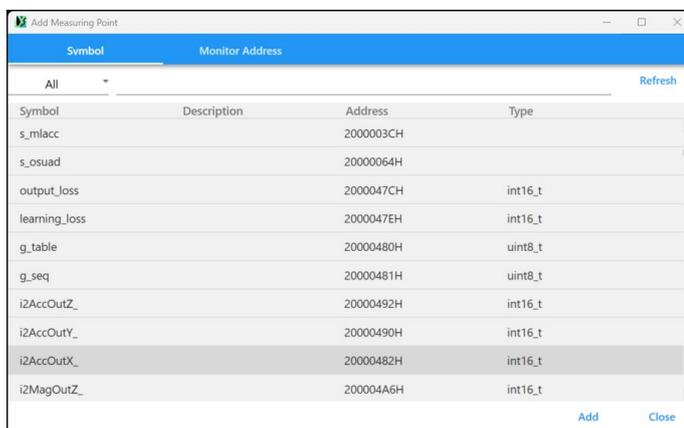


図 4-12.測定ポイントの追加

手順 3 : モニタの開始

メインウィンドウの ▶ をクリックしモニタを開始します。

続けて LEXIDE-Ω のツールバー上の[Resume (F8)]ボタン ▶ をクリックして実行を再開します。

以上により、リアルタイムに波形が変化する様子を確認することができます。

必要に応じて[Measurement Point]の[Y-Resolution]および[Y-Offset]を調整してください。



図 4-13.実行結果

改版履歴

ドキュメント No.	発行日	ページ		変更内容
		改版前	改版後	
FJXT63Q2500_LEXIDE_TUTORIAL-01	2025.12.26	—	—	正式初版発行

ご 注 意

- 1) 本資料に記載されている内容は、ロームグループ(以下「ローム」という)製品のご紹介を目的としています。ローム製品のご使用にあたりましては、別途最新のデータシートもしくは仕様書を必ずご確認ください。
- 2) ローム製品は、一般的な電子機器(AV機器、OA機器、通信機器、家電製品、アミューズメント機器等)もしくはデータシートに明示した用途への使用を意図して設計・製造されています。したがって、極めて高度な信頼性が要求され、その故障や誤動作が人の生命、身体への危険もしくは損害、またはその他の重大な損害の発生に関わるような機器または装置(医療機器、輸送機器、交通機器、航空宇宙機器、原子力制御装置、燃料制御、カーアクセサリーを含む車載機器、各種安全装置等)(以下「特定用途」という)にローム製品のご使用を検討される際は事前にローム営業窓口までご相談くださいますようお願いいたします。ロームの文書による事前の承諾を得ることなく、特定用途にローム製品を使用したことによりお客様または第三者に生じた損害等に関し、ロームは一切その責任を負いません。
- 3) 半導体を含む電子部品は、一定の確率で誤動作や故障が生じる場合があります。万が一、誤動作や故障が生じた場合であっても、人の生命、身体、財産への危険または損害が生じないように、お客様の責任においてフェールセーフ設計など安全対策をお願いいたします。
- 4) 本資料に記載された応用回路例やその定数などの情報は、ローム製品の標準的な動作や使い方を説明するためのもので、実際に使用する機器での動作を明示的にも黙示的にも保証するものではありません。したがって、お客様の機器の設計において、回路やその定数及びこれらに関連する情報を使用する場合には、外部諸条件を考慮し、お客様の判断と責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、ロームは一切その責任を負いません。
- 5) ローム製品及び本資料に記載の技術を輸出または国外へ提供する際には、「外国為替及び外国貿易法」、「米国輸出管理規則」など適用される輸出関連法令を遵守し、それらの定めにしたがって必要な手続きを行ってください。
- 6) 本資料に記載された応用回路例などの技術情報及び諸データは、あくまでも一例を示すものであり、これらに関する第三者の知的財産権及びその他の権利について権利侵害がないことを保証するものではありません。また、ロームは、本資料に記載された情報について、ロームもしくは第三者が所有または管理している知的財産権その他の権利の実施、使用または利用を、明示的にも黙示的にも、お客様に許諾するものではありません。
- 7) 本資料の全部または一部をロームの文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
- 8) 本資料に記載の内容は、本資料発行時点のものであり、予告なく変更することがあります。ローム製品のご購入及びご使用に際しては、事前にローム営業窓口で最新の情報をご確認ください。
- 9) ロームは本資料に記載されている情報に誤りがないことを保証するものではありません。万が一、本資料に記載された情報の誤りによりお客様または第三者に損害が生じた場合においても、ロームは一切その責任を負いません。



ローム製品のご検討ありがとうございます。
より詳しい資料やカタログなどをご用意しておりますので、お問い合わせください。

ROHM Customer Support System

<https://www.rohm.co.jp/contactus>