

FVL/LNX Ver4. xx
ユーザアプリケーション
リモートデバッグに関する技術情報

☆第2版☆

(株)ファースト

本書の内容は、予告なしに変更することがあります。内容の変更について、(株)ファーストはいかなる責任も負いません。本書あるいは関連ソフトウェアにおける誤りから生じる損害について、(株)ファーストはいかなる責任も負いません。

本書の内容の一部または全部を転載することは固くお断りします。

御注意

- ◎Microsoft, Windows 7, Windows 10 は、米国 Microsoft Corporation の登録商標です。
- ◎Eclipse は、Eclipse Foundation 登録商標です。
- ◎その他、文中における会社名、商品名は各社の登録商標または商標です。
- ◎文中では 商標シンボル((R)、(TM)、(C) 等)の表記は省略致します。

1 . はじめにお読みください	1
1.1 はじめに	1
1.2 動作環境について	1
1.3 ソフトウェアのダウンロードについて	1
2 . インストールについて	2
2.1 OpenJDK Runtimes のインストール	2
2.2 Eclipse IDE のインストール	2
3 . デバッグ手順について	7
3.1 アプリケーションのビルド	7
3.2 Eclipse の起動	8
3.3 プロジェクトの作成	9
3.4 デバッグの設定	10
3.4.1 Debug Configurations - Main タブ	12
3.4.2 Debug Configurations - Debugger - Main タブ	14
3.4.3 Debug Configurations - Debugger - Shared Libraries タブ	15
3.4.4 Debug Configurations - Debugger - Connection タブ	16
3.4.5 Debug Configurations - Source タブ	18
3.5 デバッグの実行	20
4 . 操作編	22

1. はじめにお読みください

1.1 はじめに

本書は、FVL/LNX Ver4.00以降で作成されたユーザアプリケーションについてリモートデバッグを実現する技術情報を提供することを目的としています。

使用するツールはEclipse Foundationによって提供されるEclipseです。なお、当社にてEclipseの動作についてお客様のすべての環境で動作することを保証するものではありません。あくまでリモートデバッグを実現するための情報提供の範囲にとどまり、使用方法や不具合等を含んだ技術的なお問い合わせにも対応できないことを予めご了承ください。

なお、2022年1月現在、FV1400-LNXシリーズでは動作しません。

1.2 動作環境について

64bit Windows環境が必要です。

弊社では、以下の組み合わせにて動作テストを行っております。

- ・Windows 10 Pro 64bit (21H2)
- ・Eclipse IDE 2021-12 (※)
- ・OpenJDK Runtimes Temurin 17(LTS) (※)

(※)2022年1月末時点でのバージョン

1.3 ソフトウェアのダウンロードについて

デバッグ環境の導入においては、以下のソフトウェアが必要となります。以下のサイトから事前にダウンロードする必要があります。

<https://www.eclipse.org/downloads/>

- ① Eclipse IDE 2021-12
- ② OpenJDK Runtimes (Temurin 17)

2. インストールについて

インストールの手順について説明します。以下の手順に従ってインストールを行ってください。

2.1 OpenJDK Runtimes のインストール

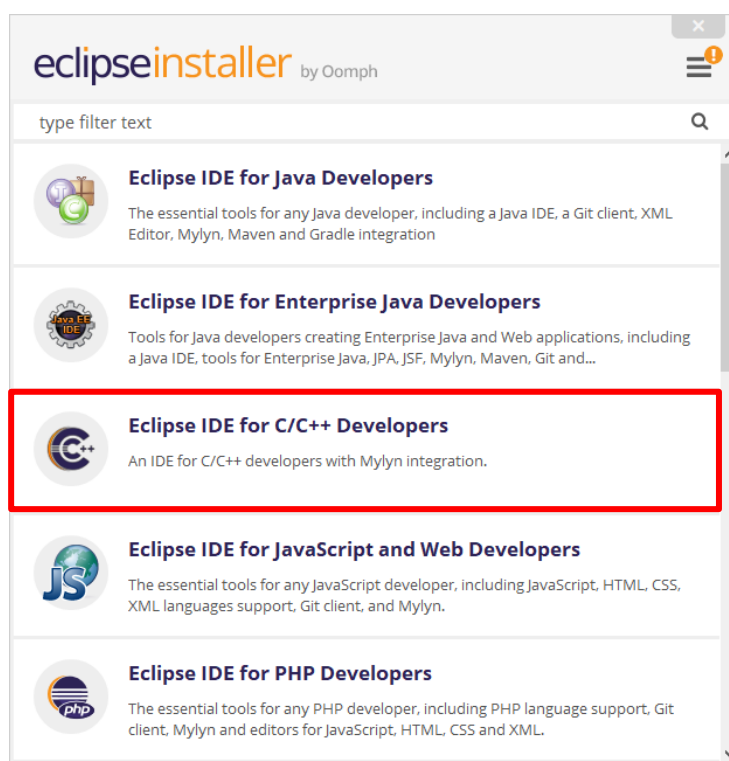
Eclipse IDE のインストールに先立ち、OpenJDK Temurin 17(以下 JDK と表記)のインストールが必要です。ダウンロードしたインストーラーを管理者権限で実行し、インストールを行ってください。

2.2 Eclipse IDE のインストール

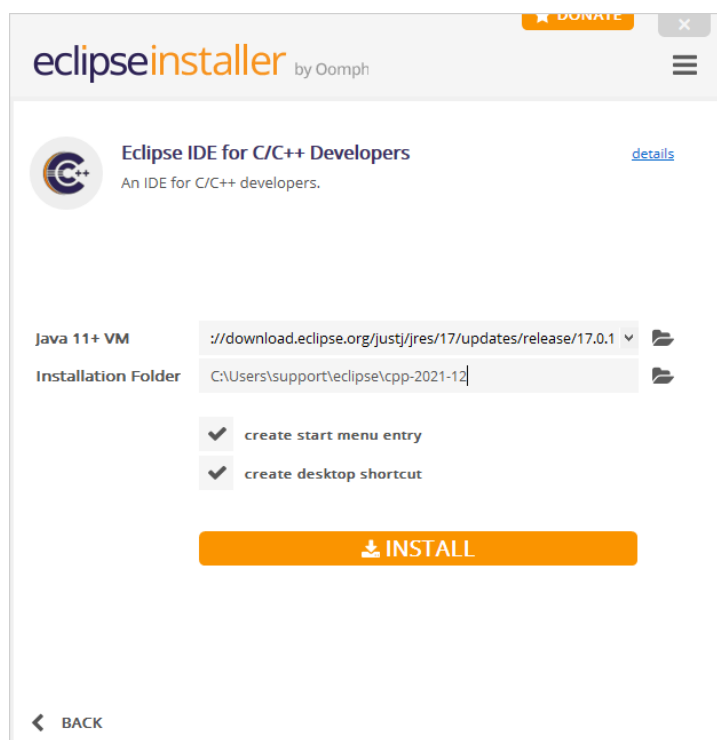
ダウンロードした Eclipse のインストーラーを管理者権限で実行します。以下のように表示され、インストーラーが起動しますのでしばらく待ちます。



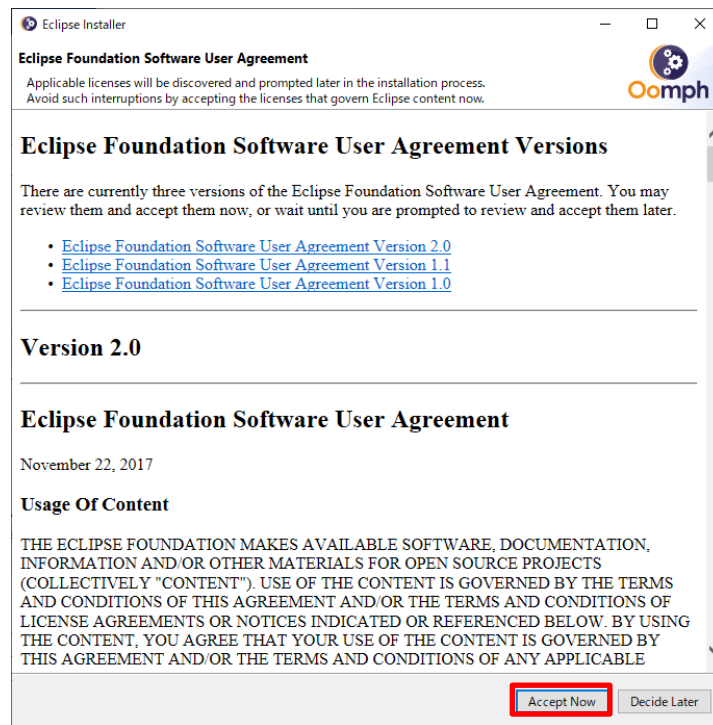
しばらく待つと以下のように表示されます。



ここでは、「Eclipse IDE for C/C++ Developers」をクリックします。
以下のように表示されますので、「INSTALL」をクリックします。



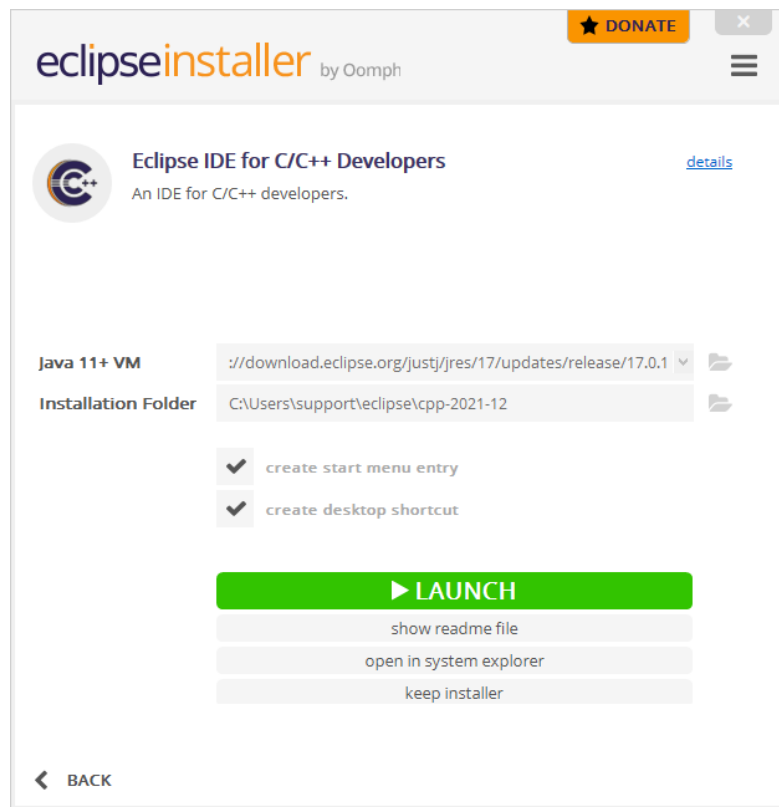
「INSTALL」をクリックすると以下の表示になりますので、「Accept Now」をクリックしてください。



以下の表示になり、インストールが進行します。



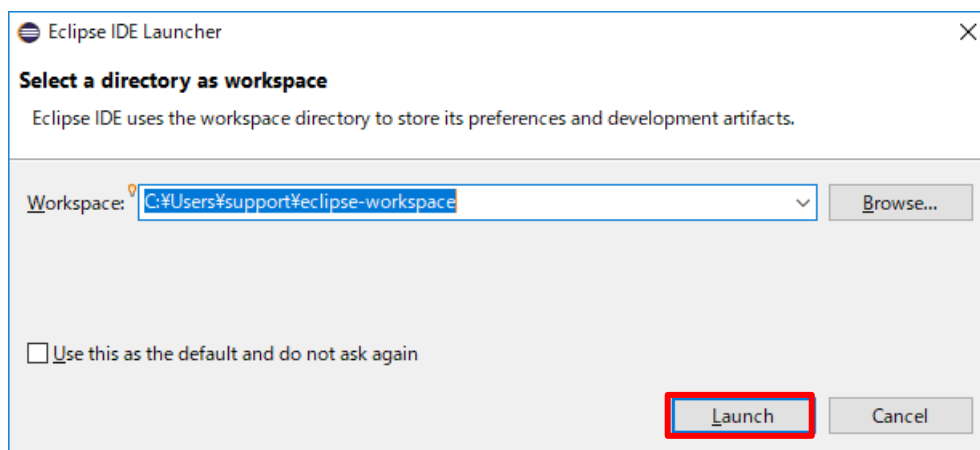
しばらく待つと以下のように表示されます。



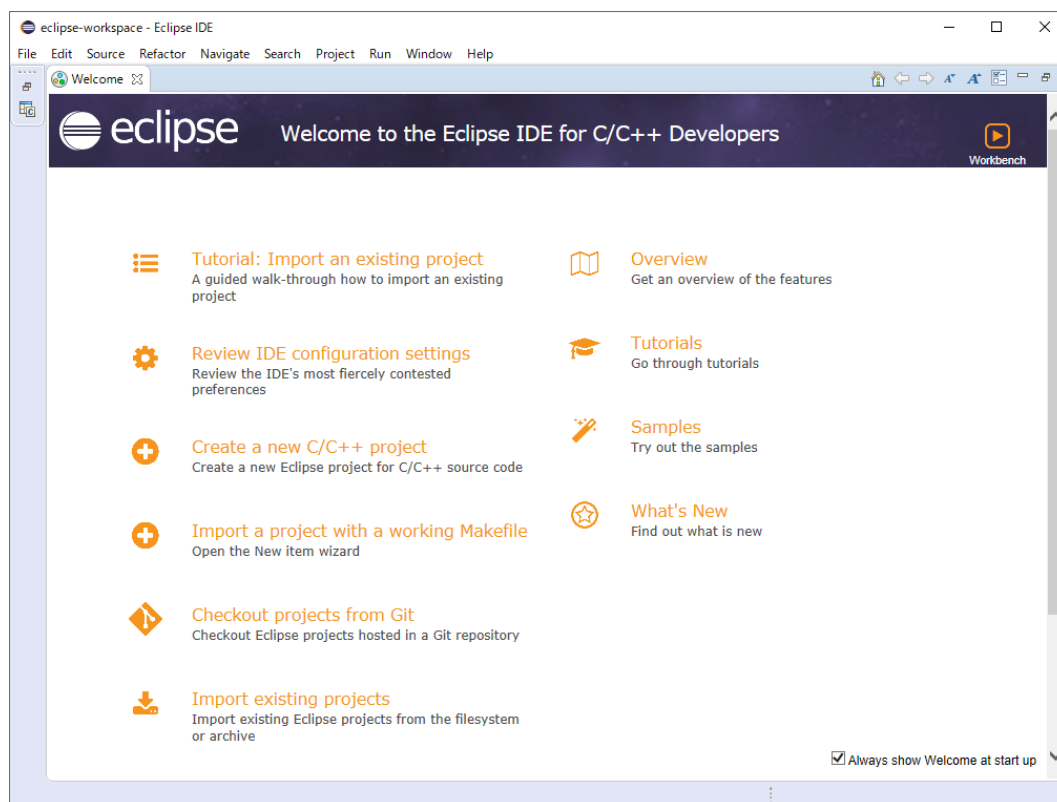
ここでは、「LAUNCH」をクリックしてください。

以下の表示になりますので、「Launch」をクリックしてください。

Workspace のフォルダは任意のフォルダで構いません。



しばらく待った後、以下のように Eclipse が起動すれば、インストールは終了です。



3. デバッグ手順について

リモートデバッグの手順について説明します。

3.1 アプリケーションのビルド

アプリケーションのビルドは従来通り、FXccIDE で行います。

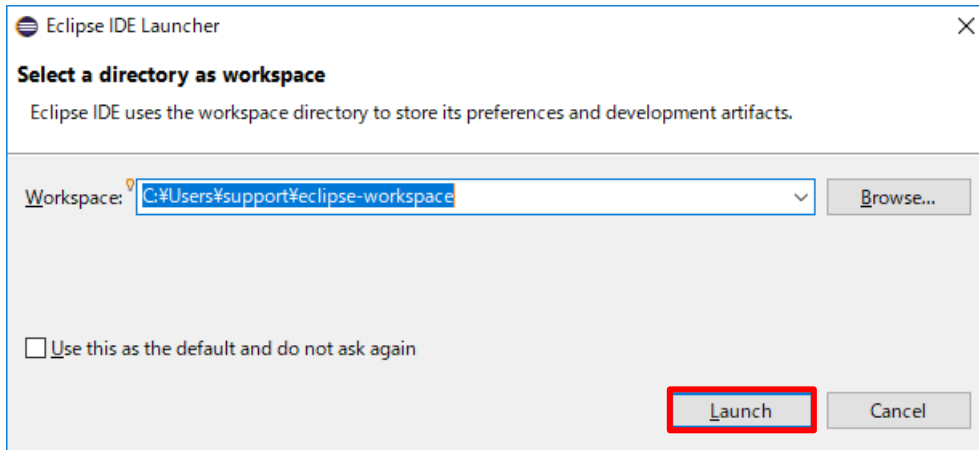
その際、「コンパイラ、リンカ設定」の Compiler Option にて「デバッグ有効」にチェックを入れてデバッグビルドを行ってください。



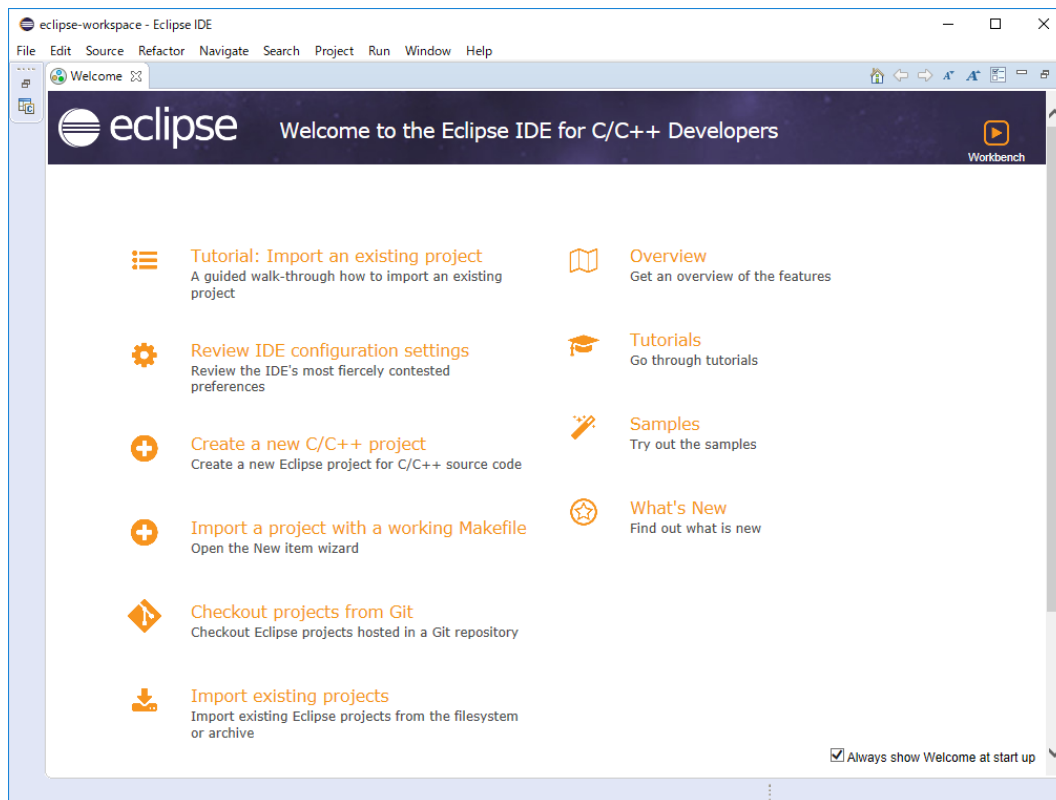
デバッグビルドを実行後、アプリケーションを FV2340-LNX に転送します。

3.2 Eclipse の起動

Windows のスタートメニューまたはデスクトップより Eclipse のアイコンをクリックします。
以下の表示になりますので、「Launch」をクリックしてください。
Workspace のフォルダは任意のフォルダで構いません。



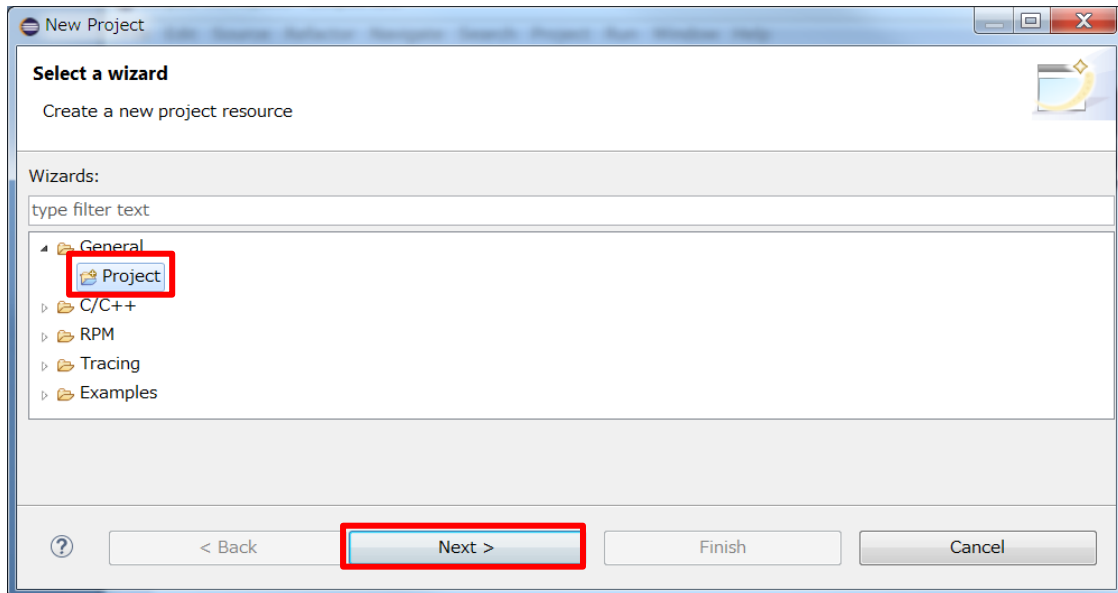
しばらく待った後、Eclipse が起動します。



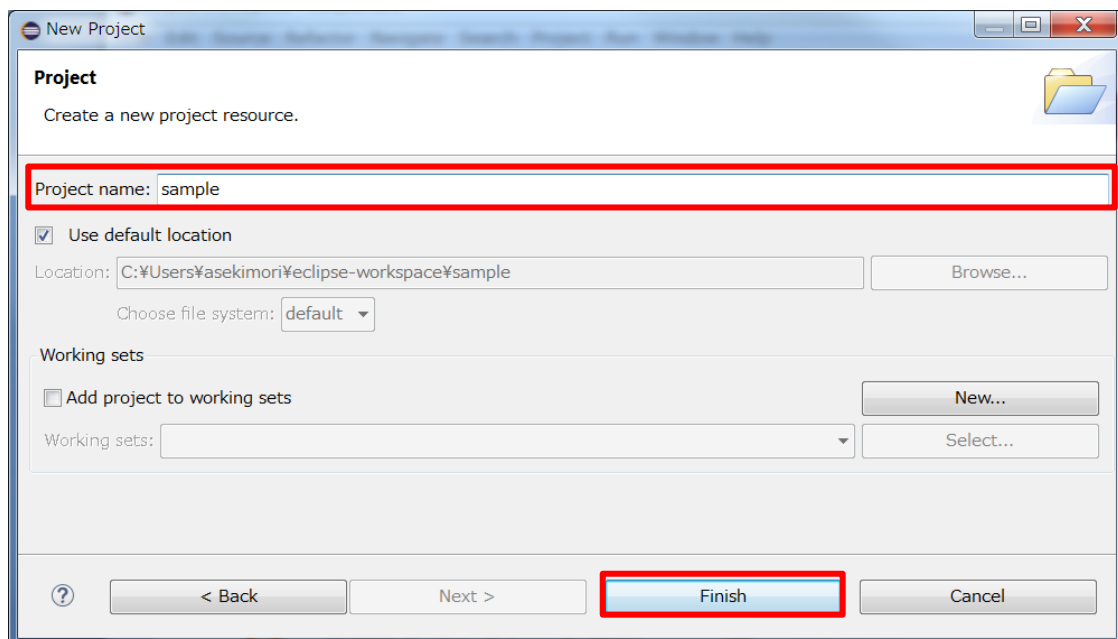
3.3 プロジェクトの作成

起動画面のFileメニューより「New」→「Project」を選択します。

以下の「Select a wizard」ダイアログが表示されるので、「General」「Project」を選択し、「Next」ボタンを押下します。



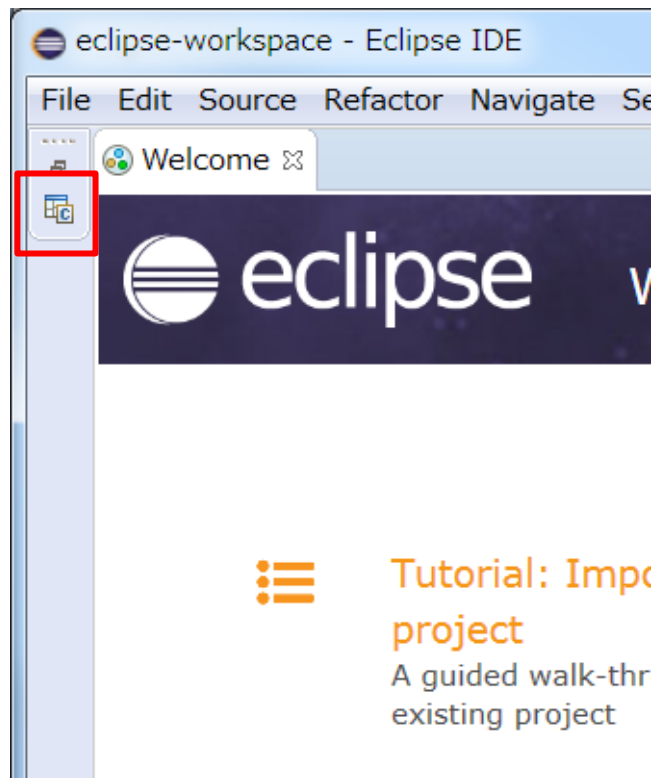
以下の「Project name」に任意のプロジェクト名(ここでは「sample」)を指定し、「Finish」ボタンを押下します。



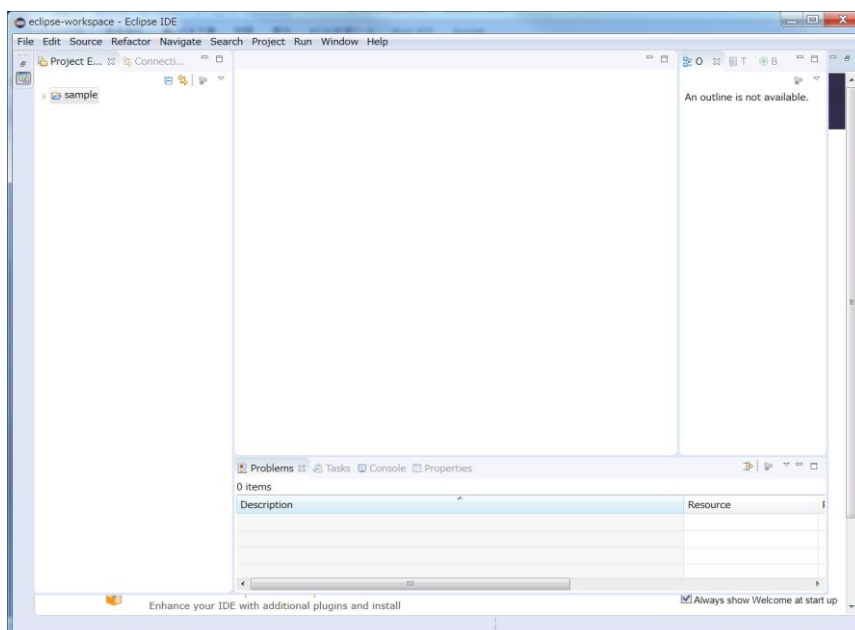
これでプロジェクトの作成は終了です。

3.4 デバッグの設定

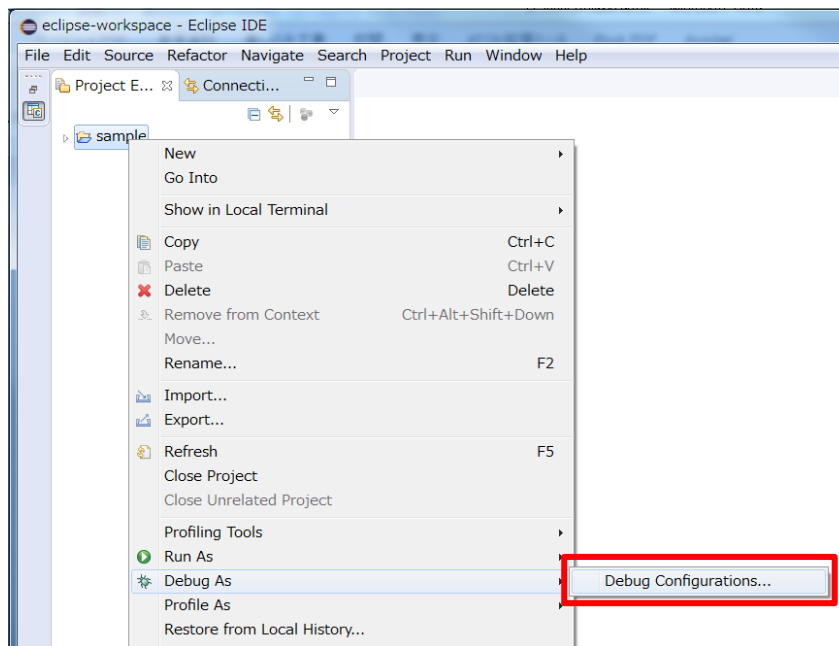
プロジェクトが作成できたので、デバッグの設定を行います。以下の画面の赤枠のアイコンをクリックします。



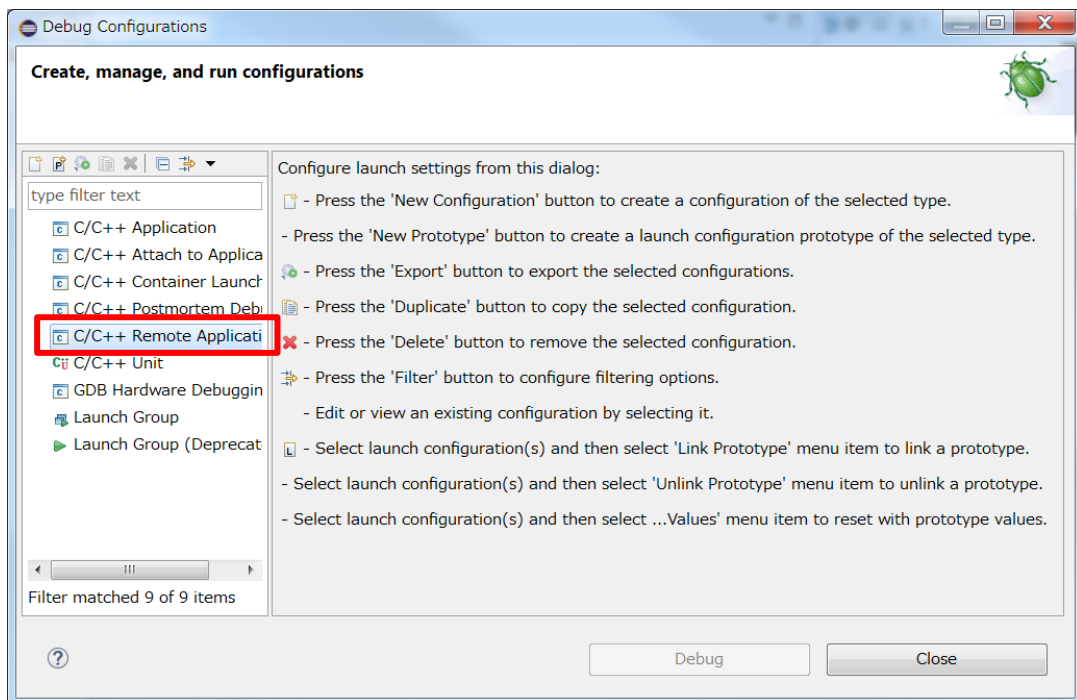
以下の画面のように「Project Explorer」が開きます。



「Project Explorer」から作成したプロジェクト名(ここでは「sample」)を指定し、マウス右クリックで「Debug As」→「Debug Configurations」を選択します。



以下のように、「Debug Configurations」ダイアログが表示されます。

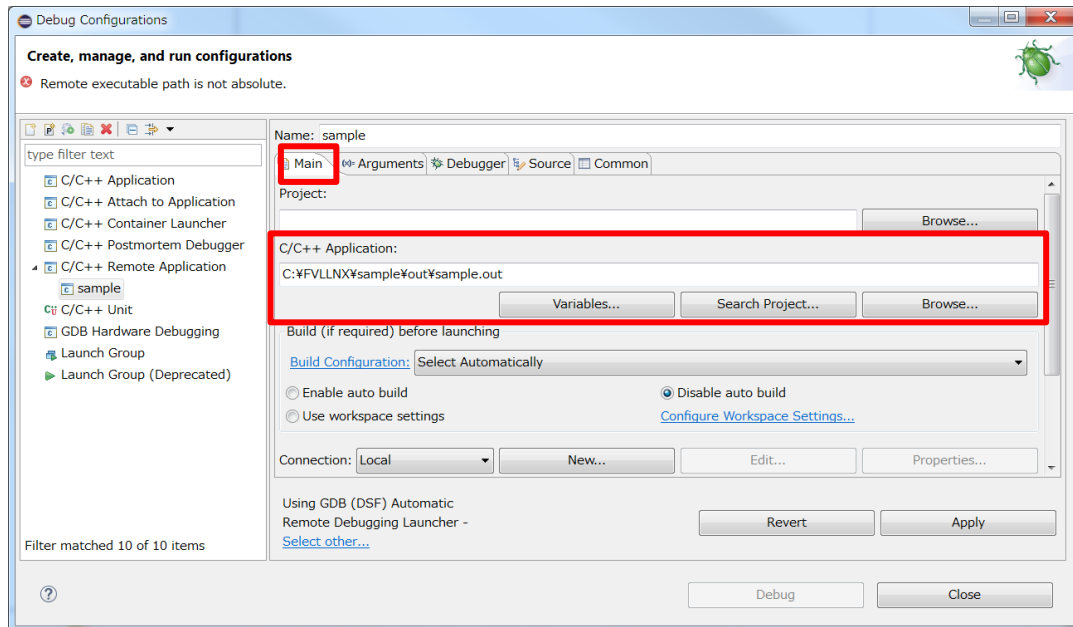


ダイアログ左ペインの「C/C++ Remote Application」をダブルクリックすると、プロジェクトのコンフィグ項目が新規に生成され、「Debug Configurations」ダイアログが表示されます。

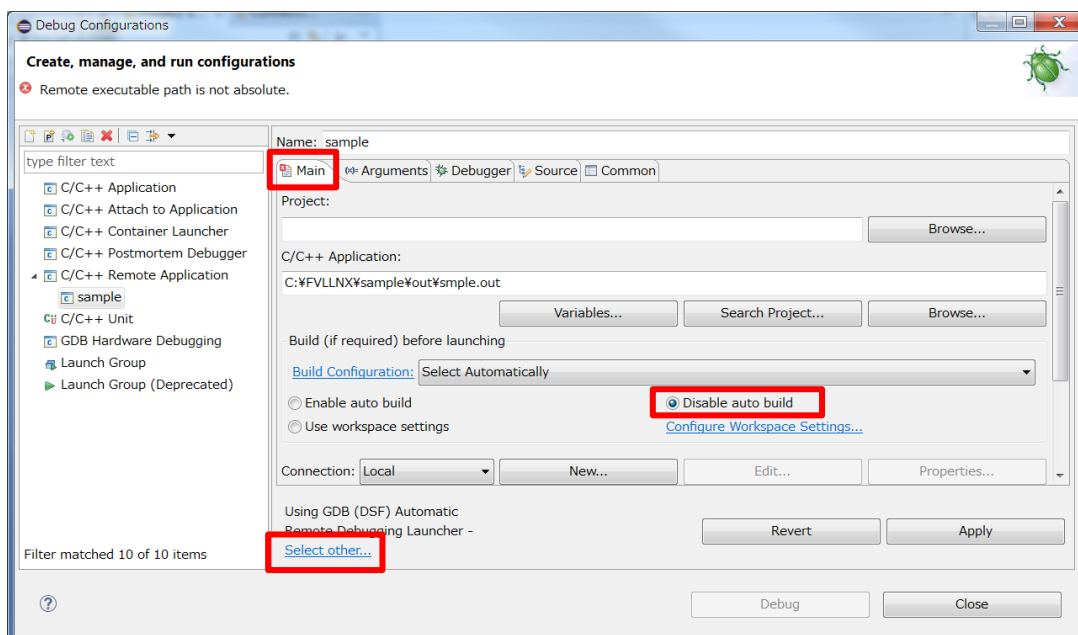
3.4.1 Debug Configurations - Main タブ

「Debug Configurations」ダイアログの右ペイン、「Main」タブ上の「C/C++ Application : 」の項目で、予めビルドしたデバッグ対象となる LNX 用実行モジュール(.out)を指定します。「Browse...」ボタンをクリックし、ファイルを選択します。

下記は、c:¥FVLLNX¥sample¥out¥sample.out を指定した例になります。

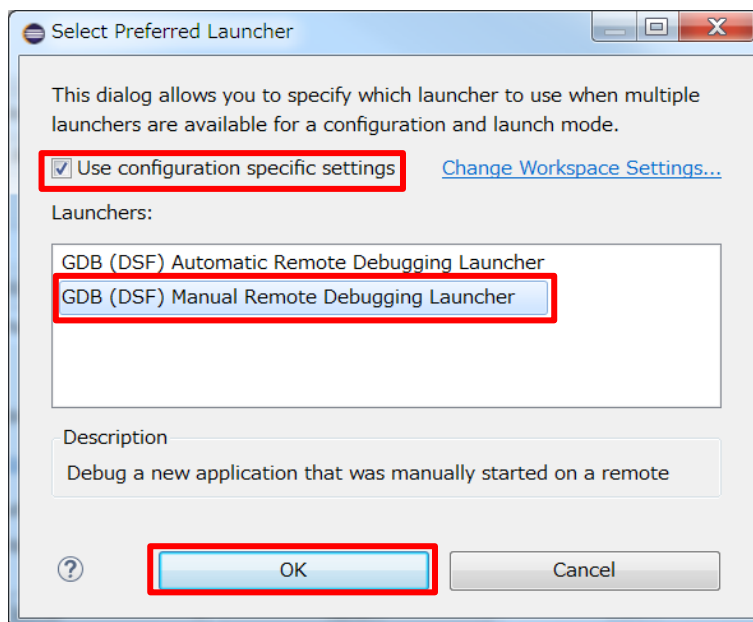


次に同じく「Main」タブ上「Build (if required) before launching」の項目で、「Disable auto build」をチェックします。



更に「Main」タブ 下端で「Select other...」をクリックします。

「Select other...」をクリックすると、「Select Preferred Lancher」ダイアログが表示されます。
ここで、「Use configuration specific settings」にチェックを入れます。
そして「launchers:」より「GDB (DSF) Manual Debugging Luancher」を選択して「OK」ボタンをクリックします。



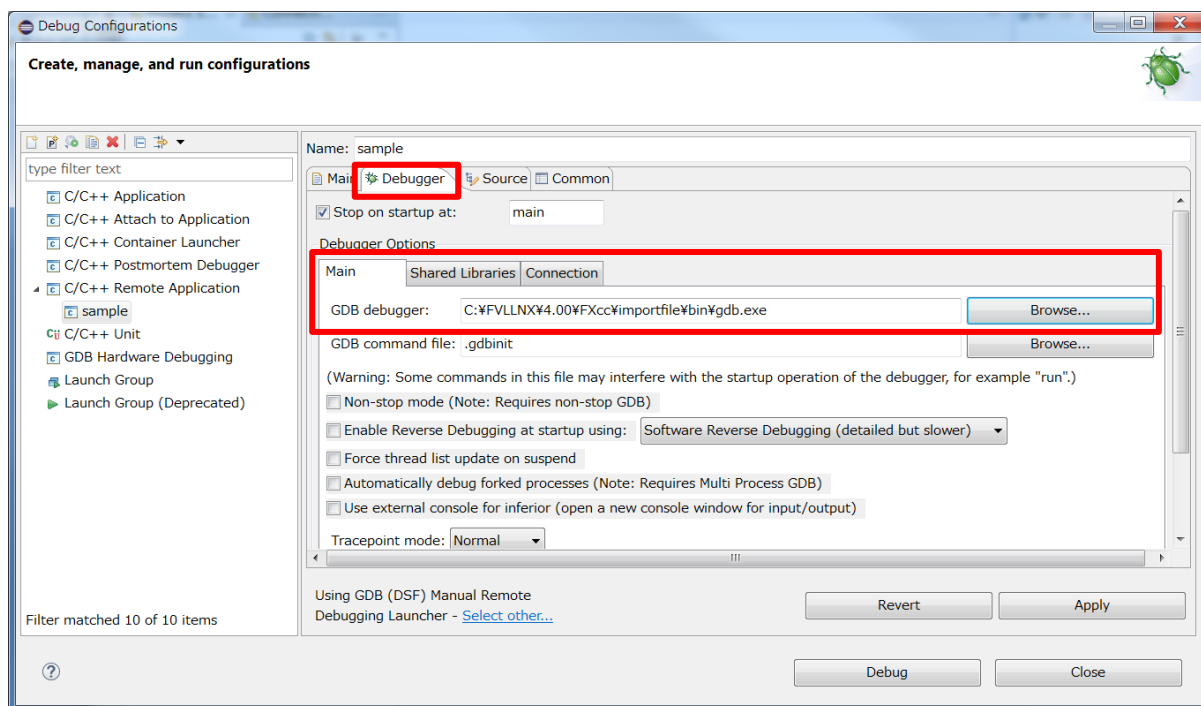
3.4.2 Debug Configurations - Debugger - Main タブ

「Debugger」タブ上の「Debugger Options」の項目から「Main」タブを選択し、「GDB debugger」を設定します。「Browse...」ボタンをクリックして、gdb.exe を指定します。

FVL/LNX SDK をデフォルトのフォルダにインストールした場合は

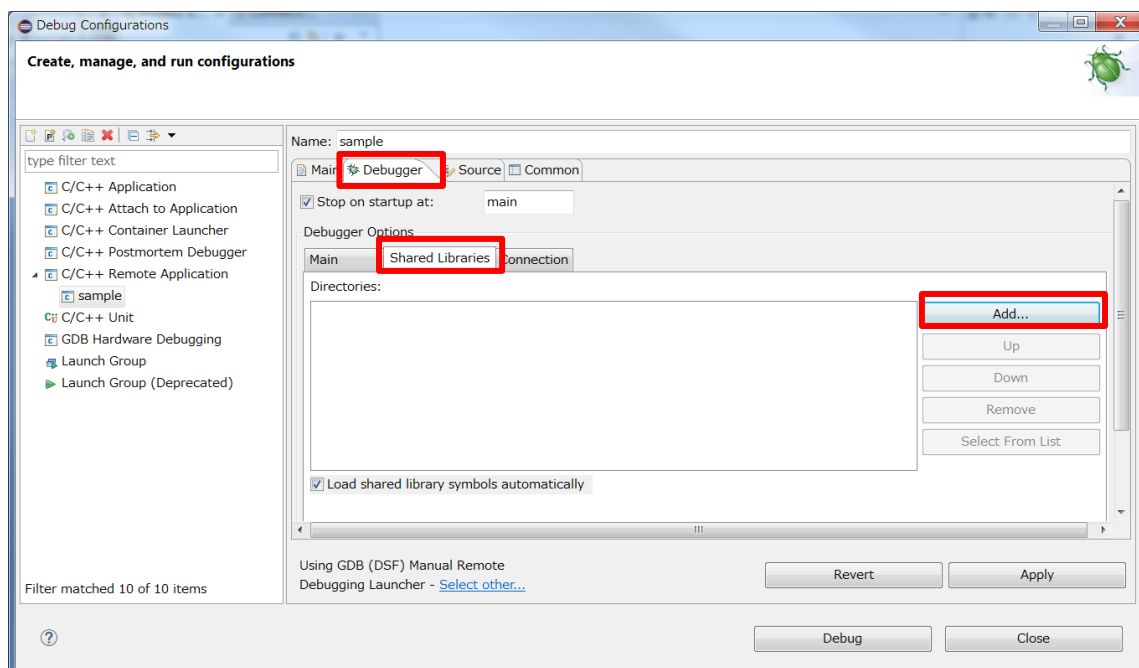
```
C:\FVLLNX\4.00\FXcc\importfile\bin\gdb.exe
```

を指定します。

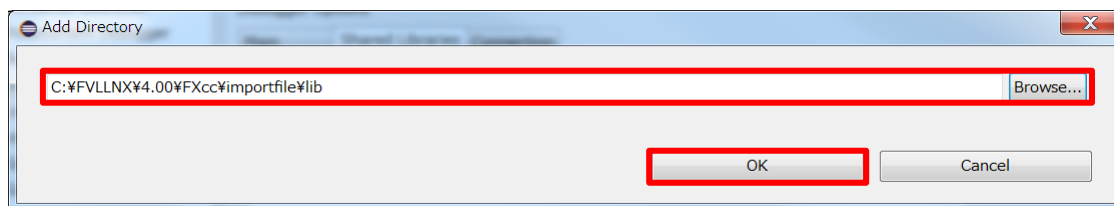


3.4.3 Debug Configurations - Debugger - Shared Libraries タブ

「Debugger」タブ上の「Debugger Options」の項目で「Shared Libraries」を追加します。
「Shared Libraries」タブを選択し、「Add...」ボタンをクリックします。



「Add Directory」ダイアログが表示されますので、共有ライブラリのパスを指定します。
FVL/LNX SDK をデフォルトのフォルダにインストールした場合、
C:¥FVLLNX¥4.00¥FXcc¥importfile¥lib
を指定し、「OK」ボタンをクリックします。



3.4.4 Debug Configurations - Debugger - Connection タブ

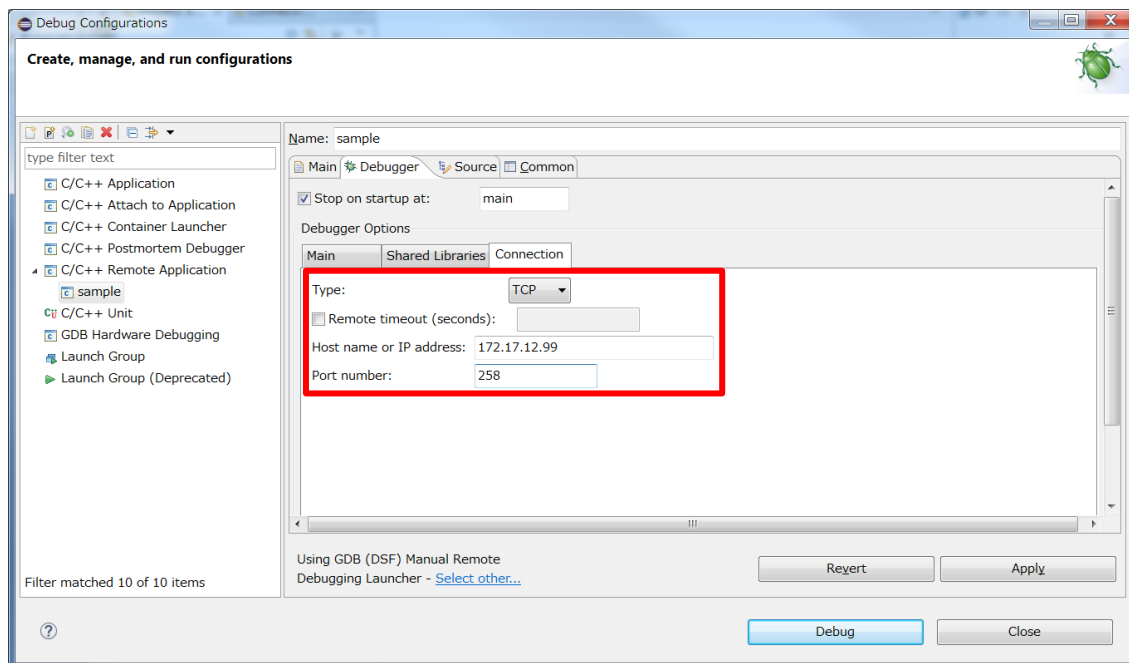
「Debugger」タブ上の「Debugger Options」の項目から「Connection」タブを選択し、FV2340-LNX との接続の設定を行います。

①LAN 接続の場合

Type : TCP を選択

Host name or IP address : FV2340-LNX 側の IP アドレス

Port number : FV2340-LNX 側のポート番号



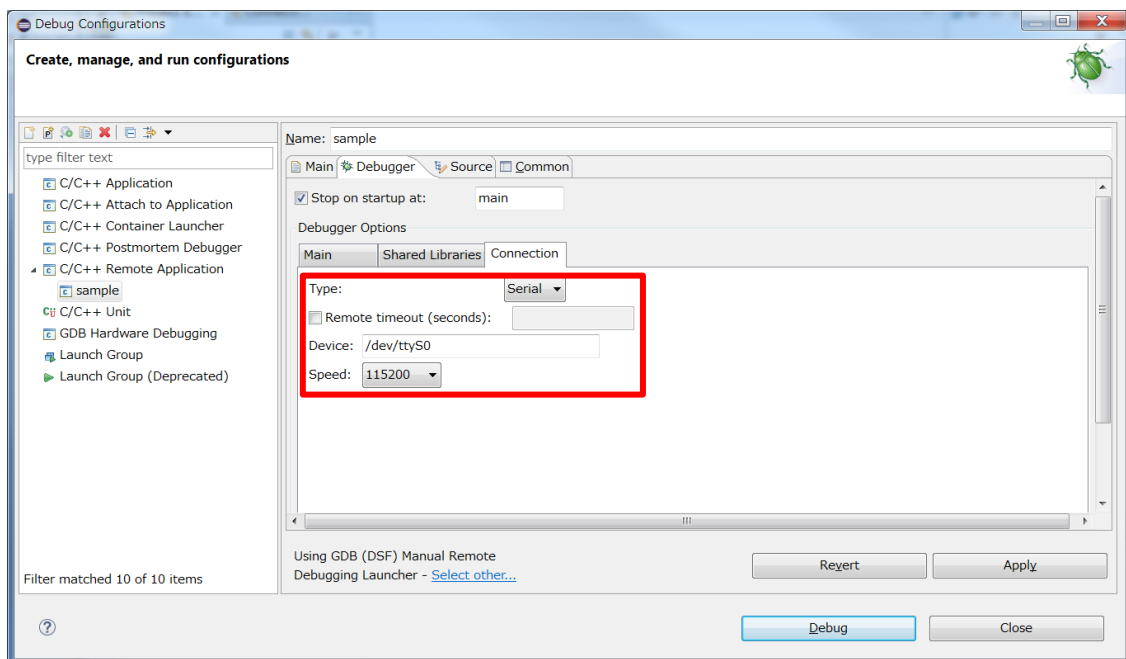
②シリアル接続の場合

Type : Serial を選択

Device : PC 側の COM 番号に適した Device を設定 (以下の表参照)

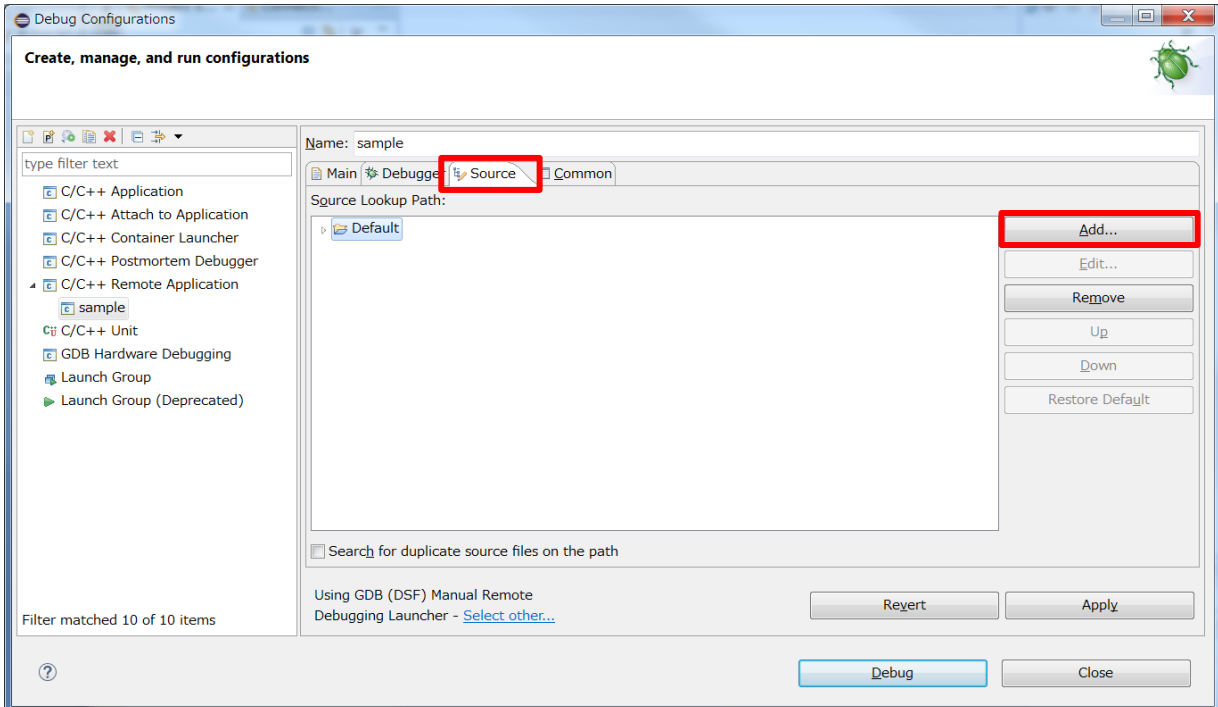
COM 番号	Device
COM1	/dev/ttyS0
COM2	/dev/ttyS1
COM3	/dev/ttyS2
COM4	/dev/ttyS3

Speed : ボーレートを選択

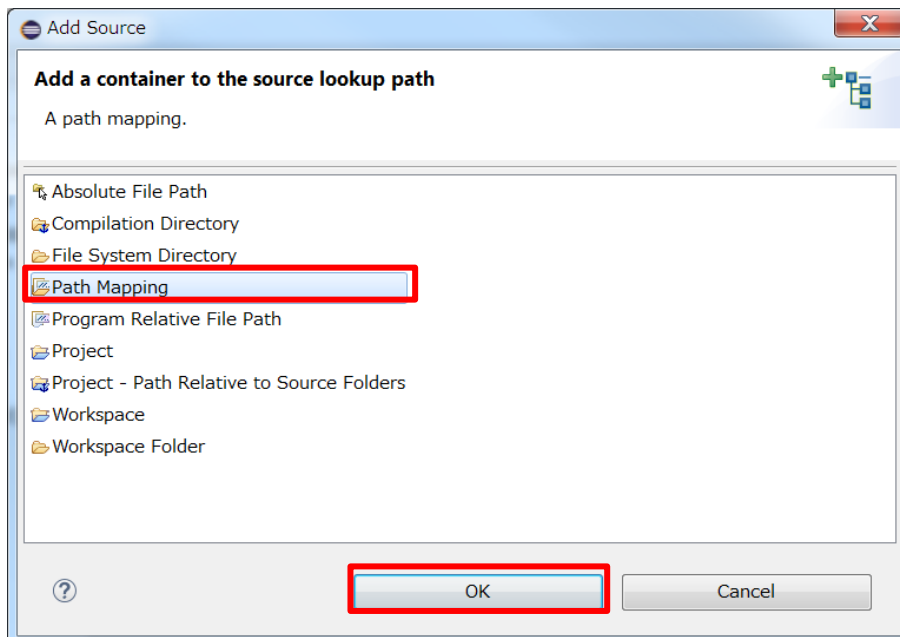


3.4.5 Debug Configurations - Source タブ

「Debug Configurations」ダイアログの右ペイン、「Source」タブ上の「Source Lookup Path」の項目を設定します。「Add...」ボタンをクリックします。



「Add Source」ダイアログが開きます。

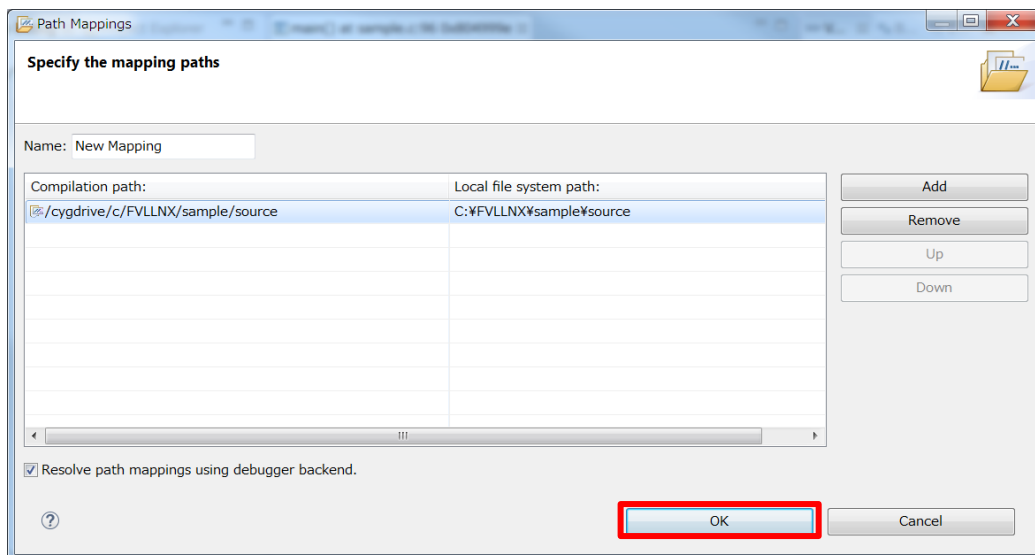


「Path Mapping」を選択して「OK」ボタンをクリックします。
「Path Mapping」ダイアログが開きますので、「Add...」ボタンをクリックしマッピングを設定します。

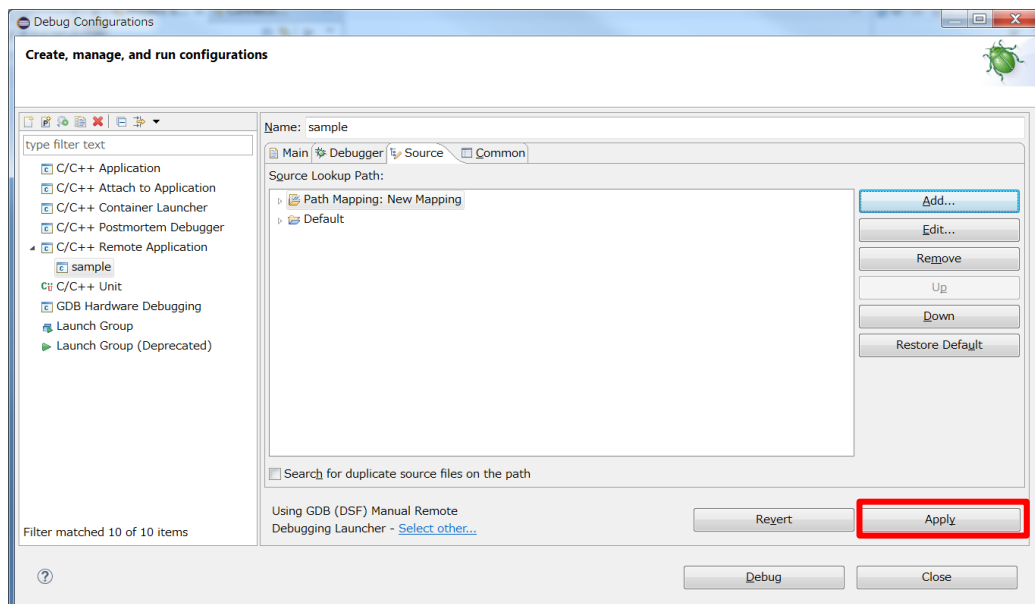
以下のようにソースファイルの場所を定義します。

例) c:¥xxx¥source にソースファイルがある場合

- Compilation path: /cygdrive/c/xxx/source
/cygdrive/に続けてアプリケーションのソースファイルがあるフォルダを記述する。
- Local file system path: c:¥xxx¥source
アプリケーションのソースファイルがあるフォルダを記述する。



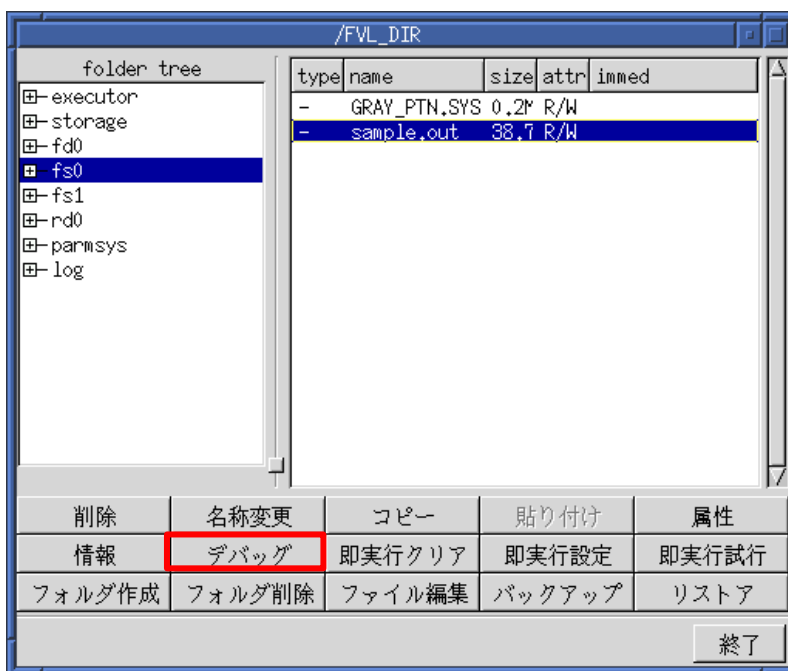
記述したら「OK」をクリックします。続いて「Apply」をクリックします。



以上で、デバッグの設定は完了です。

3.5 デバッグの実行

FV2340-LNX 装置を起動して、「ファイル」メニューからデバッグビルドした実行モジュール(.out)を指定し、「デバッグ」をクリックします。



デバッグダイアログが表示されます。



シリアル接続の場合には、デバイスに RS232C を選択します。

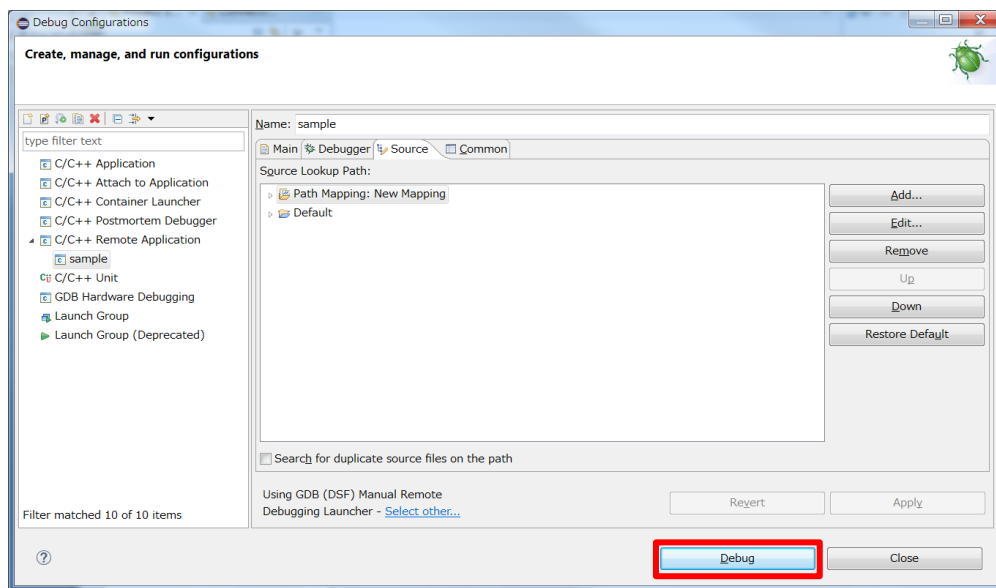
LAN 接続の場合には、デバイスに E_LINK を選択し、ポート番号を設定します。

なお、RS232C や E-LINK についての各種設定は予めシステムメニューより行う必要があります。

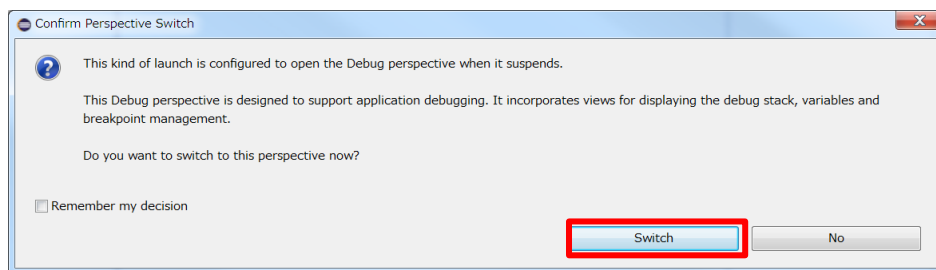
詳しくは「LNX 操作説明書」の「システムパラメーター 入出力設定」の項目をご参照ください。

接続するデバイスを選択し実行をクリックします。アプリケーションが起動し、デバッガからの接続待ち状態になります。

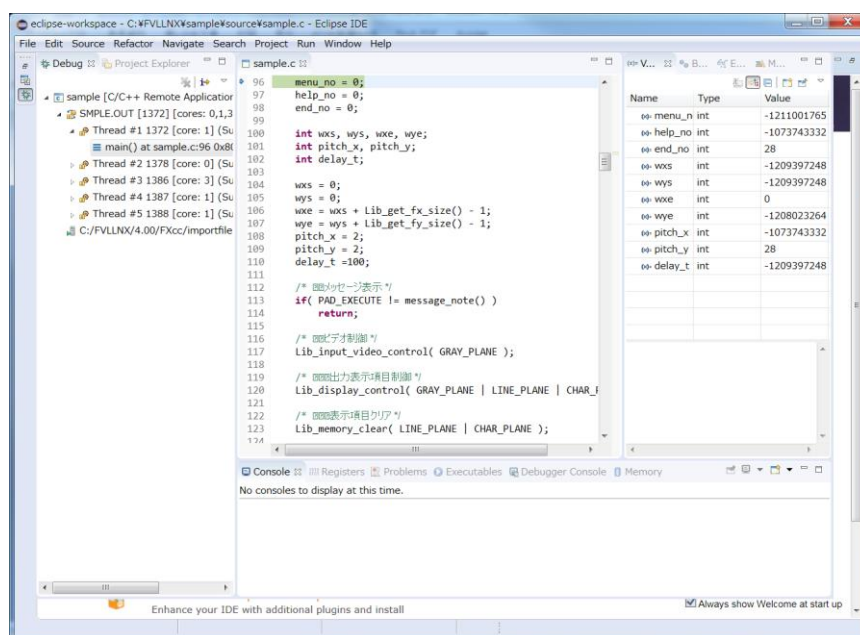
「Debug Configurations」ダイアログで、「Debug」をクリックします。



以下の「Confirm Perspective Switch」ダイアログが表示されますので「Switch」をクリックします。




しばらく待つと FV2340-LNX と接続され、以下のようなデバッグ画面が表示されます。

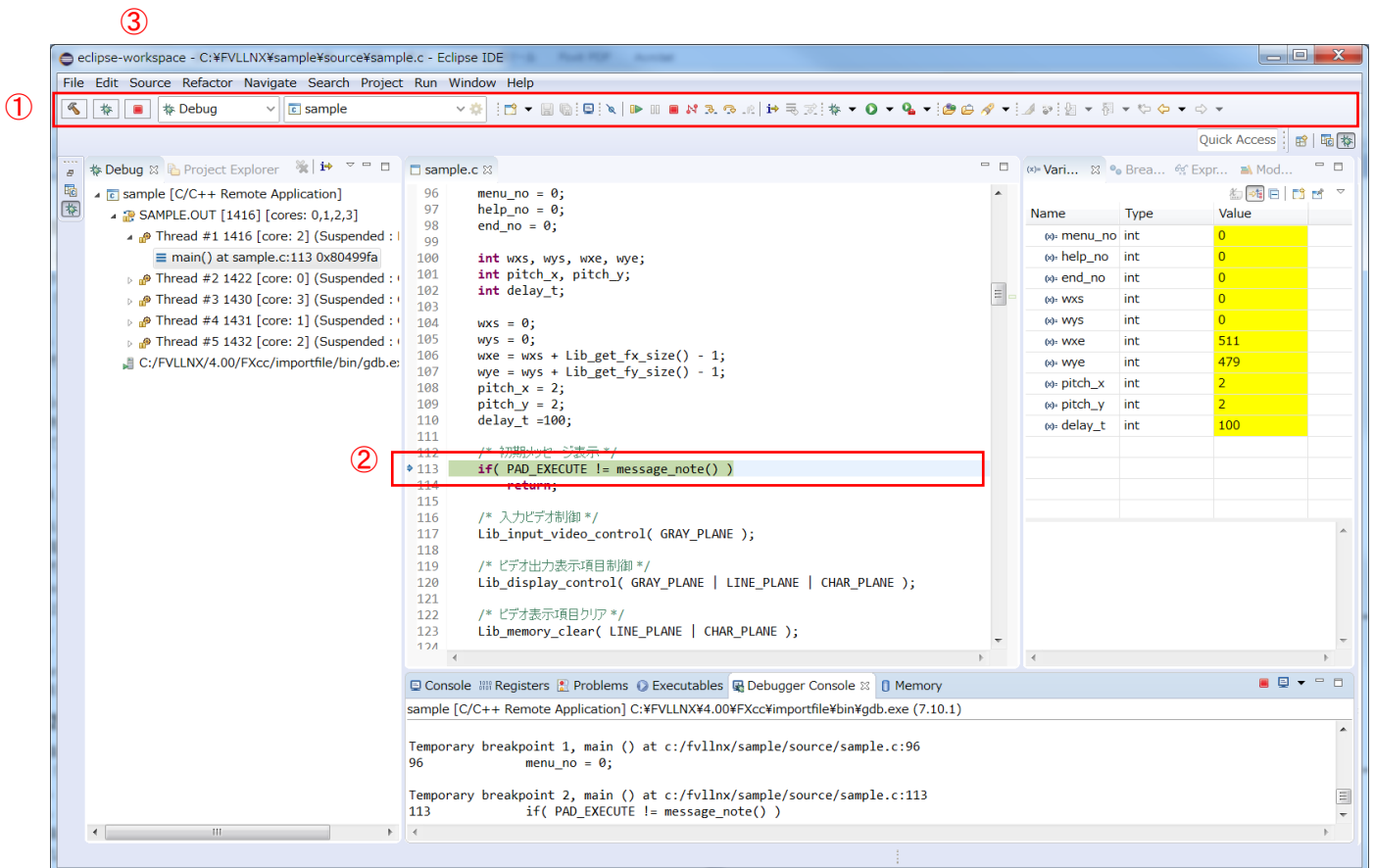


4. 操作編

主な操作を簡単にご紹介します。詳しくは Help 等をご参照ください。

操作	内容
Run → Step Over (F6)	ステップ実行。関数の中には入らない。
Run → Step Into (F5)	ステップ実行。関数の中に入る。
Run → Run to Line (Ctrl + R)	選択している行まで実行する。
Run → Resumu (F8)	実行再開。ブレイクポイントまで実行する。
Run → Terminate (Ctrl + F2)	デバッガを終了する。

- ①「Window」→「Appearance」→「Show Toolbar」で以下のようにツールバーを表示することが可能です。
- ②ソースコードの行番号の前部分をダブルクリックすると、ブレイクポイントを設定・解除できます。
- ③  「Terminate」ボタンでデバッグを終了することができます。
- ④「Window」→「Perspective」→「Close Perspective」を選択すると、デバッグウィンドウが閉じます。



B-003487

FVL/LNX Ver4. xx
ユーザアプリケーション リモートデバッグに関する技術資料

2022年1月 第2版発行

発行所 株式会社ファースト

本 社 〒242-0001 神奈川県大和市下鶴間 2791-5

ユーザ・サポート E-mail: support@fast-corp.co.jp
