

# DAVE™ version 4 – EtherCATプロジェクト作成手順

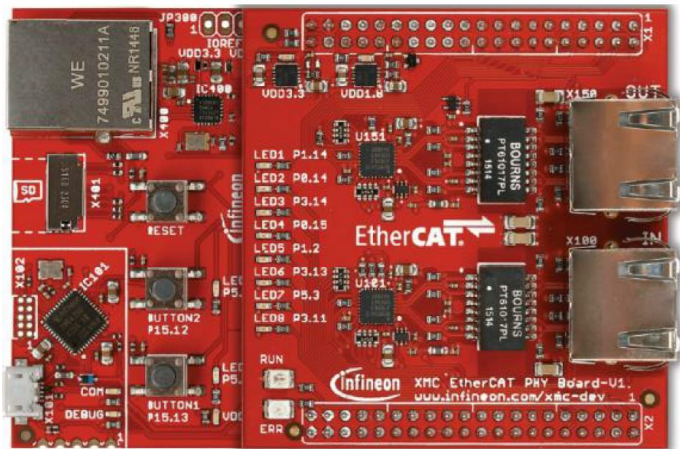


# このドキュメントについて

- › このドキュメントは、“XMC4800 EtherCAT APP SSC Slave Example Getting Started V3.0”に沿って、EtherCAT SlaveのサンプルプロジェクトをKIT\_XMC48\_RELAX\_ECAT\_V1に実装し、PCにInstallしたTwinCAT3を使用して制御を行う手順を説明するものです。
- › DAVE™の概要及びインストール手順についてはドキュメント“DAVE\_Introduction\_J”を参照してください。このドキュメントではPCにVersion4.2.2がインストールされている事を前提としています。

# 使用するハードウェア(Kit)について

- このドキュメントで作成するプロジェクトを動作させるターゲットハードウェアとして KIT\_XMC48\_RELAX\_ECAT\_V1を前提としています。



KIT\_XMC48\_RELAX\_ECAT\_V1

[https://www.infineon.com/cms/jp/product/evaluation-boards/kit\\_xmc48\\_relax\\_ecat\\_v1/](https://www.infineon.com/cms/jp/product/evaluation-boards/kit_xmc48_relax_ecat_v1/)

# 環境設定手順

# 環境設定について

- › PC側の環境設定として以下2つのソフトウェアをインストールする必要があります。
- **TwinCAT3**
  - PCをEtherCAT Master として動作させるためのソフトウェアです。
  - TwinCAT3をダウンロードするにはBeckhoff社のサイトのアカウントが必要です。
- **EtherCAT Slave Stack Code Tool(以降SSC Tool)**
  - KIT\_XMC48\_RELAX\_ECAT\_V1 にEtherCAT Slaveのサンプルプロジェクトを実装するには、Infineonが提供するプロジェクトファイル(ETHCAT\_SSC\_XMC48.zip)にEtherCATのSSC(Slave Stack Code)を組み込む必要があります。
  - このSSCをEtherCAT Slaveの入出力等の設定に合わせて生成するのが、Beckhoff社のSSC Toolとなります。
  - SSC Toolをダウンロードするには、ETG(EtherCAT Technology Group)のサイトのLoginIDとPasswordが必要となります。

# TwinCAT3インストール

1. ブラウザでBeckhoff社のサイトにアクセスし、**TwinCAT3 eXtended Automation Engineering (XAE)**をダウンロードします。  
<https://www.beckhoff.com/ja-jp/support/download-finder/software-and-tools/>
2. ダウンロードしたzipファイル(TC31-Full-Setup.3.1.4024.20.zip)を展開し、TC31-FULL-Setup.3.1.4024.20.exeをダブルクリックします。
3. TwinCAT3及び3rd Party SoftwareのLicense agreementはAcceptし、Setup typeはCompleteを選択してInstallを行います。
4. インストールの完了時に再起動を求められるのでYesをクリックします。

# SSC Toolインストール

1. ブラウザでETG(EtherCAT Technology Group)のサイトにアクセスします。  
[http://www.ethercat.org/memberarea/stack\\_code.aspx](http://www.ethercat.org/memberarea/stack_code.aspx)
2. LoginIDとPasswordの入力を行いLoginをクリックします。
3. EtherCAT Vendor IDに0000034E(InfineonのVendorID)を入力してSubmit Registrationをクリックします。
4. Personal informationに名前やメールアドレスを入力し、DownloadのVersionでSSC V5.13(or later)を選択しSubmitをクリックします。
5. Beckhoff社よりダウンロードリンクが記載されたメールが届くのでリンクをクリックしてダウンロードします。
6. ダウンロードしたzipファイル(SSC\_V5i13.zip)を展開し、EtherCAT Slave Stack Code Tool.exeをクリックしてインストールを行います。

# プロジェクト作成手順



# プロジェクトのダウンロード

1. [https://www.infineon.com/cms/en/product/promopages/aim-mc/dave\\_downloads.html](https://www.infineon.com/cms/en/product/promopages/aim-mc/dave_downloads.html)にブラウザでアクセスします。
2. Keyword Search FilterにEtherCATと入力し、I Agree with the Statement Aboveをチェックして、ETHCAT\_SSC\_XMC48をクリックしてダウンロードします。

## Download of Example Projects for DAVE™ Apps

List of examples projects that can be downloaded after accepting the terms of use. The downloaded project zip file can be imported into DAVE™ version 4 using the DAVE™ import functionality: ->File ->Import -> DAVE Project; then browse to the project zip file.

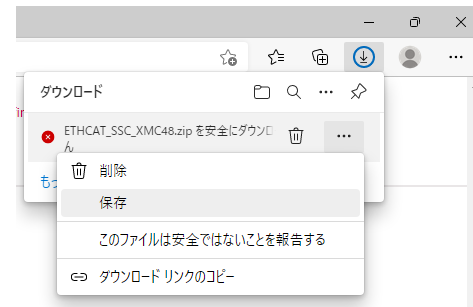
The software that can be downloaded from the list below is protected by national and international copyright laws and may be protected by other intellectual property rights. You shall use the software only in accordance with the applicable licensing terms and conditions which may be different from the terms and conditions of the DAVE v4 Software License Agreement. For copyright information, licensing terms and additional information (e.g. on how to obtain the source code of such Open Source Software), please check the "Help Function" Section "Copyright and Licensing Information" of the software. By downloading the software, you acknowledge that you have read and understood this download notice.

☒ I Agree with the Statement Above

Keyword Search Filter  [Reset Search](#)

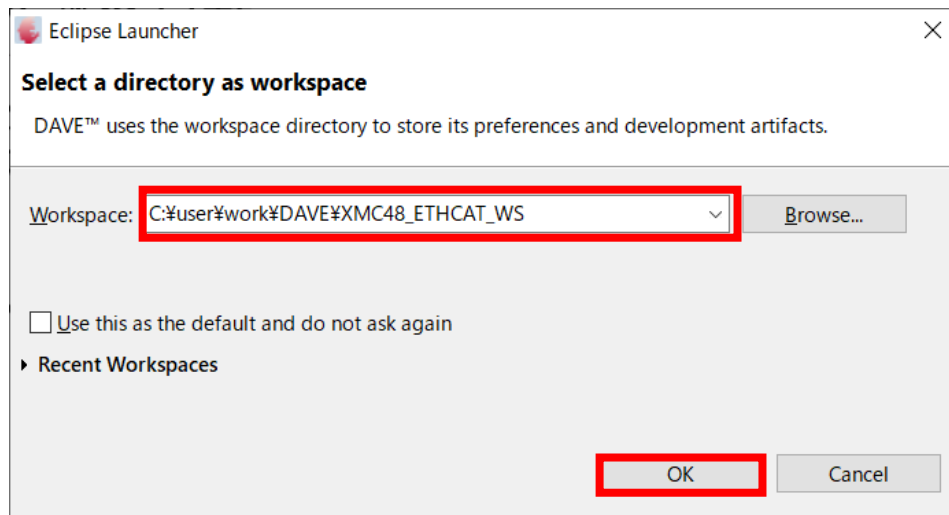
Project	Description	Target MCU series	Version	Last Update	Tool
ETHCAT_FWUPDATE_SSC_APPLICATION_XMC48	This example demonstrates the implementation of a EtherCAT slave node which is capable doing firmware updates via FoE (File over EtherCAT). The example runs on the XMC4800 Relax EtherCAT Kit.	XMC4800	3.0.0	2018-03-15	DAVE
<b>ETHCAT_SSC_XMC48</b>	<b>This example demonstrates the implementation of a EtherCAT slave node using the Beckhoff SSC Tool to generate the slave stack code for the XMC4800 Relax EtherCAT Kit.</b>	<b>XMC4800</b>	3.0.0	2018-03-15	DAVE
ETHCAT_SSC_XMC48	This example demonstrates the implementation of a EtherCAT slave node using the Beckhoff SSC Tool to generate the slave stack code for the XMC4800 Relax EtherCAT Kit.	XMC4300	3.0.0	2018-03-15	DAVE

**Note:** chrome ブラウザでは正しくダウンロードできない場合があります。その場合は、Edge等他のブラウザを使用してください。また、安全にダウンロードできません等の表示があった場合は、手動で保存してください。



# DAVE™の起動


1. スタートメニュー, タスクバー, ショートカット等を使用してDAVE™を起動します。
2. プロジェクトを格納するワークスペース・フォルダを指定します。  
**Note:** 旧版のDAVE™で作成したワークスペース・フォルダは指定しないでください。  
**Note:** フォルダ名、Pathには、マルチバイト文字を使用しないでください。
3. OKをクリックします。

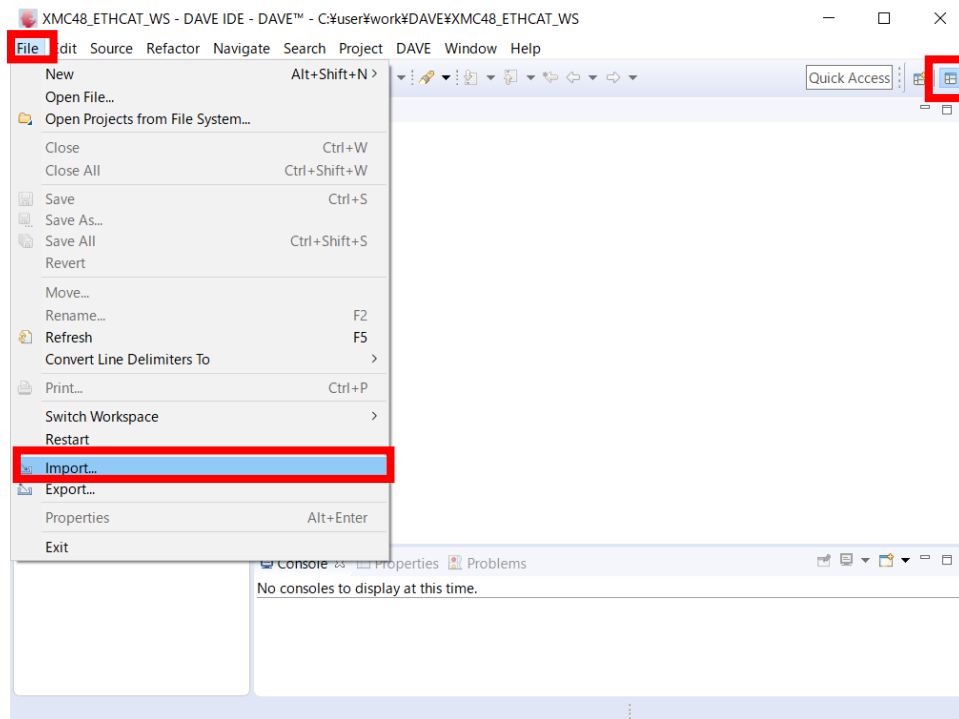


# プロジェクトの新規作成(1/3)

› DAVE™ APPを使用するため、DAVE™ CE(Code Engine) プロジェクトを作成します。

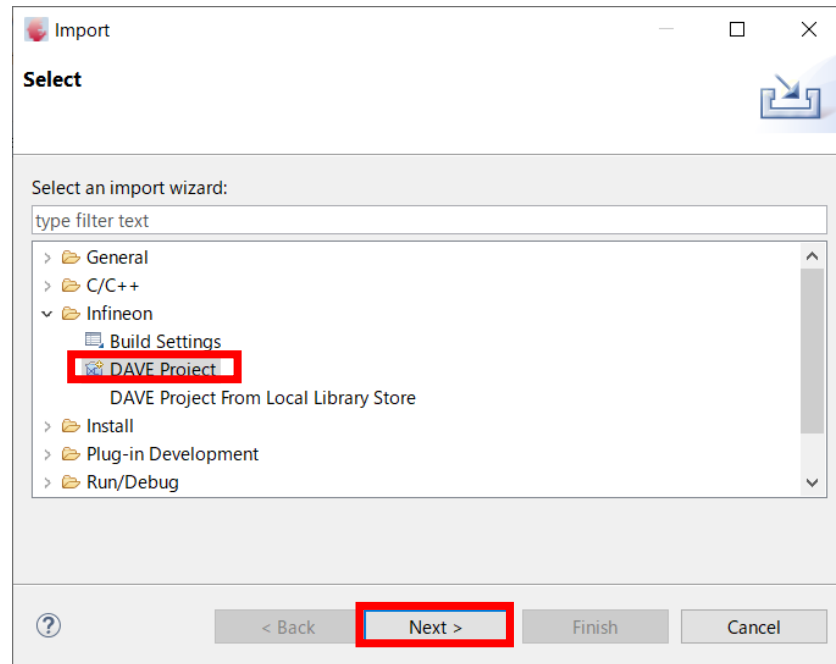
1. File → Importを選択します。

**Note:** DAVE IDE以外のパースペクティブが選択されている場合は、右上のDAVE IDEパースペクティブをクリックしてください 



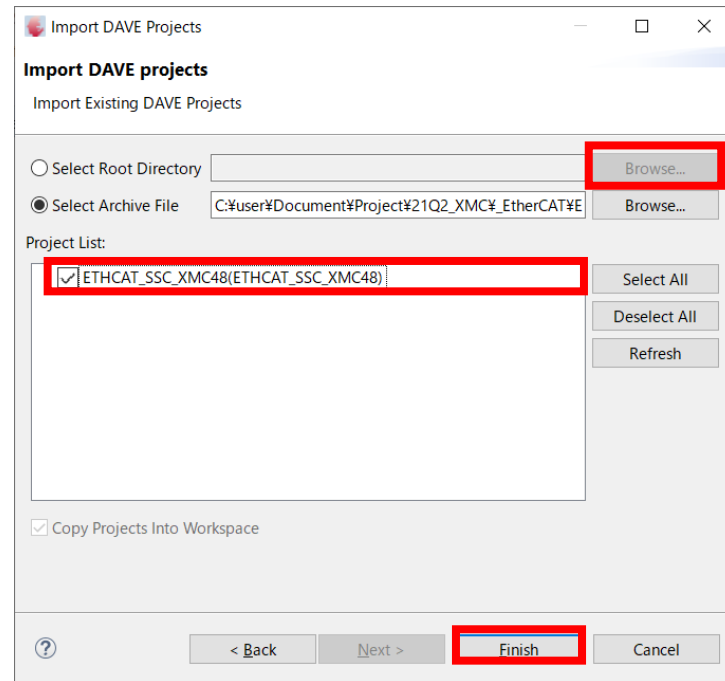
## プロジェクトの新規作成(2/3)

2. Infineon→DAVE™ Projectを選択します
3. Next をクリックします



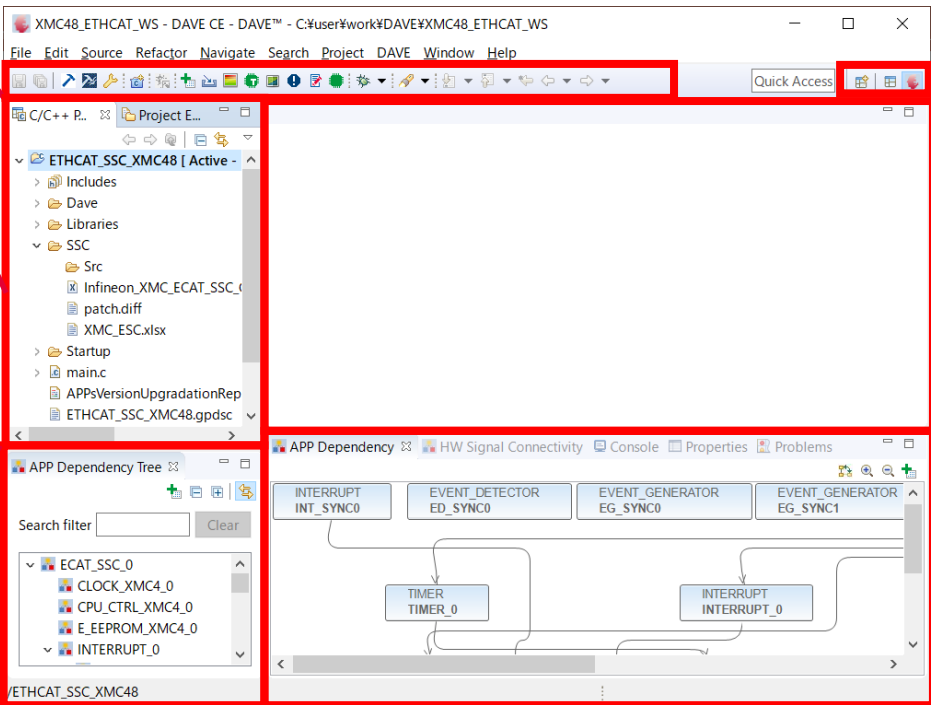
# プロジェクトの新規作成(3/3)

4. Browseをクリックしてダウンロードしたプロジェクトファイルを選択し、Project Listで ETHCAT\_SSC\_XMC48にチェックを入れます。
5. Finishをクリックします。



# DAVE™ CEパースペクティブ

- プロジェクトを生成すると自動的にDAVE™ CEパースペクティブ(画面レイアウト)に切り替わります。

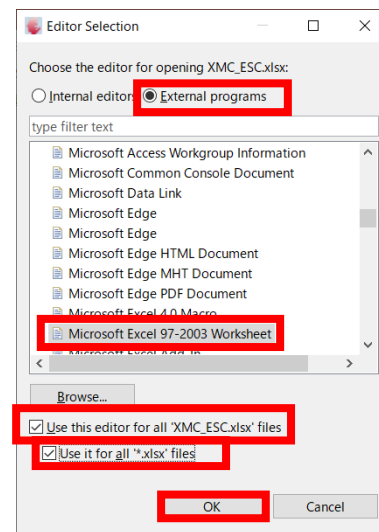
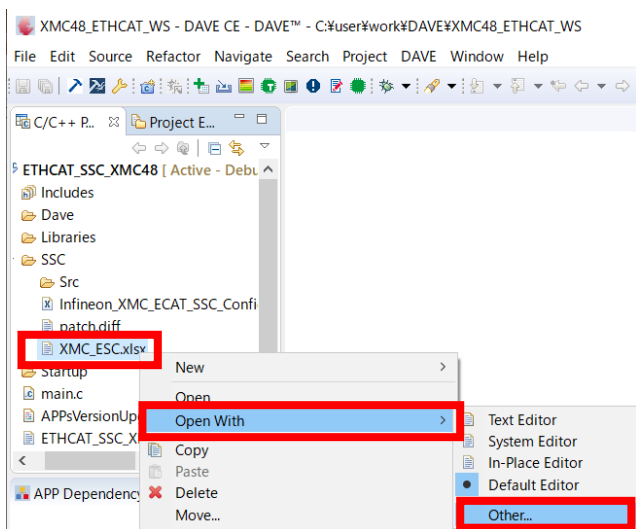


The screenshot shows the DAVE™ CE interface with the following panels and callouts:

- Tool Panel**: Located at the top left, containing various tool icons.
- Project Explorer**: Located on the left side, showing the project structure (Project Explorer, Includes, Dave, Libraries, SSC, Src, Startup, main.c, APPsVersionUpgradationRep, ETHCAT\_SSC\_XMC48.gpdsc).
- APP Dependency Tree**: Located at the bottom left, showing the APP dependency tree (ECAT\_SSC\_0, CLOCK\_XMC4\_0, CPU\_CTRL\_XMC4\_0, E\_EEPROM\_XMC4\_0, INTERRUPT\_0).
- Perspective (画面レイアウト)の切替**: Located at the top right, showing the perspective view icon.
- APP Config / Code editor**: Located in the center, showing the APP configuration and code editor (DAVE™ APPの設定 / コードエディタ View).
- APP Dependency / APP Connectivity**: Located at the bottom right, showing the APP dependency and connectivity view (APP依存性 / 接続性 View).

# SSCのコンフィグレーション確認(1/2)

1. Project explorer 内のSSC/XMC\_ESC.xlsファイルを右クリックしてOpen With→Otherを選択します。
2. External programsを選択し、Microsoft Excel 97-2003 Worksheetを選択して、Use this editor for all 'XMC\_ESC.xlsx' filesとUse it for all '\*.xlsx' fileにチェックを入れOKをクリックします。



## SSCのコンフィグレーション確認(2/2)

1. Excelで開かれたファイルXMC\_ESC.xlsの内容が以下であることを確認します。

- 0x6000からの入力データに16bit整数 x 4, 1bit Boolean x 8が定義されている事
- 0x7000からの出力データに16bit整数 x 4, 1bit Boolean x 8が定義されている事

**Note:** インターフェイスを定義する方法の詳細については、  
SSCツールのEtherCAT\_Slave\_Design\_Quick\_Guide.pdfを参照してください

Index	ObjectCode	SI	DataType	Name	Default
//0x6nnx	Input Data of the Module (0x6000 - 0x6FFF)				
0x6000	RECORD			IN_GENERIC	
		0x01	UINT	IN_GEN_INT1	0
		0x02	UINT	IN_GEN_INT2	0
		0x03	UINT	IN_GEN_INT3	0
		0x04	UINT	IN_GEN_INT4	0
		0x05	BOOL	IN_GEN_Bit1	0
		0x06	BOOL	IN_GEN_Bit2	0
		0x07	BOOL	IN_GEN_Bit3	0
		0x08	BOOL	IN_GEN_Bit4	0
		0x09	BOOL	IN_GEN_Bit5	0
		0x0A	BOOL	IN_GEN_Bit6	0
		0x0B	BOOL	IN_GEN_Bit7	0
		0x0C	BOOL	IN_GEN_Bit8	0

Index	ObjectCode	SI	DataType	Name	Default
//0x7nnx	Output Data of the Module (0x7000 - 0x7FFF)				
0x7000	RECORD			OUT_GENERIC	
		0x01	UINT	OUT_GEN_INT0	
		0x02	UINT	OUT_GEN_INT0	
		0x03	UINT	OUT_GEN_INT0	
		0x04	UINT	OUT_GEN_INT0	
		0x05	BOOL	OUT_GEN_Bit0	
		0x06	BOOL	OUT_GEN_Bit0	
		0x07	BOOL	OUT_GEN_Bit0	
		0x08	BOOL	OUT_GEN_Bit0	
		0x09	BOOL	OUT_GEN_Bit0	
		0x0A	BOOL	OUT_GEN_Bit0	
		0x0B	BOOL	OUT_GEN_Bit0	
		0x0C	BOOL	OUT_GEN_Bit0	



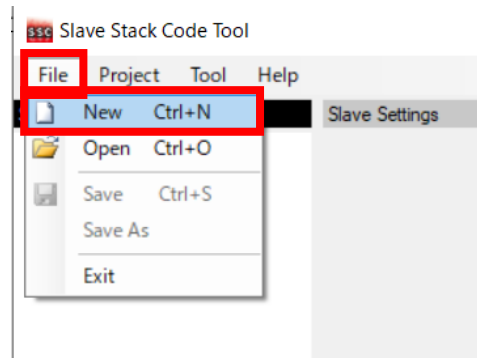
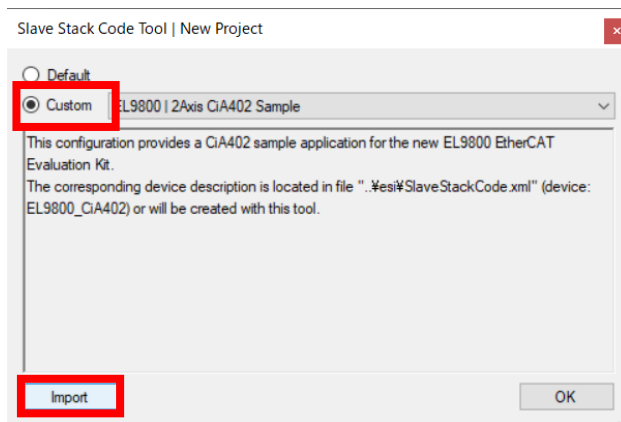
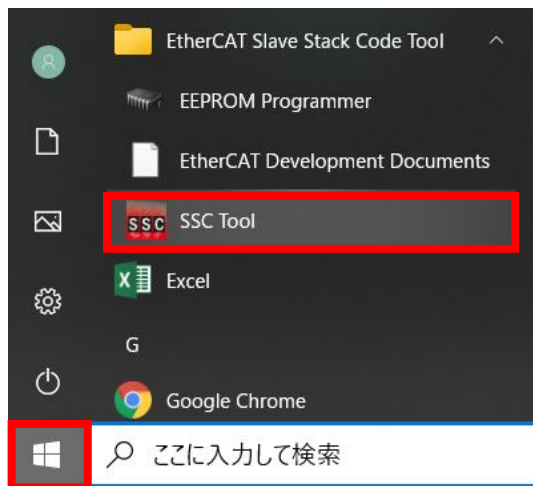
# SSC Toolによるコード生成 (1/6)

1. スタートメニューからSSC Toolを起動します。

**Note:** Check for Updates?の確認ダイアログが表示された場合はYesかNoを選択します。

2. New Projectのダイアログが表示されるので、Customを選択し、Importをクリックします。

**Note:** ダイアログが表示されない場合、File→Newを選択します。

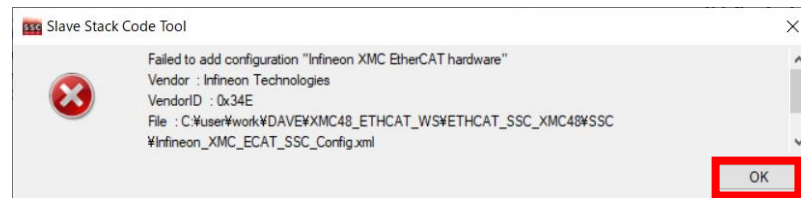
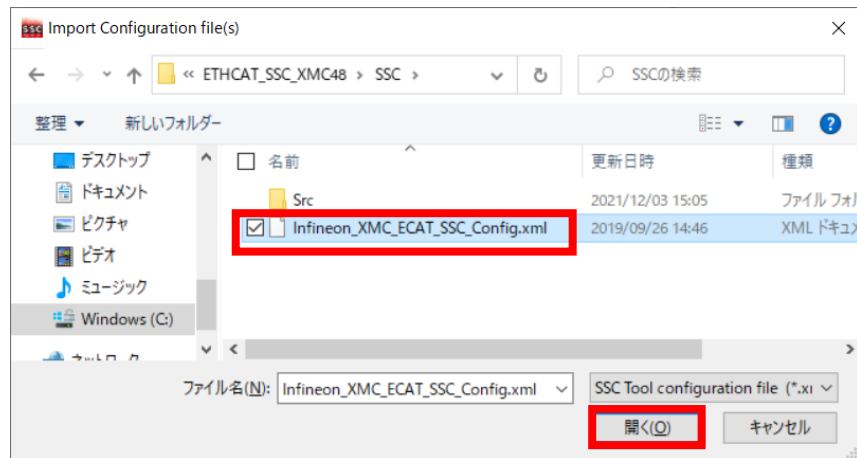


## SSC Toolによるコード生成 (2/6)

3. Import Configuration filesダイアログが表示されるので、XMC48\_ETHCAT\_WS¥ETHCAT\_SSC\_XMC48¥SSC以下にある Infineon\_XMC\_ECAT\_SSC\_Config.xmlを選択して開く(O)をクリックします。

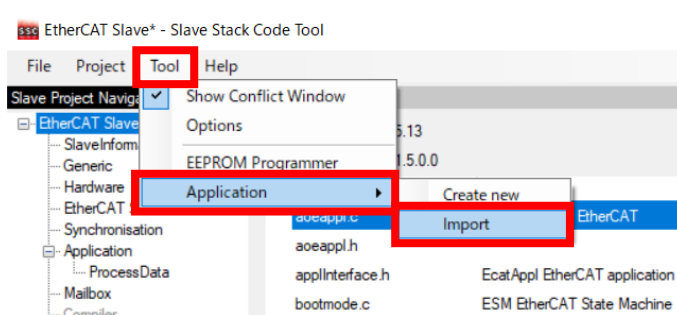
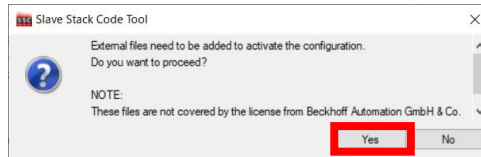
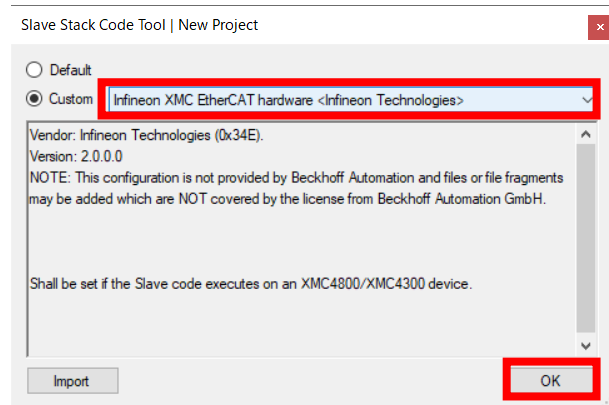
**Note:** XMLパーサーを使用するためPCにMicrosoft Excelがインストールされている必要があります。

**Note:** ExcelのバージョンによってはFailed to add configuration Errorが発生しますが、OKを押して続行します。



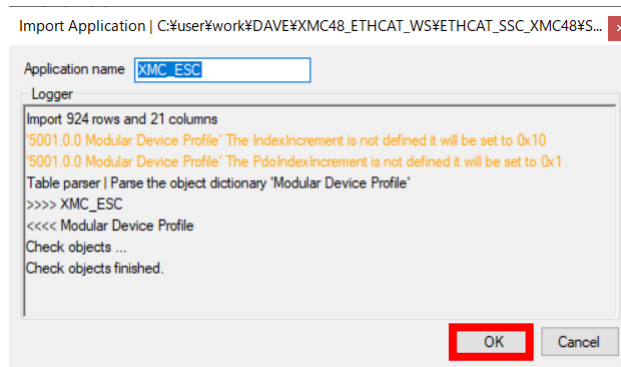
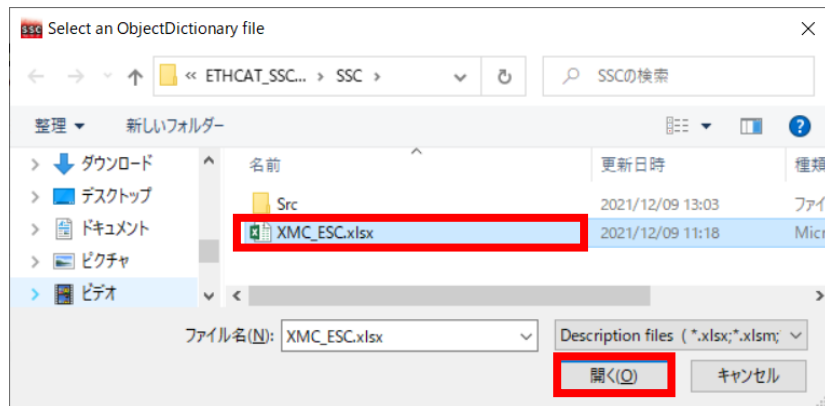
# SSC Toolによるコード生成 (3/6)

4. プルダウンリストからInfineon XMC EtherCAT hardware を選択しOKをクリックします。  
**Note:** Infineon XMC EtherCAT Slave sampleを選択しないでください
5. 続行するか尋ねるダイアログが表示されるのでYesをクリックします。
6. Tool → Application → Importを選択します。



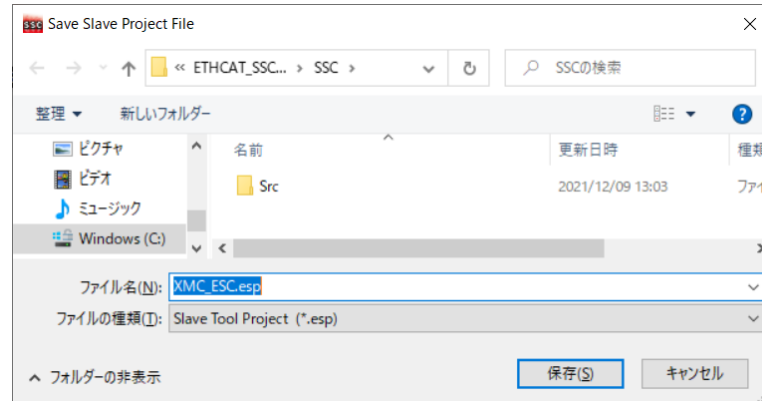
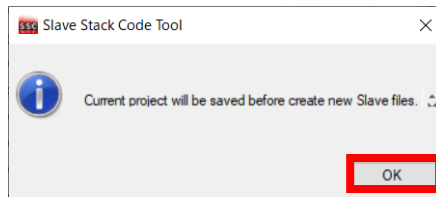
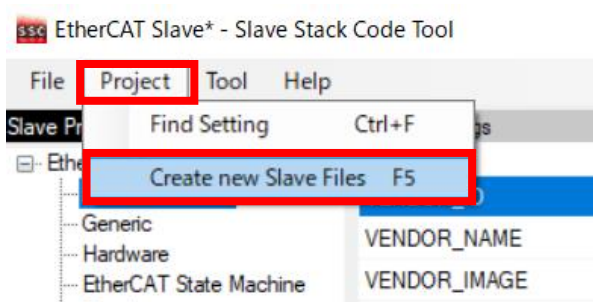
# SSC Toolによるコード生成 (4/6)

7. Select an ObjectDictionary fileのダイアログが表示されるので  
XMC48\_ETHCAT¥ETHCAT\_SSC\_XMC48¥SSC以下のXML\_ESC.xlsxを選択して  
開く(O)をクリックします。
8. Import Appricationのダイアログが表示されるのでOKをクリックします。



# SSC Toolによるコード生成 (5/6)

9. Project→Create new Slave Filesを選択します。
10. Current projectをSaveするダイアログが表示されるのでOKをクリックします。
11. XMC48\_ETHCAT¥ETHCAT\_SSC\_XMC48¥SSC以下で保存(S)をクリックしてXMC\_ESC.espファイルを保存します。



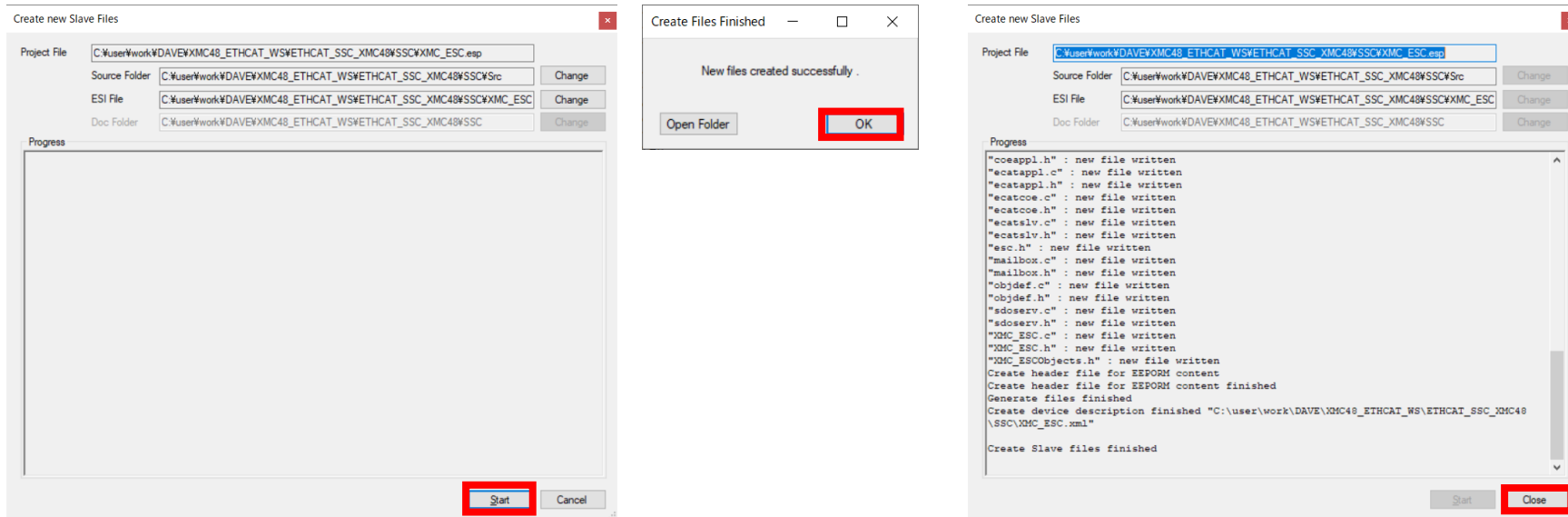
# SSC Toolによるコード生成 (6/6)

12. Create new Slave Filesのダイアログが表示されるのでStartをクリックします。

13. Create Files Finishedダイアログが表示されるのでOKをクリックします。

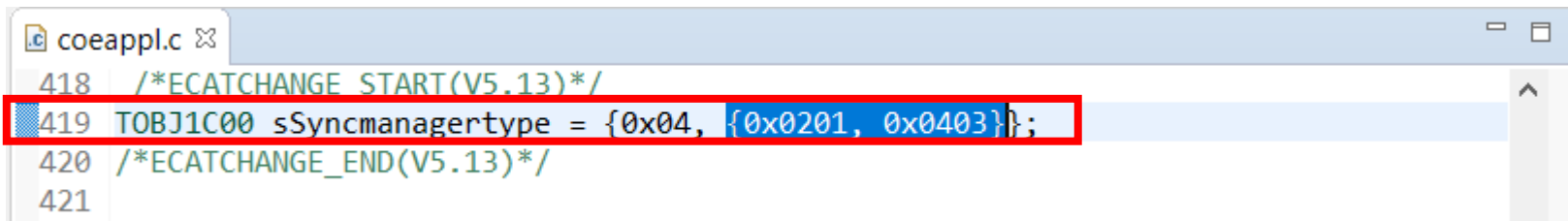
14. Create new Slave Filesのダイアログに戻るのでCloseをクリックします。

**Note:** Source Folderで指定された場所にコードが生成されます。また、ESI Fileで指定された場所に生成されたXMC\_ESC.xmlはTwinCAT3(EtherCAT Master)で使用されます。



# 生成されたコードcoeappl.cの確認と修正

1. Project explorer 内のETHCAT\_SSC\_XMC48¥SSC¥Src以下のcoeappl.cをダブルクリックして生成コードを表示します。
2. sSyncmanagertypeへ代入されている値が{0x04, {0x0201, 0x0403}}であることを確認し、異なっている場合は修正します。
3. ファイルを保存します。



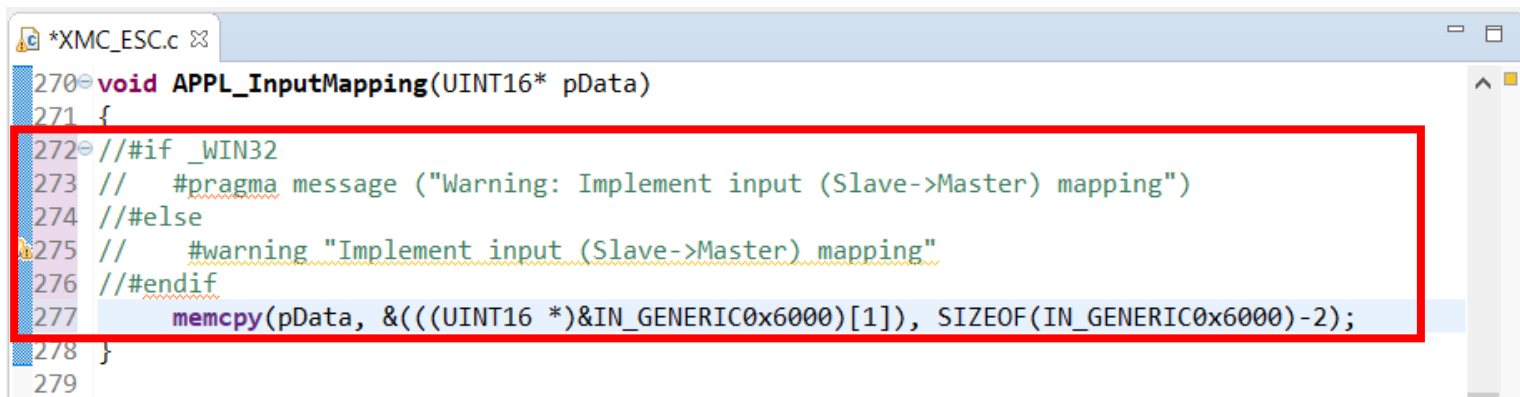
```

418  /*ECATCHANGE START(V5.13)*/
419  TOBJ1C00 sSyncmanagertype = {0x04, {0x0201, 0x0403}};
420  /*ECATCHANGE_END(V5.13)*/
421

```

# 生成されたコードXMC\_ESC.cの確認と修正(1/2)

1. Project explorer 内のETHCAT\_SSC\_XMC48¥SSC¥Src以下のXMC\_ESC.cをダブルクリックして生成コードを表示します。
2. APPL\_InputMapping()関数内のコードをコメントアウトし、下記のコードを追加します。  
memcpy(pData, &(((UINT16 \*)&IN\_GENERIC0x6000)[1]), sizeof(IN\_GENERIC0x6000)-2);



```

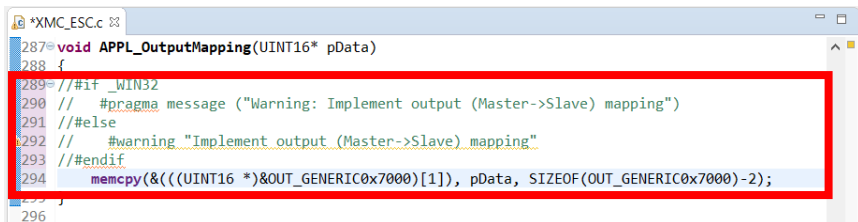
270 void APPL_InputMapping(UINT16* pData)
271 {
272 // #if _WIN32
273 // #pragma message ("Warning: Implement input (Slave->Master) mapping")
274 // #else
275 // #warning "Implement input (Slave->Master) mapping"
276 // #endif
277 memcpy(pData, &(((UINT16 *)&IN_GENERIC0x6000)[1]), sizeof(IN_GENERIC0x6000)-2);
278 }
279

```



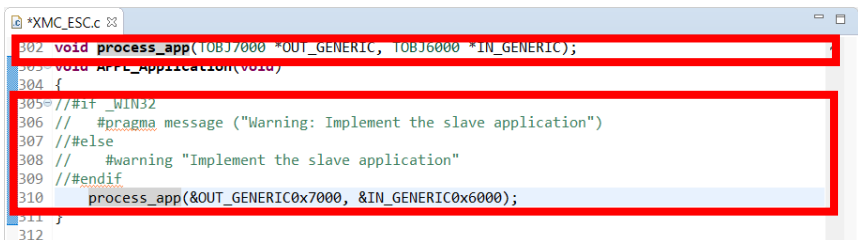
## 生成されたコードXMC\_ESC.cの確認と修正(2/2)

3. APPL\_OutputMapping()関数内のコードをコメントアウトし、下記のコードを追加します。  
memcpy(&(((UINT16 \*)&OUT\_GENERIC0x7000)[1]), pData, sizeof(OUT\_GENERIC0x7000)-2);



```
*XMC_ESC.c
287 void APPL_OutputMapping(UINT16* pData)
288 {
289 // #if _WIN32
290 // #pragma message ("Warning: Implement output (Master->Slave) mapping")
291 // #else
292 // #warning "Implement output (Master->Slave) mapping"
293 // #endif
294 memcpy(&(((UINT16 *)&OUT_GENERIC0x7000)[1]), pData, sizeof(OUT_GENERIC0x7000)-2);
295 }
296
```


4. APPL\_Application()関数直前に下記プロトタイプ宣言を追加し、関数内のコードをコメントアウトし、下記のコードを追加します。  
void process\_app(TOBJ7000 \*OUT\_GENERIC, TOBJ6000 \*IN\_GENERIC);  
process\_app(&OUT\_GENERIC0x7000, &IN\_GENERIC0x6000);

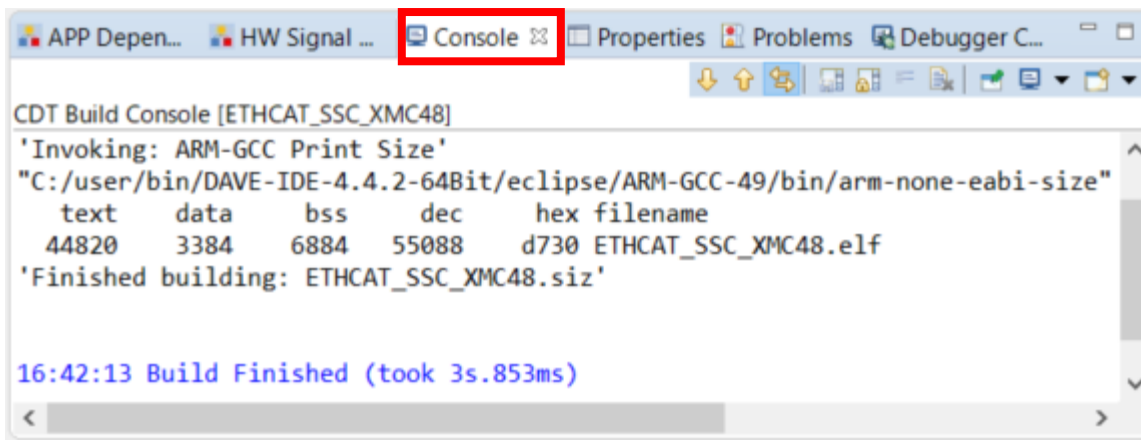


```
*XMC_ESC.c
302 void process_app(TOBJ7000 *OUT_GENERIC, TOBJ6000 *IN_GENERIC);
303 void APPL_Application(void)
304 {
305 // #if _WIN32
306 // #pragma message ("Warning: Implement the slave application")
307 // #else
308 // #warning "Implement the slave application"
309 // #endif
310 process_app(&OUT_GENERIC0x7000, &IN_GENERIC0x6000);
311 }
312
```

5. ファイルを保存します。

# プロジェクトのビルド

- › DAVEに戻りTool PanelのBuild Active projectアイコンをクリックしてプロジェクトをビルドします 
- › ビルド結果はAPP Dependency / APP ConnectivityのConsoleタブで確認できます。



```

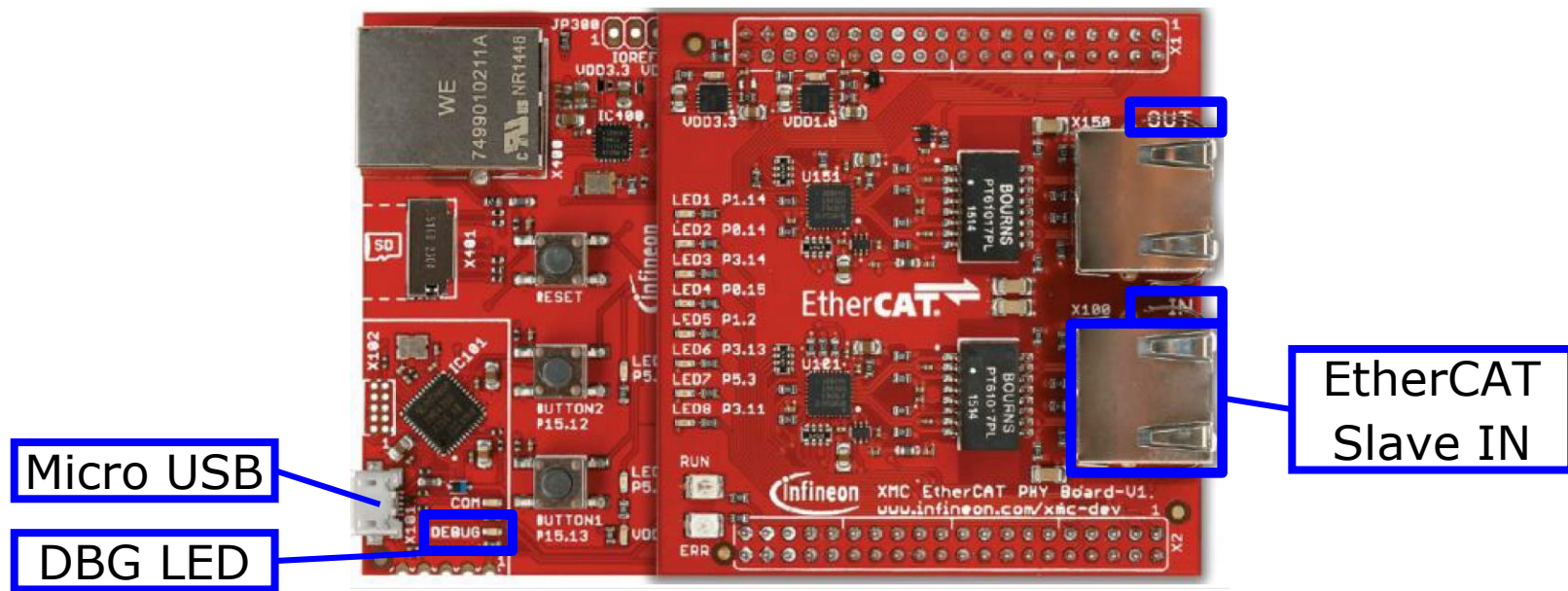
CDT Build Console [ETHCAT_SSC_XMC48]
'Invoking: ARM-GCC Print Size'
"C:/user/bin/DAVE-IDE-4.4.2-64Bit/eclipse/ARM-GCC-49/bin/arm-none-eabi-size"
  text    data    bss     dec     hex filename
 44820    3384    6884    55088    d730 ETHCAT_SSC_XMC48.elf
'Finished building: ETHCAT_SSC_XMC48.siz'

16:42:13 Build Finished (took 3s.853ms)
  
```


# 動作確認

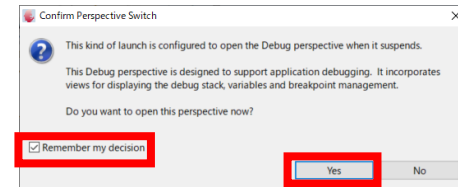
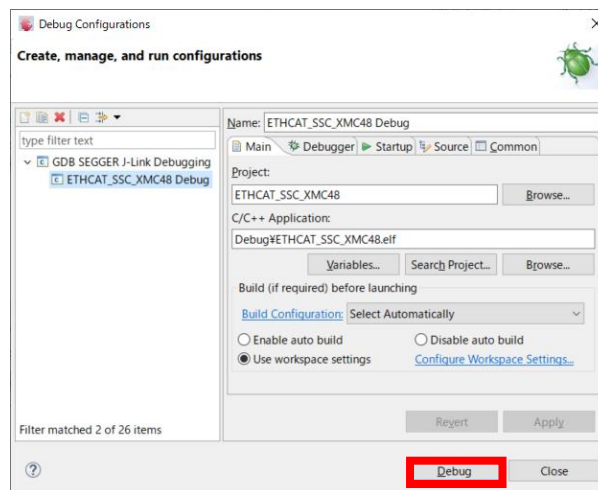
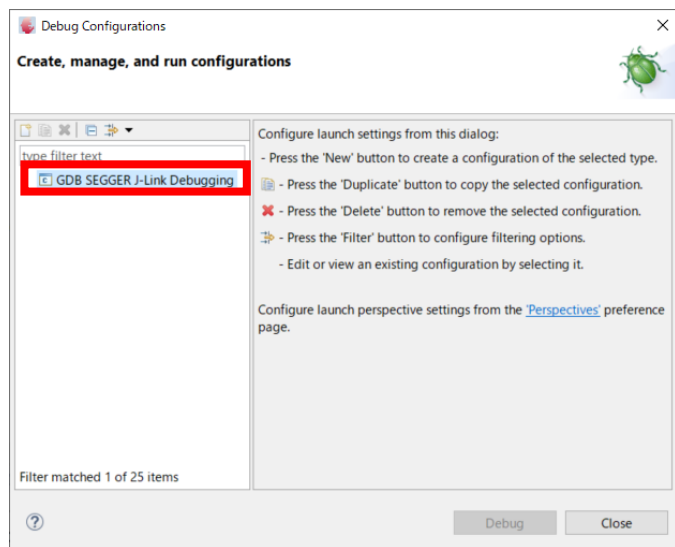
# ターゲットハードウェアの準備

- › DAVEを動作させているPCとKIT\_XMC48\_RELAX\_ECATCH\_V1 をMicro USBで接続し、DBG LEDが点灯していることを確認します。
- › LANケーブル(ストレート)を使用してEtherCAT Slave INとTwinCATがInstallされているPCのLANコネクタに接続します。



# Debuggerの起動

- ▶ Tool PanelのDebugアイコンをクリックしてDebugパースペクティブに移動します 
- 1. プロジェクトで初めてDebugを行う場合、Debug Configuration生成画面が表示されるのでGDB SEGGER J-Link Debuggingをダブルクリックします。
- 2. Debugをクリックします。Debugパースペクティブに移動の確認ダイアログが出た場合、“Remember my decision”をチェックしてYesを選択します。

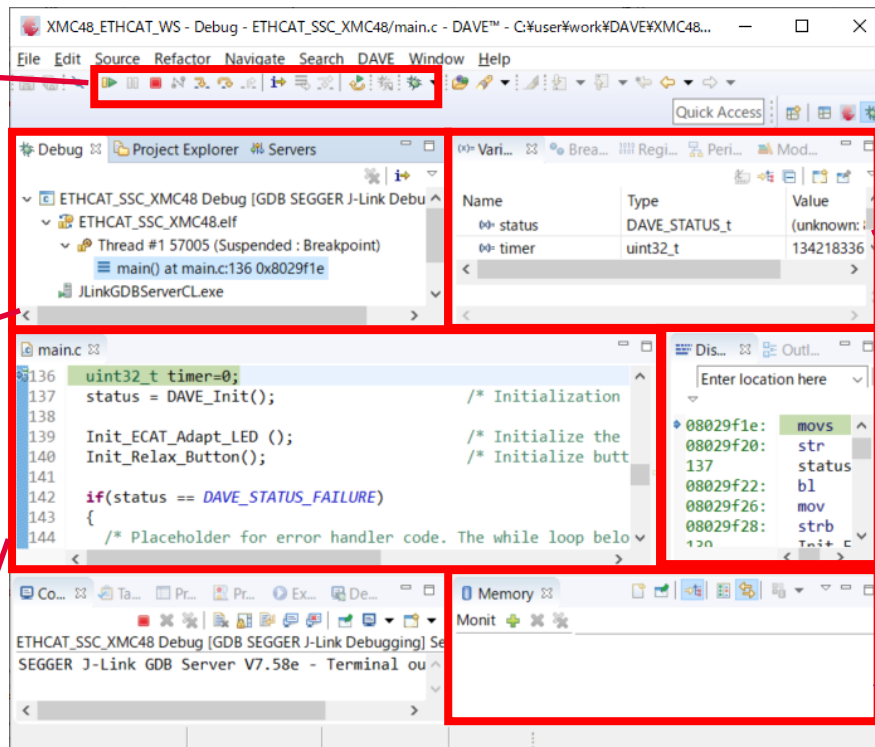


# Debugパースペクティブ各部説明

**Debugger Action**  
プログラムの実行/停止  
操作アイコン

**Debug View**  
Debugのスレッド情報  
表示

**Editor View**  
ソースコードレベルで  
の実行位置表示及びブ  
レークポイントの設定/  
解除



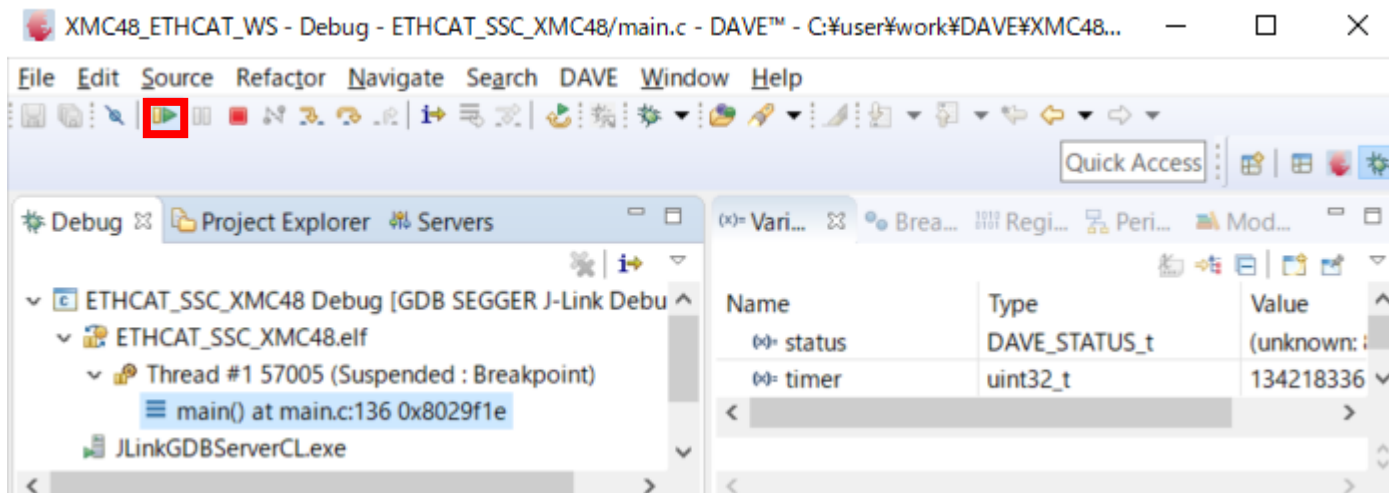
**Variable View**  
実行中の変数監視

**Disassembly View**  
機械語レベルでの実行  
位置表示

**Memory View**  
メモリの監視及び変  
更

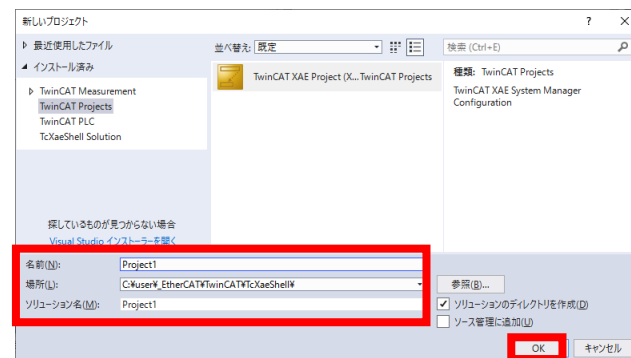
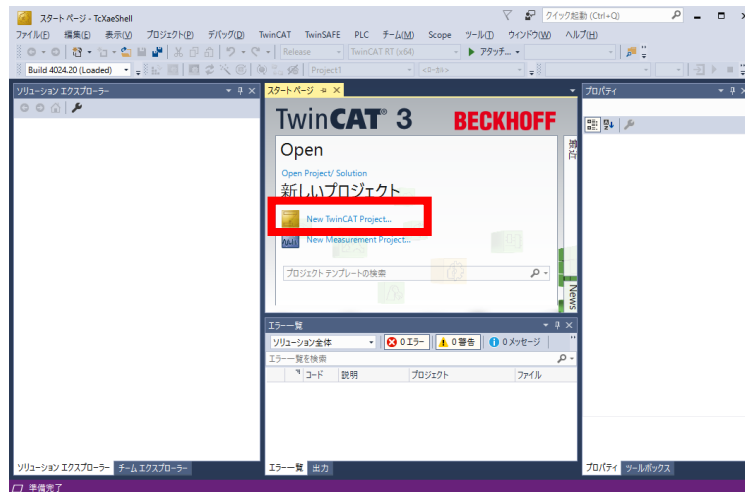
# Slave側の起動

- › Debugger Action のResume アイコンをクリックします 



# TwinCAT3による動作確認(1/10)

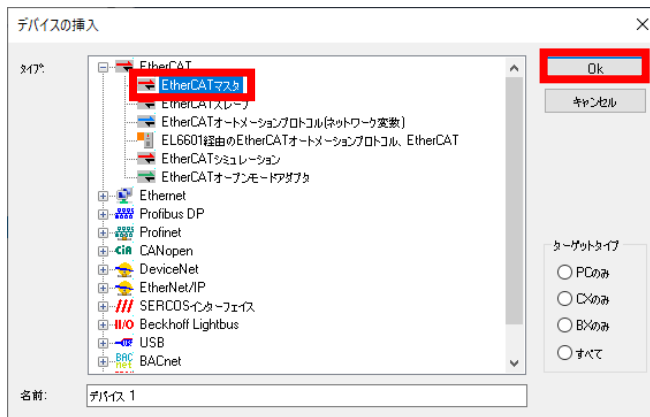
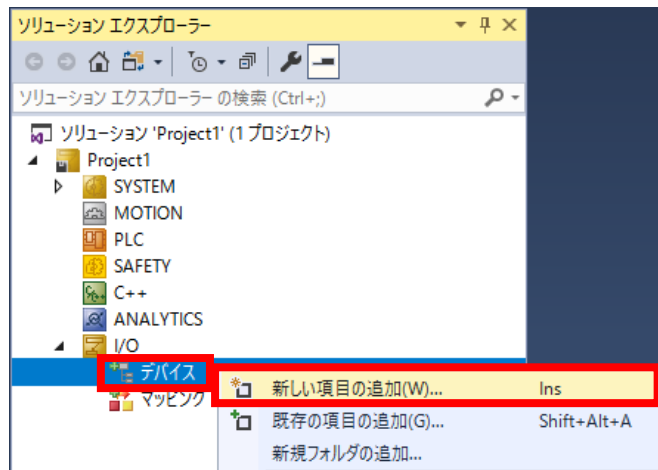
1. スタートメニューからTwinCAT XAE Shellを起動します。
2. ファイル→新規作成→プロジェクトを選択又はスタートページのNew TwinCAT Projectをクリックします。
3. 新しいプロジェクトのダイアログが表示されるので名前と格納場所を指定しOKをクリックします。





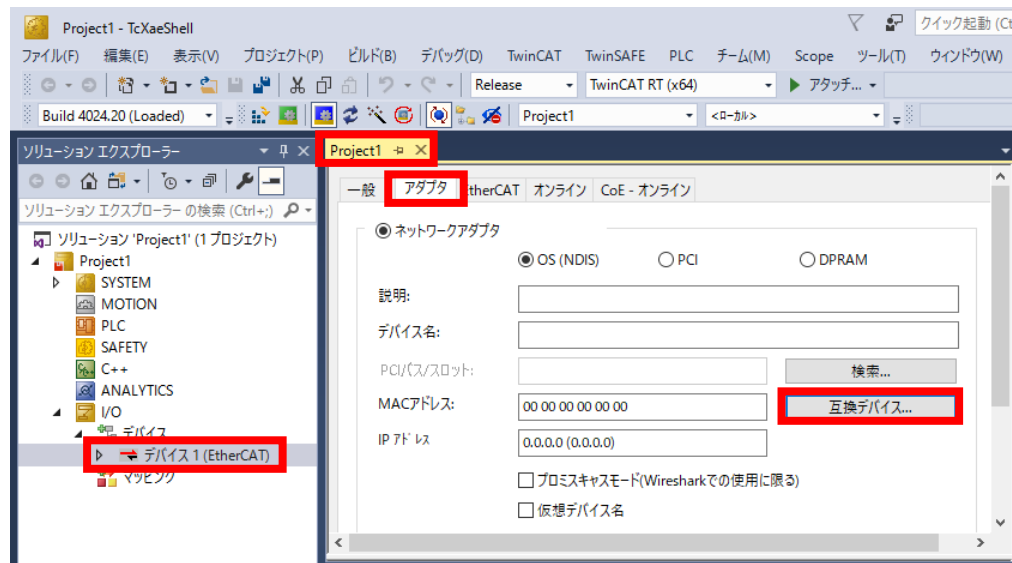
# TwinCAT3による動作確認(2/10)

4. ソリューションエクスプローラのI/O→デバイスを選択し、右クリックして新しい項目の追加を選択します。
5. デバイスの挿入ダイアログが表示されるのでEtherCATマスタを選択し、OKをクリックします。



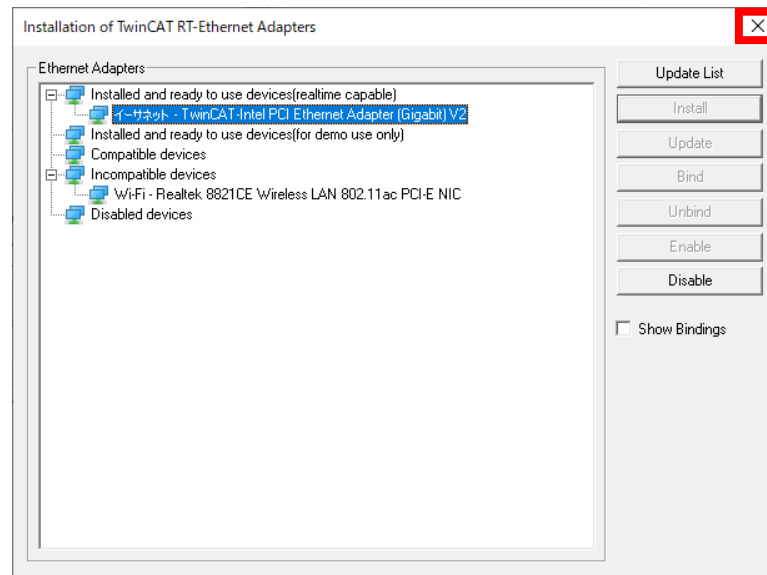
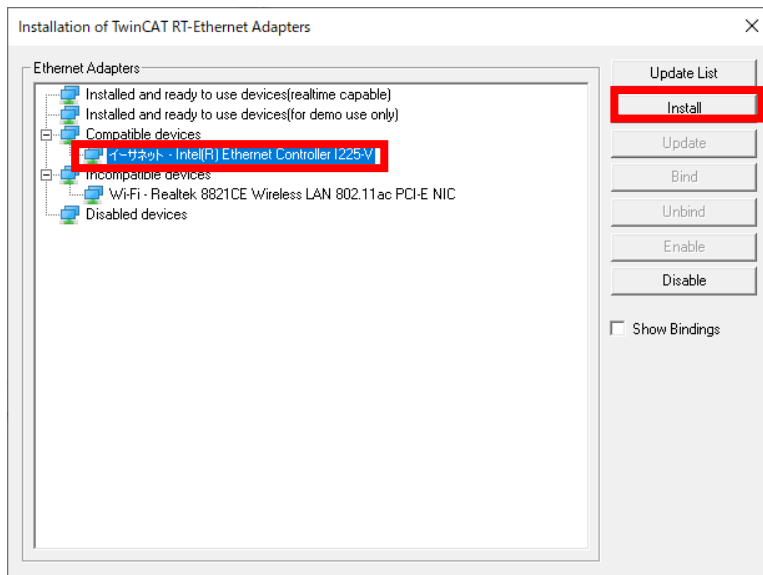
# TwinCAT3による動作確認(3/10)

6. ソリューションエクスプローラに追加されたデバイス1 (EtherCAT)をダブルクリックし、Project1を表示します。
7. Project1のアダプタのタブをクリックし、互換デバイスをクリックします。



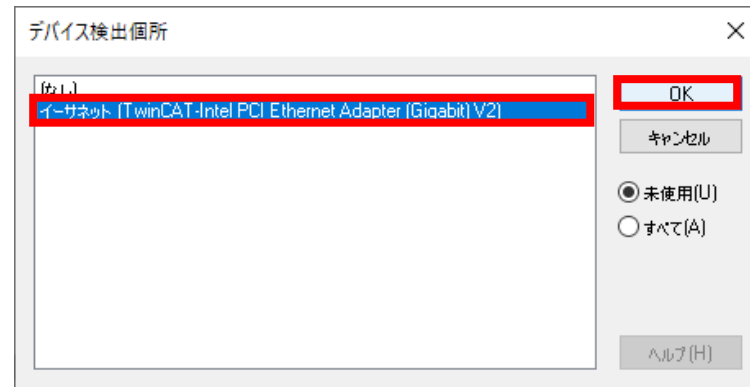
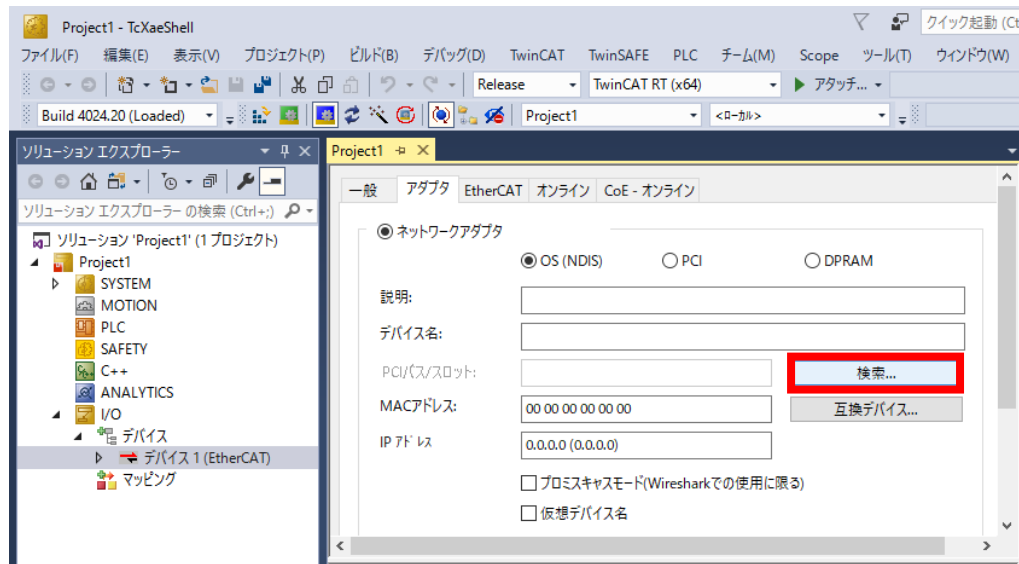
# TwinCAT3による動作確認(4/10)

8. Installation of TwinCAT RT-Ethernet Adaptersのダイアログが表示されるので、Compatible devicesのデバイスを選択しInstallをクリックします。
9. 選択したデバイスがInstalled and ready to use devicesに移動したことを確認してダイアログを閉じます。



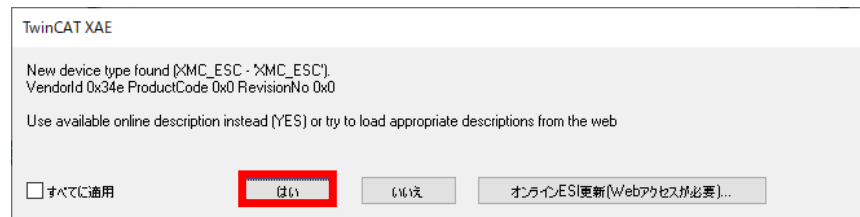
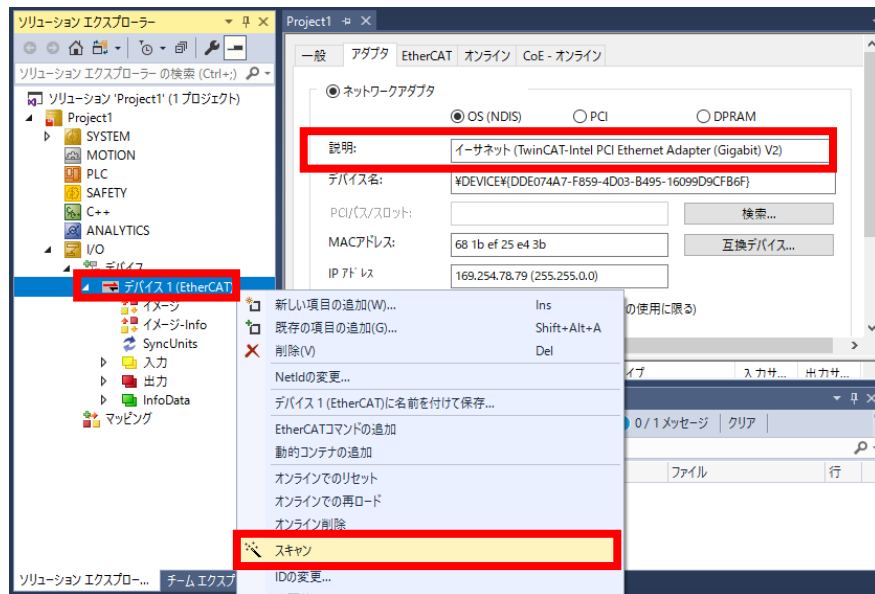
# TwinCAT3による動作確認(5/10)

10. Project1のアダプタのタブで検索をクリックします。
11. イーサネットアダプタを選択し、OKをクリックします。



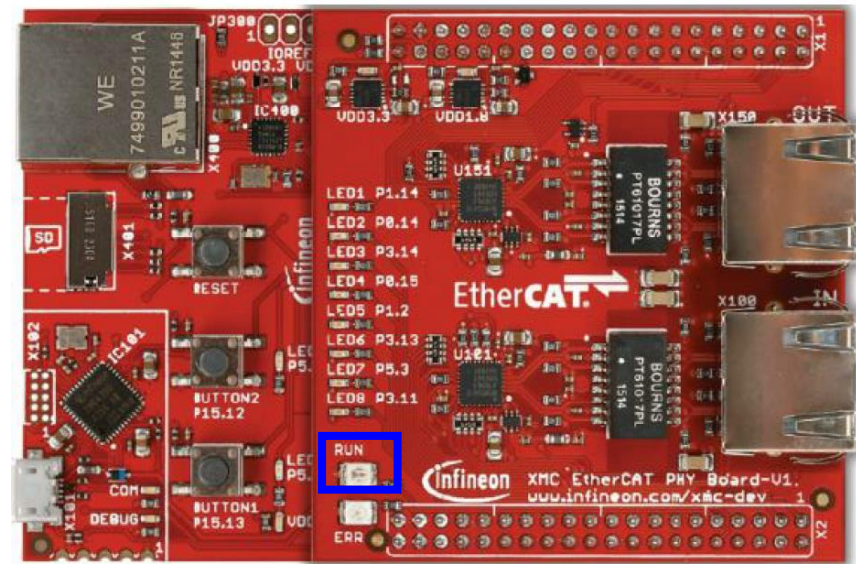
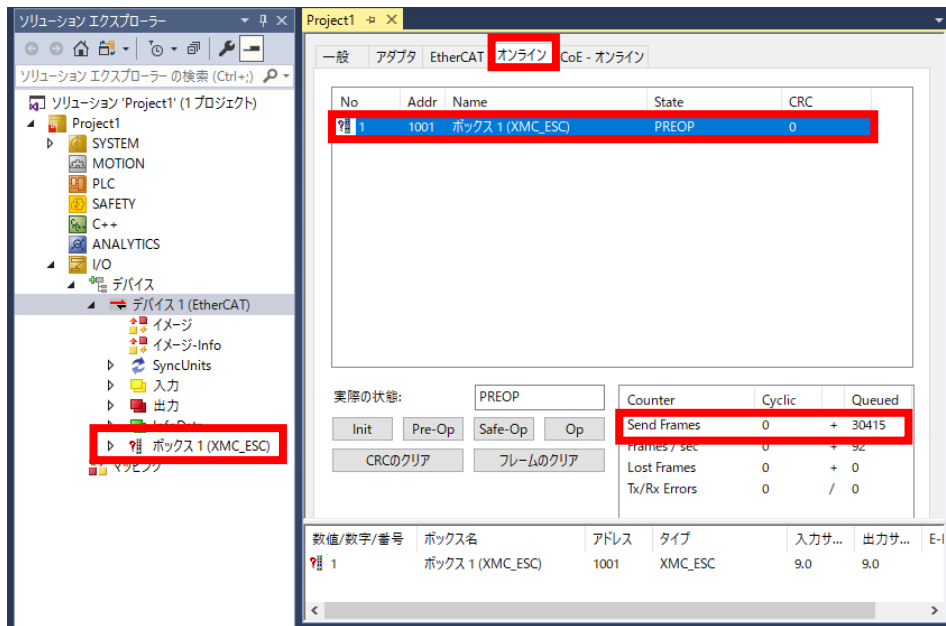
# TwinCAT3による動作確認(6/10)

12. Project1のアダプタのタブで説明の項目が選択したイーサネットアダプタになっていることを確認します。
13. ソリューションエクスプローラのデバイス1(EtherCAT)を右クリックしてスキャンを選択します。
14. New device type found(XMC\_ESC)のダイアログが表示されるので“はい”をクリックします。




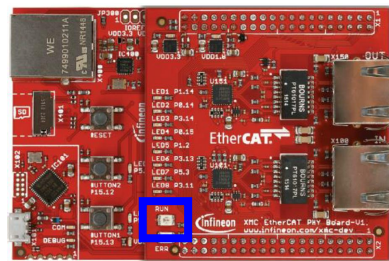
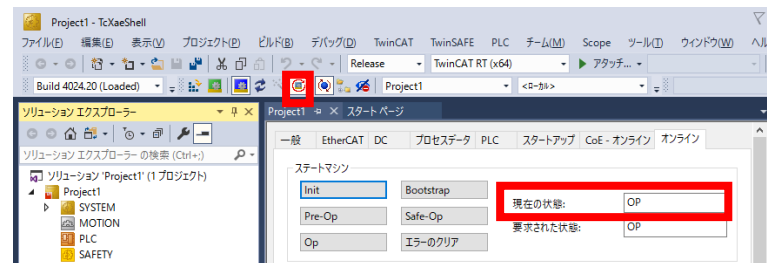
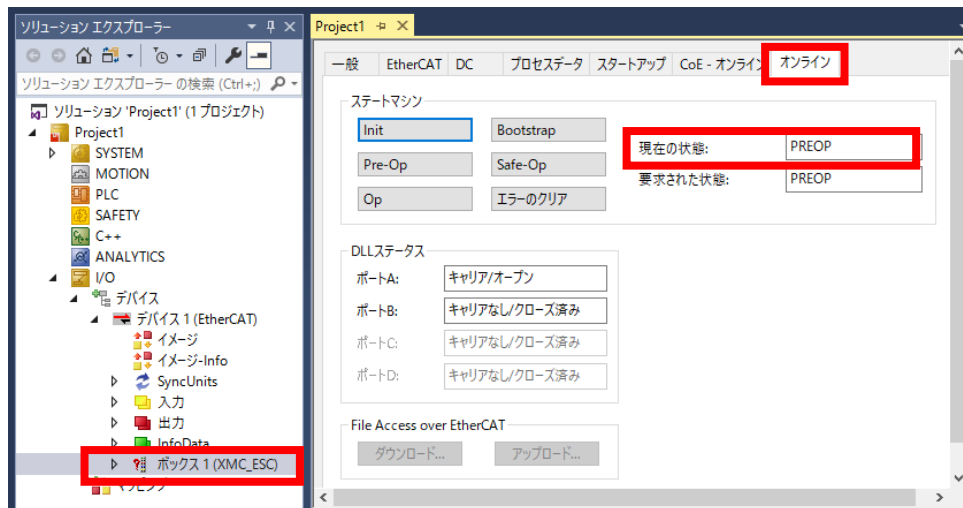
# TwinCAT3による動作確認(7/10)

15. Project1のオンラインのタブでボックス1 (XMC\_ESC)が追加されている事を確認し、Send framesの値が加算されていく事を確認します。
16. KIT\_XMC48\_RELAX\_ECAT\_V1のRUN LEDが点滅していることを確認します。



# TwinCAT3による動作確認(8/10)

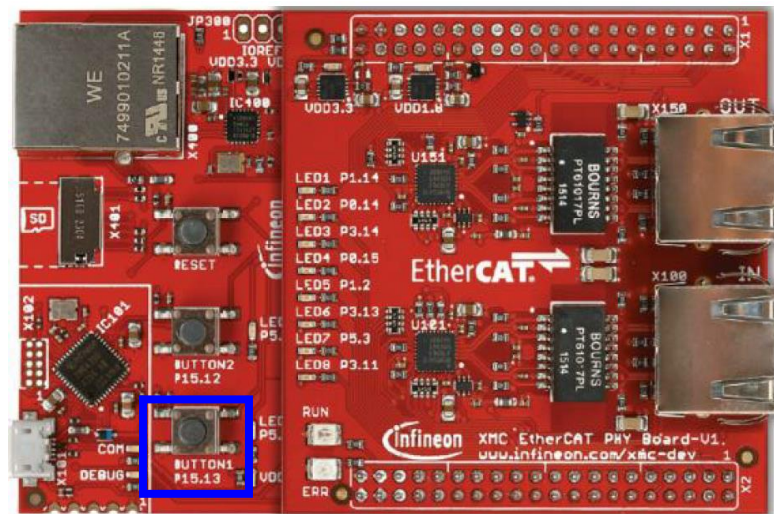
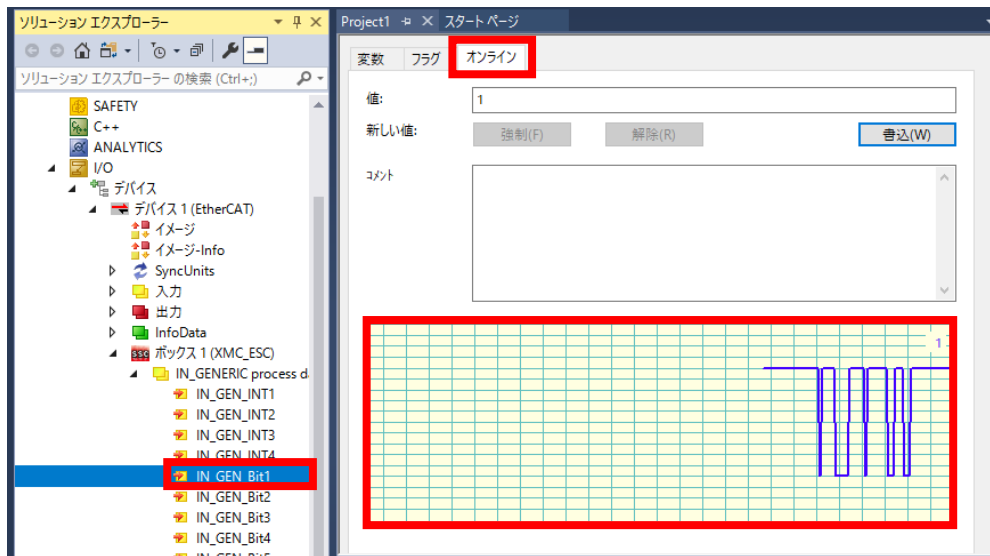
17. ソリューションエクスプローラのボックス1(XMC\_ESC)を選択し、Project1ののオンラインのタブで現在の状態がPREOPになっていることを確認します。
18. Free Runボタンをクリックして現在の状態がOPに変わる事、RUN LEDが点滅から点灯に変わる事ことを確認します。 





# TwinCAT3による動作確認(9/10)

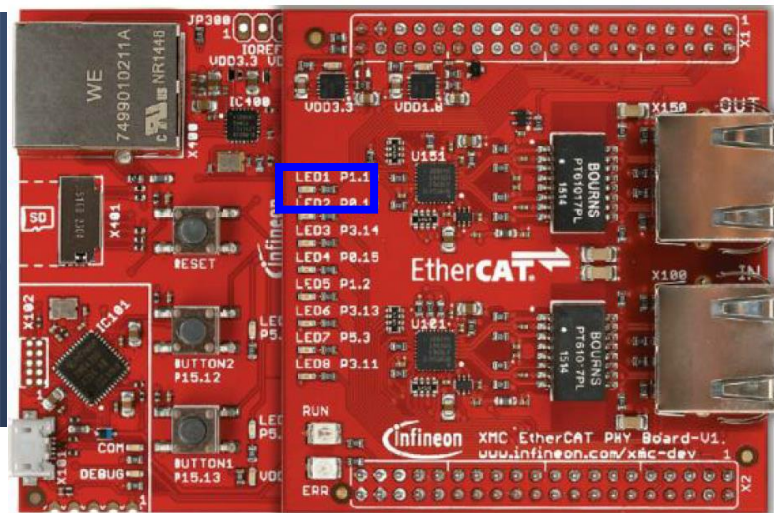
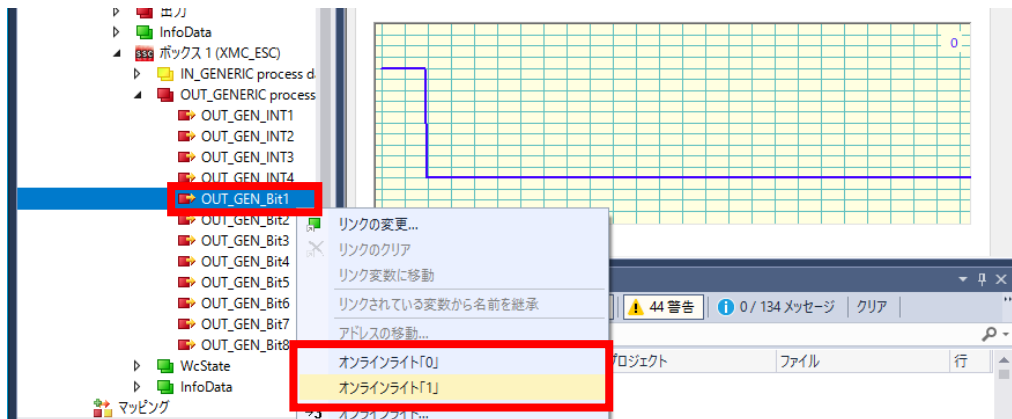
19. ソリューションエクスプローラのボックス1(XMC\_ESC)以下の  
IN\_GENERIC→IN\_GEN\_Bit1を選択し、Project1のオンラインタブをクリックします。
20. KIT\_XMC48\_RELAX\_ECAT\_V1のBUTTON1を押す/離すを繰り返し、ボタンの状態  
がグラフに反映さえる事を確認します。





# TwinCAT3による動作確認(10/10)

19. ソリューションエクスプローラのボックス1(XMC\_ESC)以下のOUT\_GENERIC→OUT\_GEN\_Bit1を選択し、右クリックしてオンラインライト「1」/オンラインライト「0」を選択し、KIT\_XMC48\_RELAX\_ECAT\_V1のLED1がオンラインライト「1」を選択した際に点灯する事を確認します。



# Appendix



Part of your life. Part of tomorrow.