
ROBIN-M7 対応
MATLAB/Simulink 実稼働システム
Pass/ROBIN WPF 版
型式 PASS-ROBIN-WPF
インストール説明書

2024 年 12 月 24 日

Ver. 1. 0. 0

目次

1 概要	1
1-1 ARM開発環境インストール手順.....	2
1-2 MATLABインストール手順.....	6
1-3 SDKインストール手順.....	11
1-4 Pass/ROBINインストール手順.....	19
1-5 Pass/ROBIN環境設定手順.....	24
1-6 Pass/ROBIN MATLAB検索パス設定手順.....	27
2 アンインストール手順.....	28
3 インストールファイル一覧.....	28

1 概要

本書は、ROBIN 用ソフトウェア「Pass/ROBIN WPF 版」のインストール説明書です。

Pass/ROBIN WPF 版（以下、Pass/ROBIN と表記）のインストールは、以下の手順で行います。
 なお、各ソフトウェアのインストール先及び Windows のユーザ名に 2 バイト文字を含む場合、
 正常に動作しない可能性がありますのでご注意ください。

表 1-1 Pass/ROBIN インストール手順

手順	項目
1	ARM 開発環境インストール
2	MATLAB インストール
3	SDK インストール
4	Pass/ROBIN インストール
5	Pass/ROBIN 環境設定
6	MATLAB 検索パス設定

1-1 ARM 開発環境インストール手順

ARM 開発環境である GNU Arm Embedded Toolchain のインストール手順を説明します。

本書では Windows10 環境で GNU Arm Embedded Toolchain Version 10.3 をインストールする手順を説明します。

※特に支障がなければ、Version 10.3 をインストールしてください。

※OS 及び GNU Arm Embedded Toolchain のバージョンにより画面表示等は異なる場合があります。

GNU Arm Embedded Toolchain についてのお問い合わせは、弊社では回答いたしかねますのでご了承ください。

- (1) GNU Arm Embedded Toolchain のダウンロードページ (<https://developer.arm.com/downloads/-/gnu-rm>) を開きます。
- (2) 「gcc-arm-none-eabi-10.3-2021.10-win32.exe」をダウンロードします。
※「win32」と記載されていますが、Windows 64bit 環境でも問題ありません。

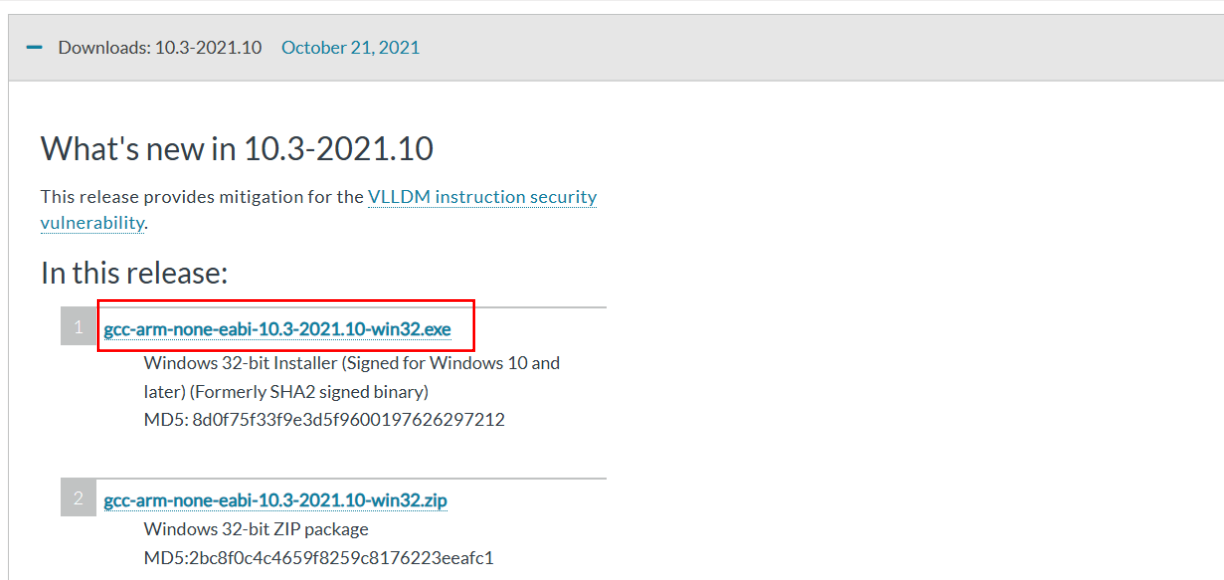


図 1-1 GNU Arm Embedded Toolchain ダウンロードページ

- (3) (2)でダウンロードしたファイルを実行します。

- (4) 「Japanese」を選択して OK ボタンを押します。

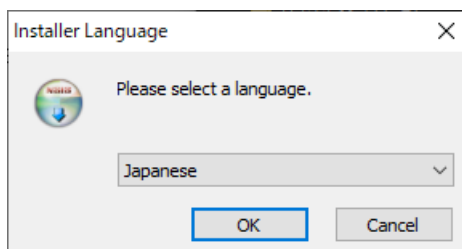


図 1-2 インストーラの言語選択画面

- (5) 「次へ」ボタンを押します。



図 1-3 セットアップ開始画面

- (6) ライセンス契約書を確認し、「同意する」ボタンを押します。

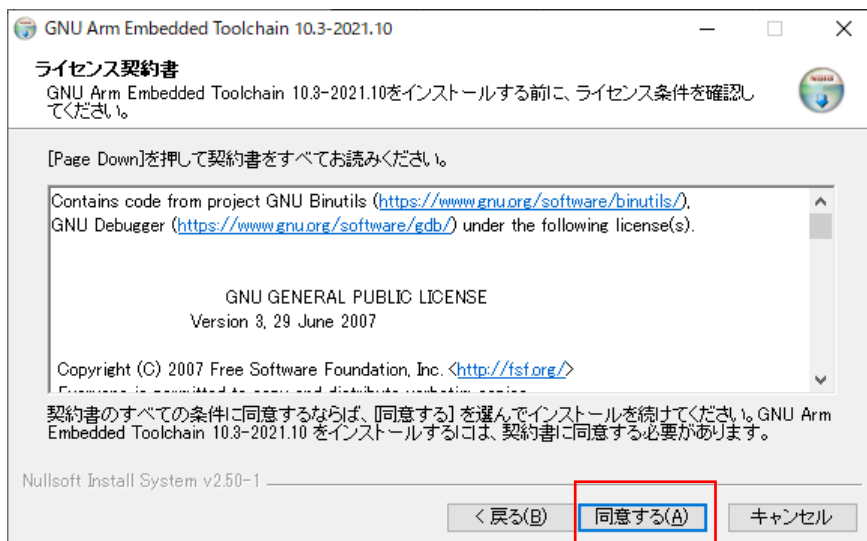


図 1-4 ライセンス契約書確認画面

- (7) インストール先を選択し、「インストール」ボタンを押します。
 ※特に支障がなければ、インストール先はデフォルトのままご利用ください。

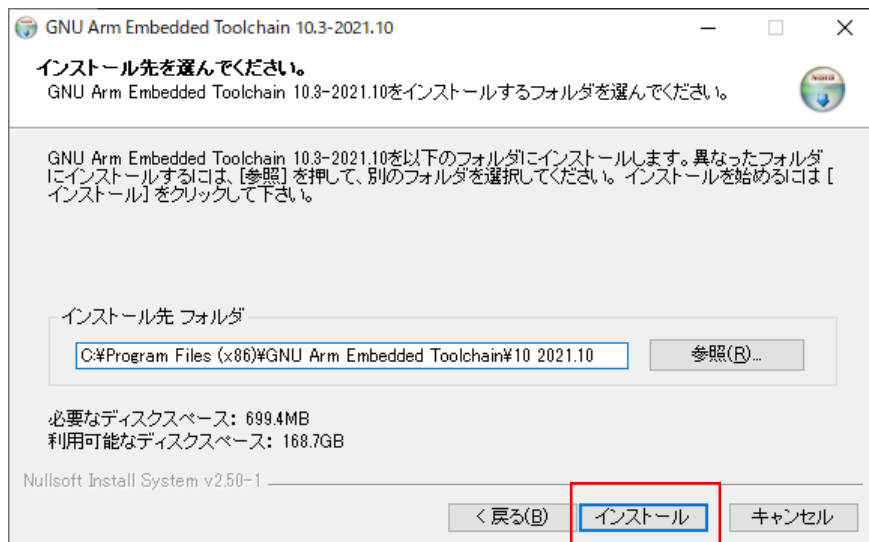


図 1-5 インストール先設定画面

- (8) インストール完了までお待ちください。

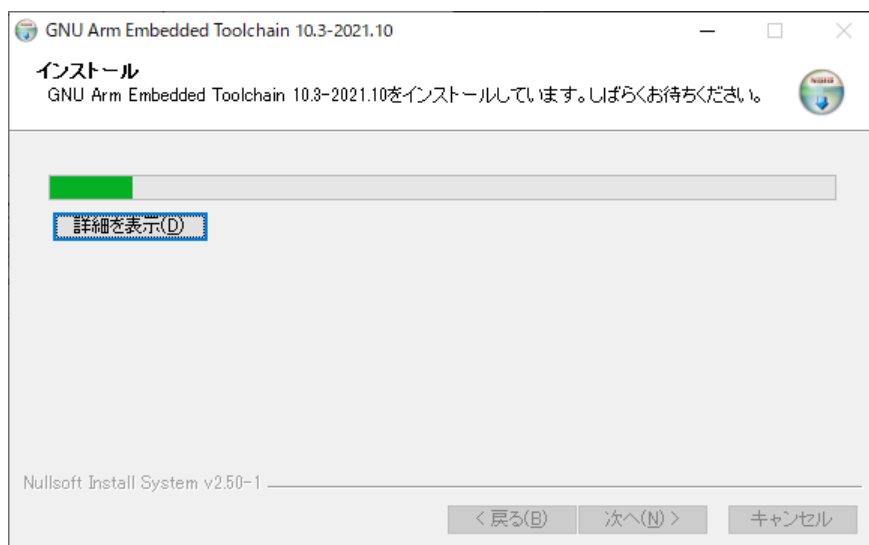


図 1-6 インストール状況確認画面

- (9) インストールが完了しました。「完了」ボタンを押してください。

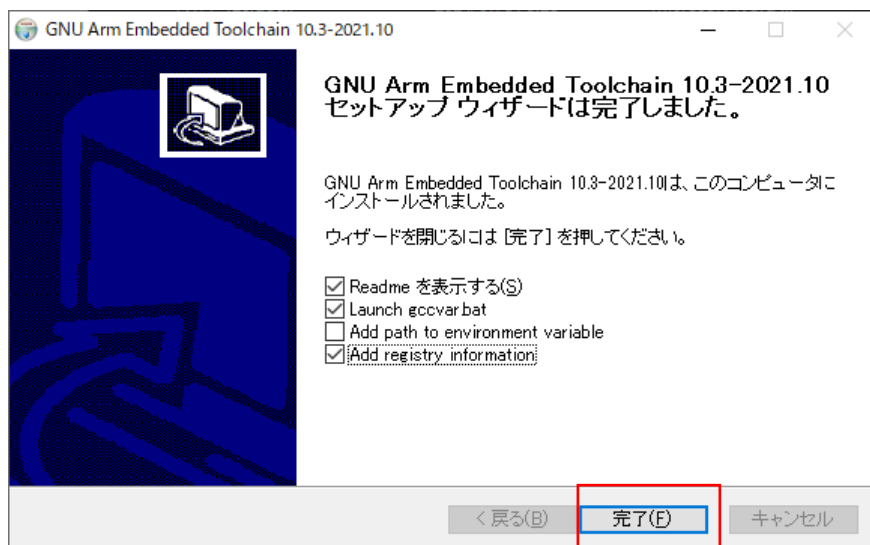


図 1-7 インストール完了画面

1-2 MATLAB インストール手順

MATLAB のインストール手順を説明します。

なお、本書では弊社で所持している MATLAB ライセンスを使用してウェブインストーラでインストールする場合の手順を例として示します。

MATLAB のライセンス形態やバージョンによってはインストール手順が本書の記載とは大きく異なる可能性があります。お客様が所持されている MATLAB ライセンスのインストール方法については弊社での対応をいたしかねます。

インストールについてご不明点がある場合は MathWorks 社にお問い合わせください。

(1) MATLAB のウェブインストーラを実行します。

(2) MathWorks アカウントでログインします。

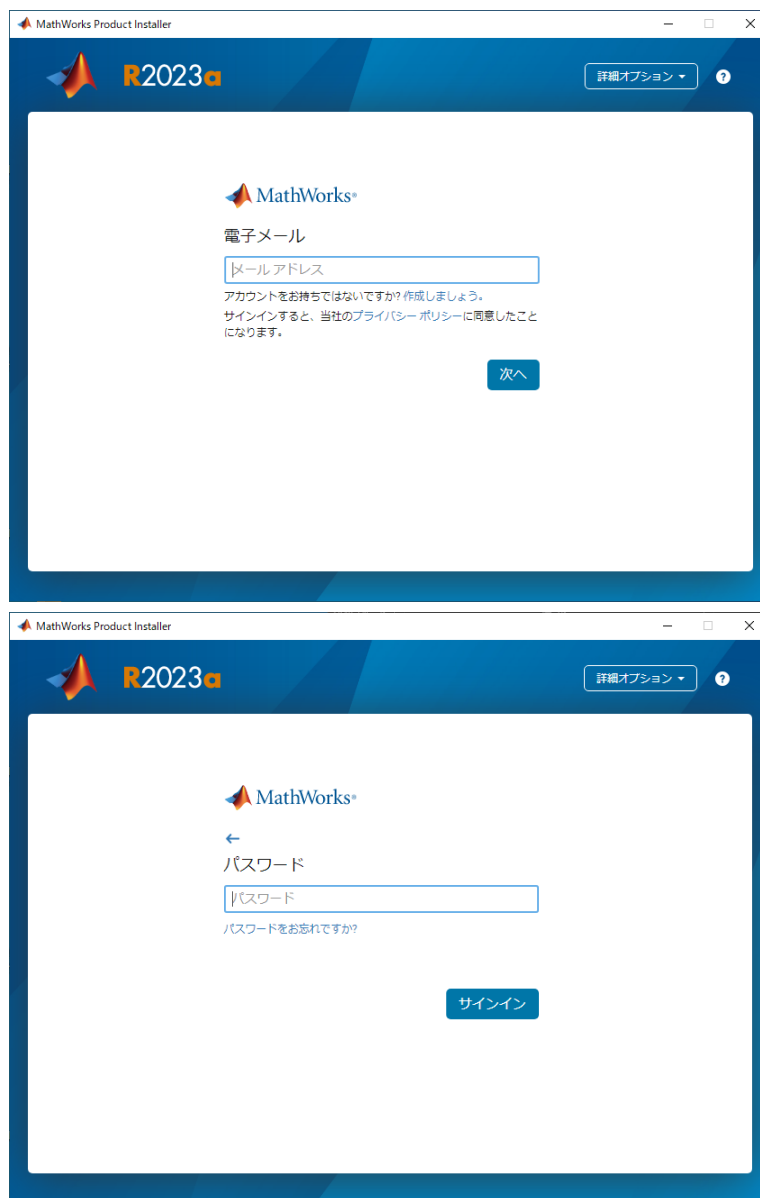


図 1-8 MathWorks アカウントでログイン

- (3) 「はい」を選択し、「次へ」をクリックします。

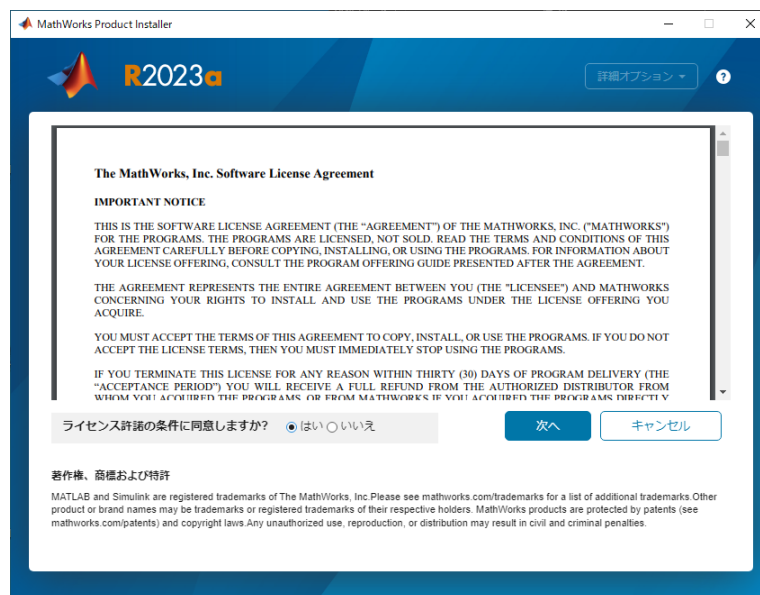


図 1-9 ライセンス許諾

- (4) 使用するライセンスを選択し、「次へ」をクリックします。



図 1-10 ライセンス選択

- (5) ユーザを確認し、「次へ」をクリックします。



図 1-11 ユーザの確認

- (6) インストール先を選択し、「次へ」をクリックします。
インストール先を変更する場合、漢字／カナ等の 2 バイトコードを含むフォルダにはインストールしないでください。



図 1-12 インストールフォルダ選択

(7) インストールする製品を選択します。Pass/SEAGULL の使用には下図の 4 製品が必要です。選択できない製品がある場合、弊社担当営業までご連絡ください。



図 1-13 インストール製品選択

(8) インストールオプションを選択し、「次へ」をクリックします。



図 1-14 インストールオプションタイプ選択

- (9) インストールの最終確認を行います。問題がなければ「インストールの開始」をクリックします。



図 1-15 インストール設定確認

- (10) インストールを実行します。
- インストール終了後、MATLAB Coder 及び Simulink Coder の機能の一部に Visual C++が必須となるため、以下の画面を表示します。
- Pass/SEAGULL で使用する機能には Visual C++は必要ないため、このメッセージを無視しても問題はありません。



図 1-16 コンパイラ警告

- (11) MATLAB のアクティベーションを実行します。

1-3 SDK インストール手順

MCUXpresso SDK for EVK-MIMX8MP（SDK）のインストール手順を説明します。

本書では Windows10 環境で SDK Version 2.15.0 をインストールする手順を説明します。

※特に支障がなければ、Version 2.15.0 をインストールしてください。

※OS 及び SDK のバージョンにより画面表示等は異なる場合があります。

(1) SDK のダウンロードページ (<https://mcuxpresso.nxp.com/en>) を開きます。

(2) 「Select Development Board」を押します。

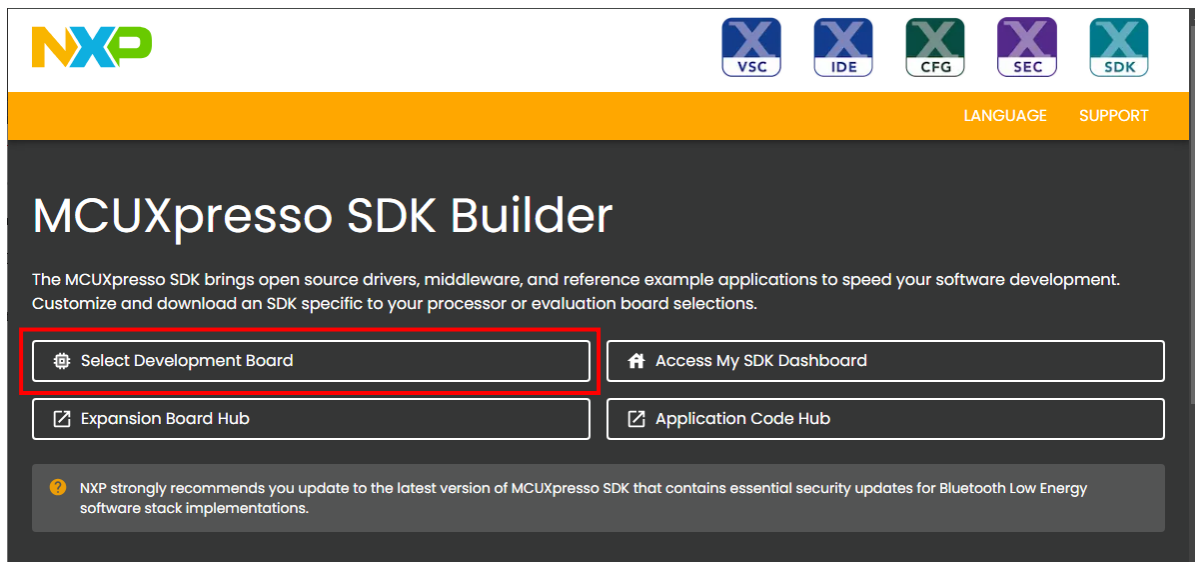


図 1-17 SDK ダウンロードページ

(3) サインインします（アカウントがない場合は作成してください）。

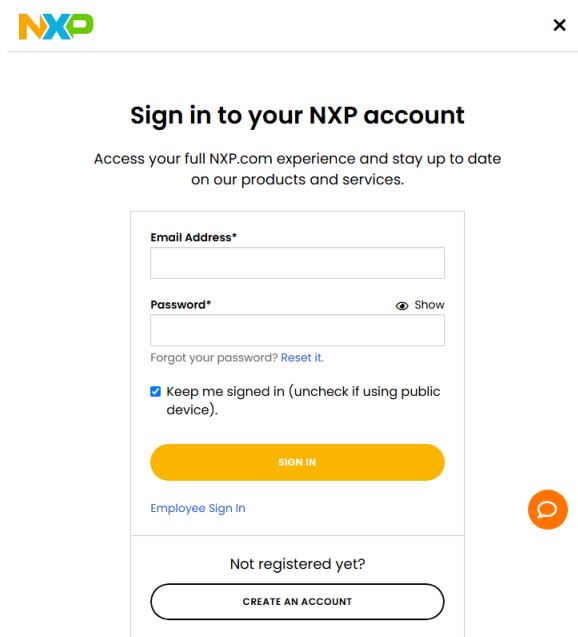


図 1-18 サインイン画面

(4) 「Processors」を選択します。

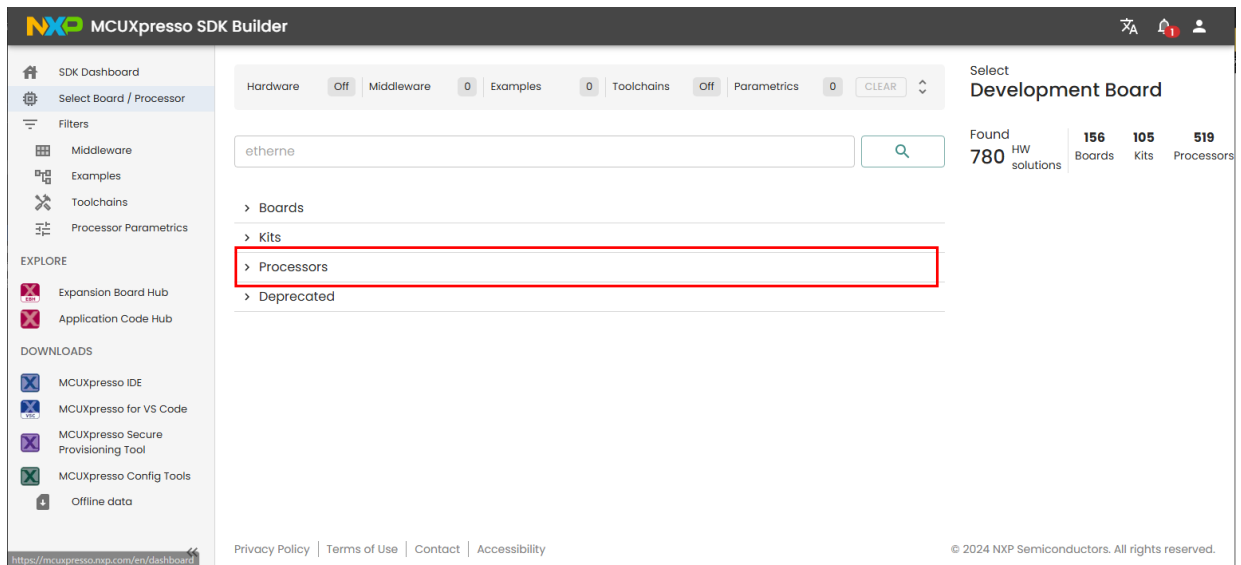


図 1-19 SDK カスタマイズ画面 1

(5) 「i.MX」を選択します。

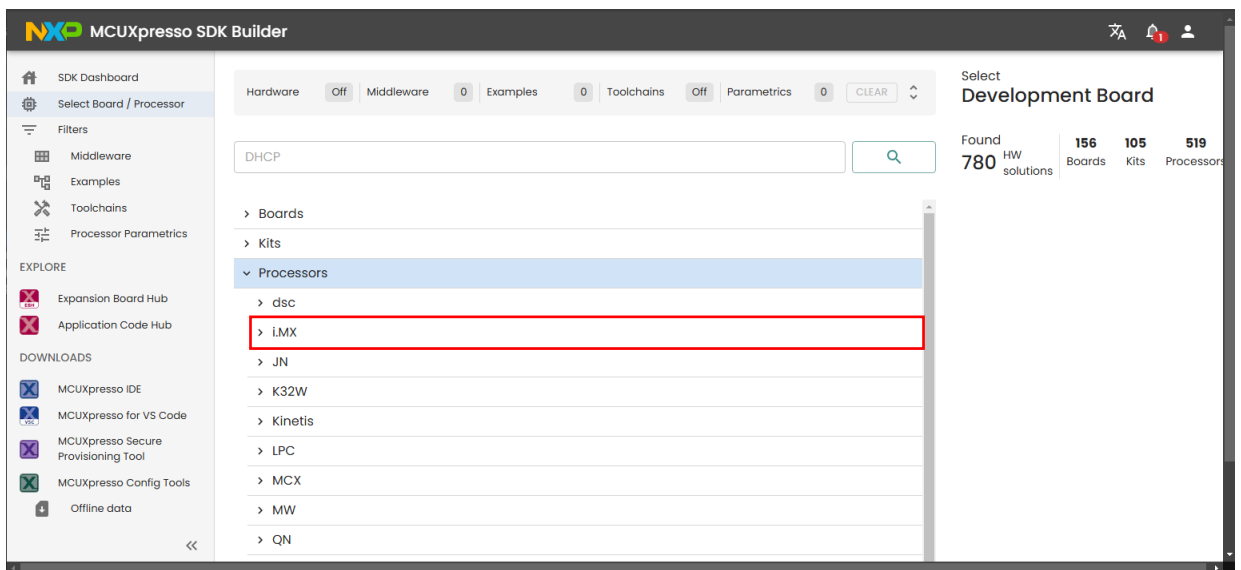


図 1-20 SDK カスタマイズ画面 2

(6) 「8M Plus Quad」を選択します。

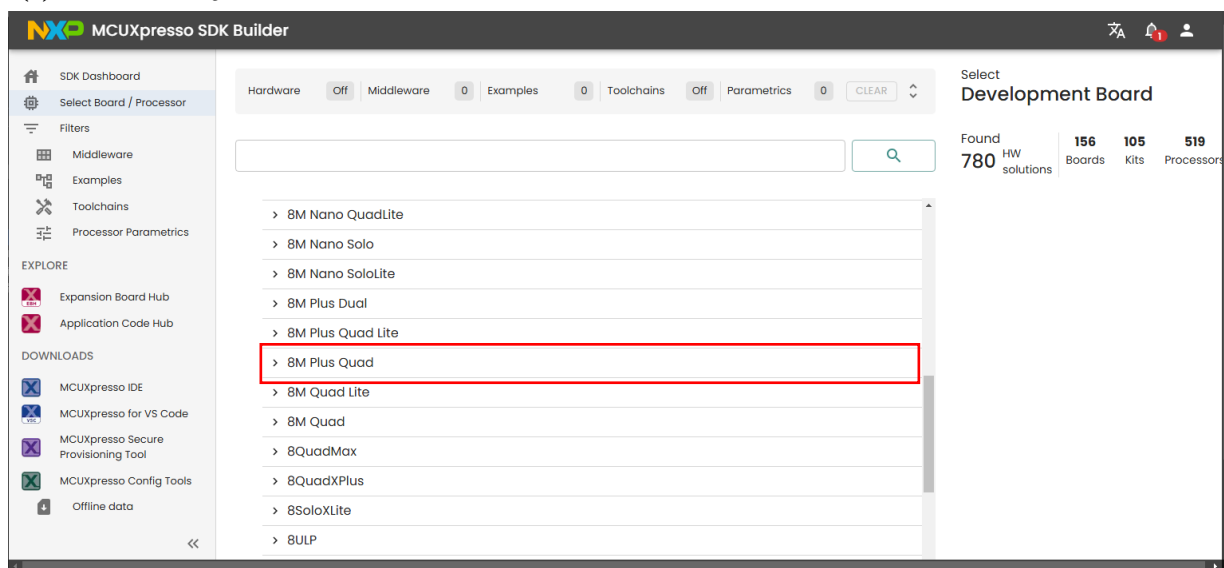


図 1-2 1 SDK カスタマイズ画面 3

(7) 「MIMX8MLx」を選択します。

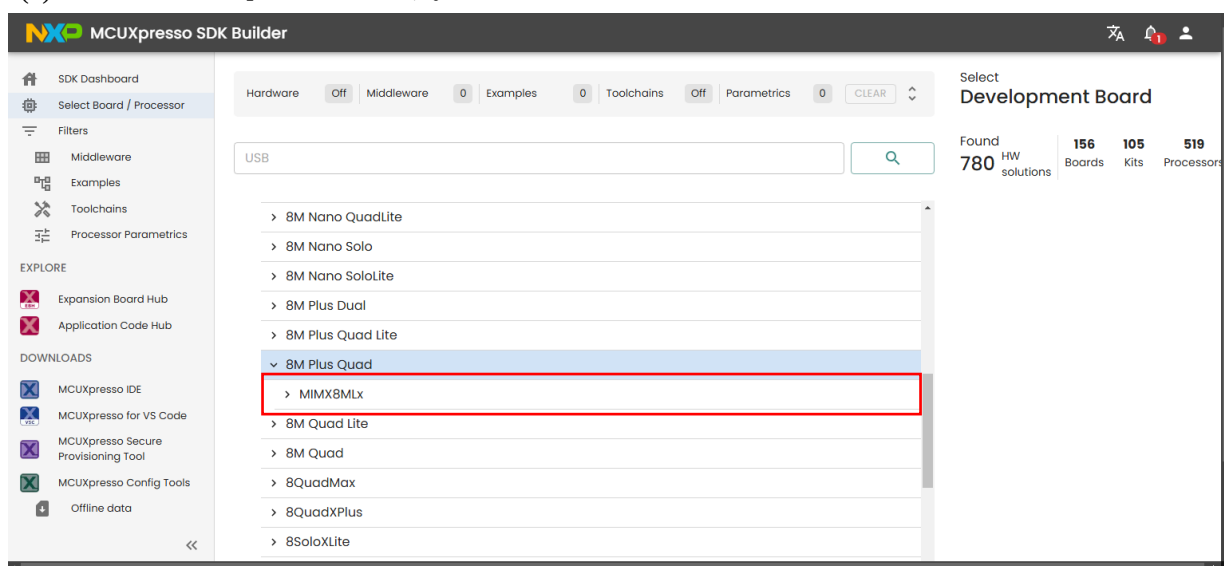


図 1-2 2 SDK カスタマイズ画面 4

(8) 「MIMX8ML8xxxKZ」を選択します。

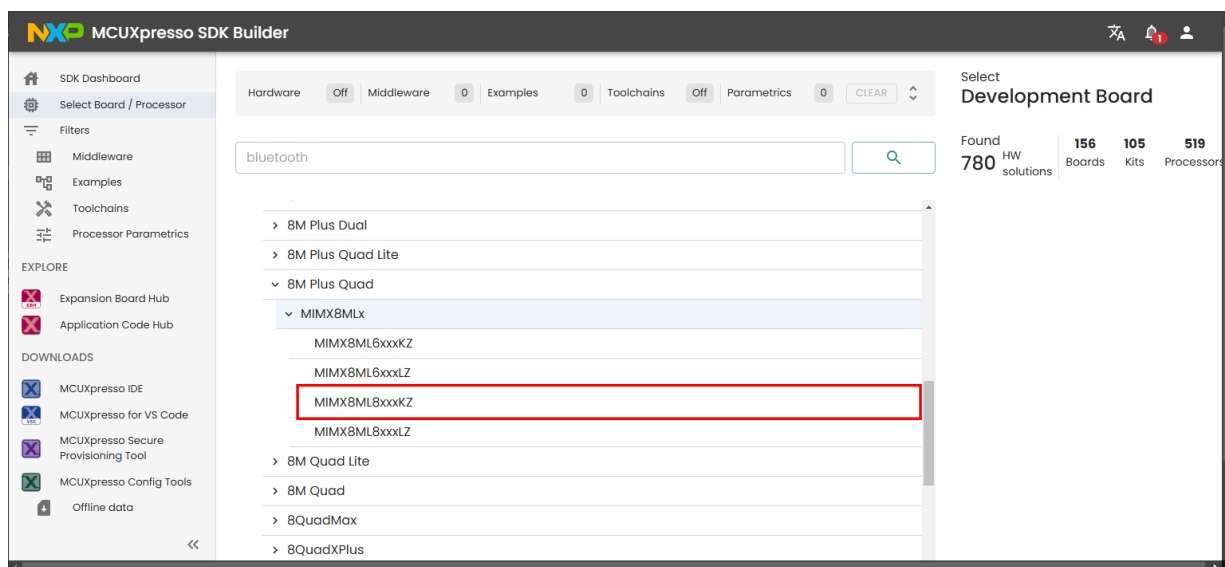


図 1-23 SDK カスタマイズ画面 5

(9) 画面右側の表示が変わることを確認してください。
右側の表示が変わらない場合は、ブラウザの画面サイズを大きくしてください。

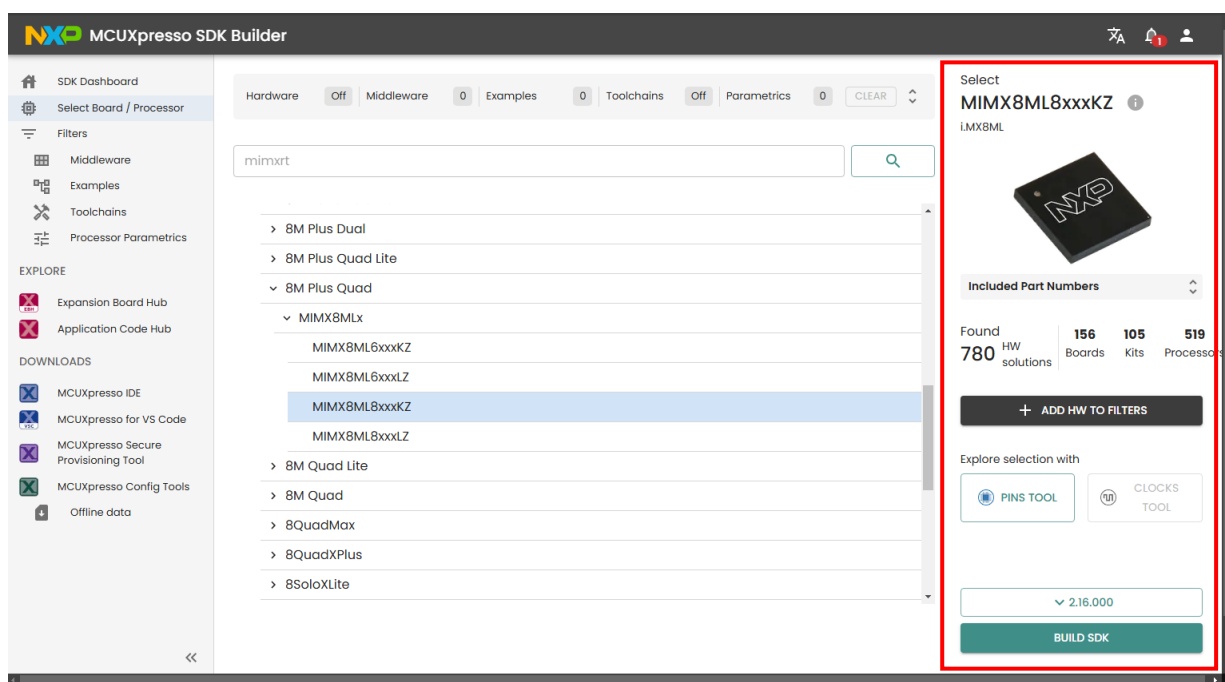


図 1-24 SDK カスタマイズ画面 6

(10) バージョンを「2.15.000」に変更します。

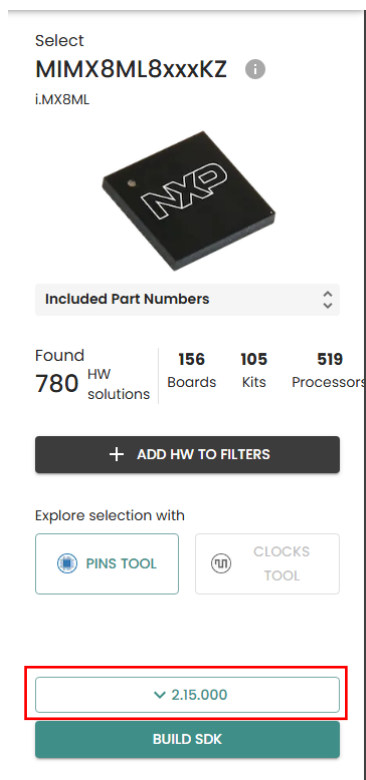


図 1-2 5 SDK のバージョン変更画面

(11) 「BUILD SDK」を押します。

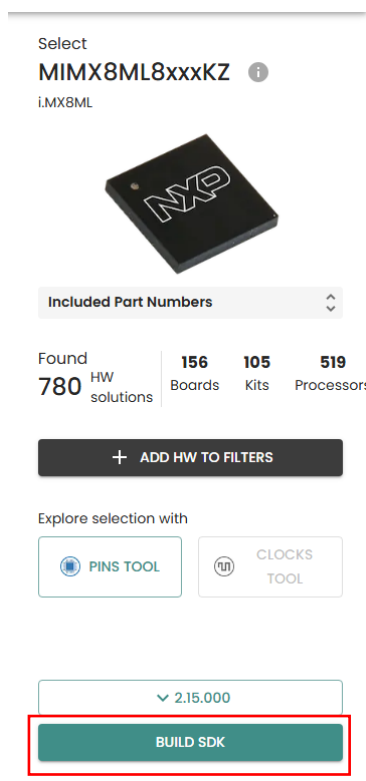


図 1-2 6 SDK のビルド画面

(12) Toolchain / IDE が「ARM GCC / MCUXpresso for VS Code」であることを確認します。

Developer Environment Settings
 Selections here (operating host system, toolchain or middleware) will impact files and examples projects included in the SDK and Generated Projects

Host OS: Windows

Toolchain / IDE: ARM GCC / MCUXpresso for VS Code

☒	Name	Description	Dependencies
☒	CMSIS DSP Library	CMSIS DSP Software Library	
☒	coreHTTP	coreHTTP	
☒	FreeMASTER	FreeMASTER communication driver for 32bit platfor...	
☒	FreeRTOS	FreeRTOS Real-time operating system for microc...	
☒	llhttp	HTTP parser llhttp	
☒	multicore	Multicore Software Development Kit This optional c...	
☒	PKCS #11	OASIS TC PKCS #11	
☒	TinyCBOR	TinyCBOR from Intel	
☒	VoiceSeeker (no AEC)	VoiceSeeker is a multi-microphone voice control a...	
☒	VoiceSpot	VoiceSpot is a highly accurate, small memory and ...	

Build SDK for MIMX8ML8xxxKZ
i.MX8M1

SDK Version: 2.15.000 (released 2024-01-15)

Additional Information

Included Part Numbers: MIMX8ML8CVNKZ
 Board(s): EVK-MIMXBMP
 Processor: MIMXBML8
 Core Type / Max Freq: Cortex-M7F / 800 MHz
 Device Memory Size: 0 KB Flash / 256 KB RAM

Documentation

Tested toolchain version

BUILD SDK

図 1-27 開発環境設定 1

(13) 「FreeRTOS」のみチェックし、「BUILD SDK」を押します。

Developer Environment Settings
 Selections here (operating host system, toolchain or middleware) will impact files and examples projects included in the SDK and Generated Projects

Host OS: Windows

Toolchain / IDE: ARM GCC / MCUXpresso for VS Code

☐	Name	Description	Dependencies
☐	CMSIS DSP Library	CMSIS DSP Software Library	
☐	coreHTTP	coreHTTP	
☐	FreeMASTER	FreeMASTER communication driver for 32bit platforms	
☒	FreeRTOS	FreeRTOS Real-time operating system for microc...	
☐	llhttp	HTTP parser llhttp	
☐	multicore	Multicore Software Development Kit This optional co...	
☐	PKCS #11	OASIS TC PKCS #11	
☐	TinyCBOR	TinyCBOR from Intel	
☐	VoiceSeeker (no AEC)	VoiceSeeker is a multi-microphone voice control aud...	
☐	VoiceSpot	VoiceSpot is a highly accurate, small memory and ML...	

Build SDK for MIMX8ML8xxxKZ
i.MX8M1

SDK Version: 2.15.000 (released 2024-01-15)

Additional Information

Included Part Numbers: MIMX8ML8CVNKZ
 Board(s): EVK-MIMXBMP
 Processor: MIMXBML8
 Core Type / Max Freq: Cortex-M7F / 800 MHz
 Device Memory Size: 0 KB Flash / 256 KB RAM

Documentation

Tested toolchain version

BUILD SDK

図 1-28 開発環境設定 2

(14) 「Download」ボタンを押します。

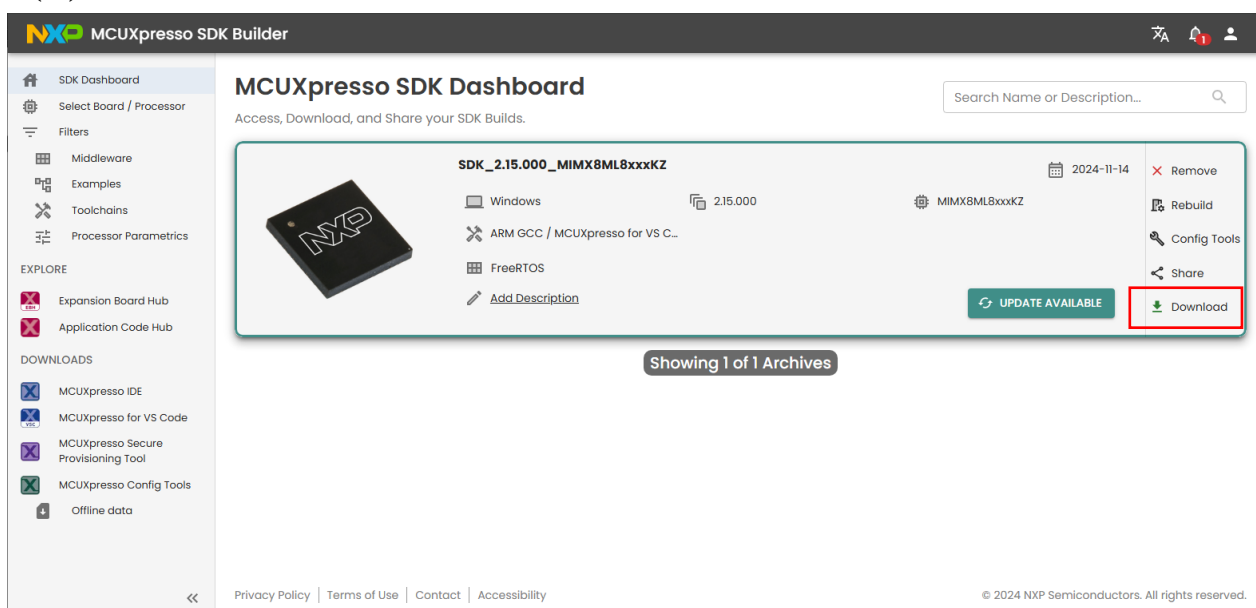


図 1-29 SDK ダウンロード

(15) 「Download SDK archive including documentation」を押します。

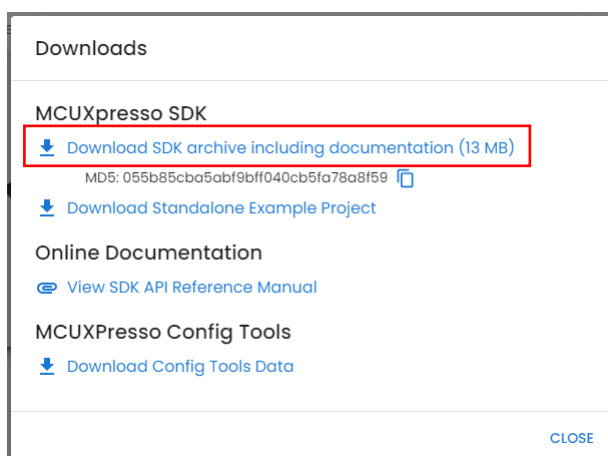


図 1-30 オプション選択画面

- (16) 「AGREE」 ボタンを押します。

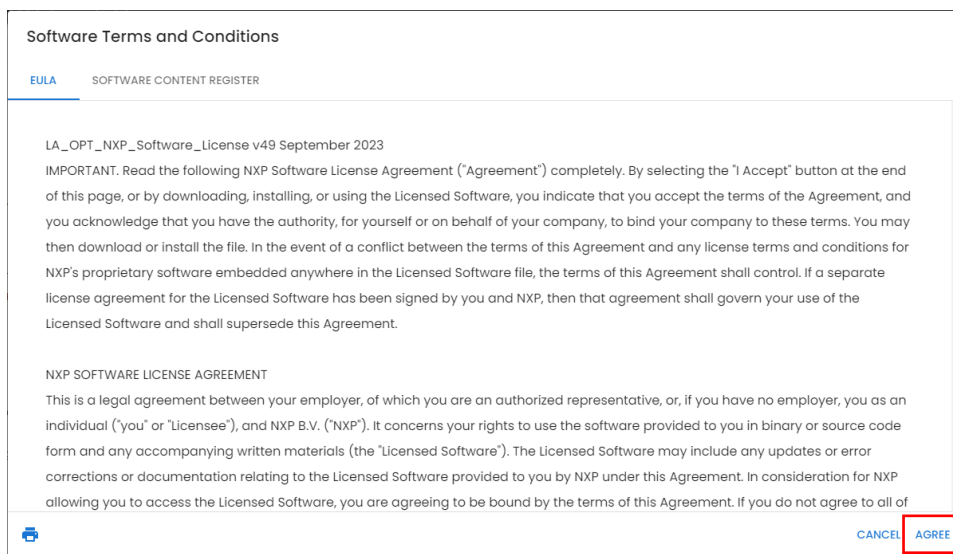


図 1-3 1 ソフトウェア利用規約

- (17) ダウンロードした「SDK_2_15_000_MIMX8ML8xxxKZ.zip」を展開します。

展開先に以下のパスを入力した後に、「展開」ボタンを押します。

「C:¥WinPass¥RobinWpf¥SDK」

※別の展開先を指定すると、「1-5 Pass/ROBIN 環境設定手順」での環境変数自動設定が行われません。

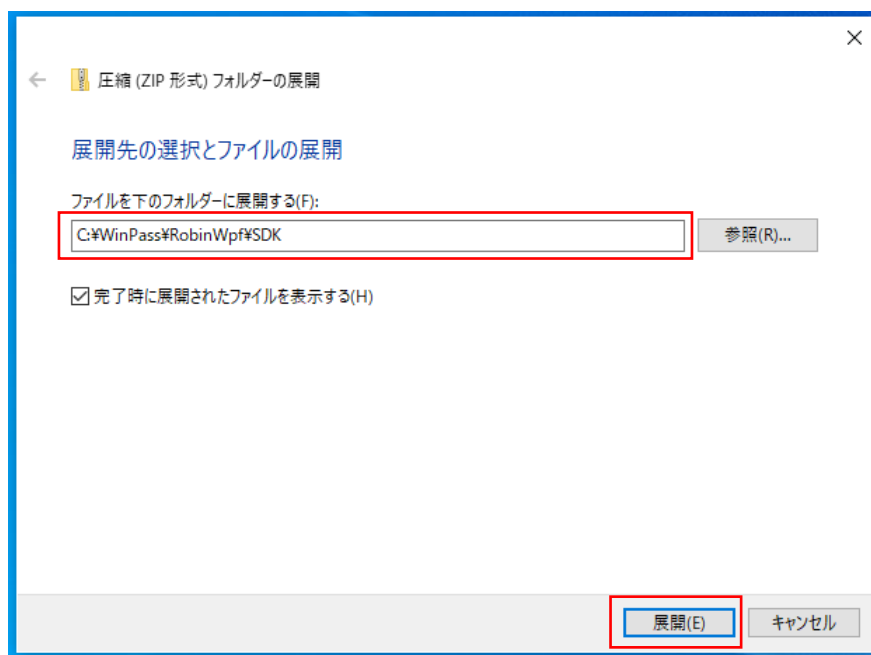


図 1-3 2 SDK の展開

1 - 4 Pass/ROBIN インストール手順

Pass/ROBIN のインストール手順を説明します。

- (1) インストールディスクを DVD ドライブに挿入します。
エクスプローラから DVD ドライブの Install フォルダ内の setup.exe を実行します。
- (2) 「インストール」をクリックします。
(.NET デスクトップ ランタイムがインストール済みの場合、この手順はスキップされます)

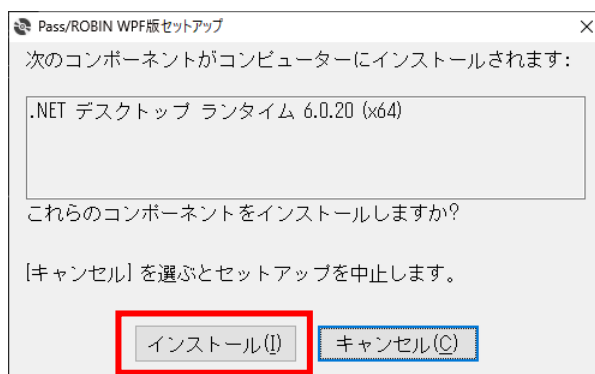


図 1-3 3 .NET デスクトップ ランタイムのインストール確認

- (3) 「このアプリがデバイスの変更を加えることを許可しますか？」の確認が 2 回あります。
いずれも「はい」をクリックします。
- (4) Pass/ROBIN のインストールを開始します。「次へ」をクリックします。



図 1-3 4 Pass/ROBIN のインストール開始

- (5) インストールフォルダとユーザを選択して、「次へ」をクリックします。

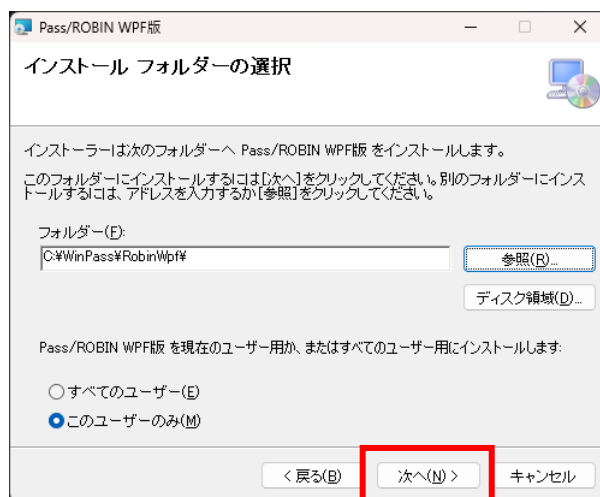


図 1-35 インストールフォルダとユーザの選択

- (6) Pass/ROBIN のインストールの確認があります。「次へ」をクリックします。

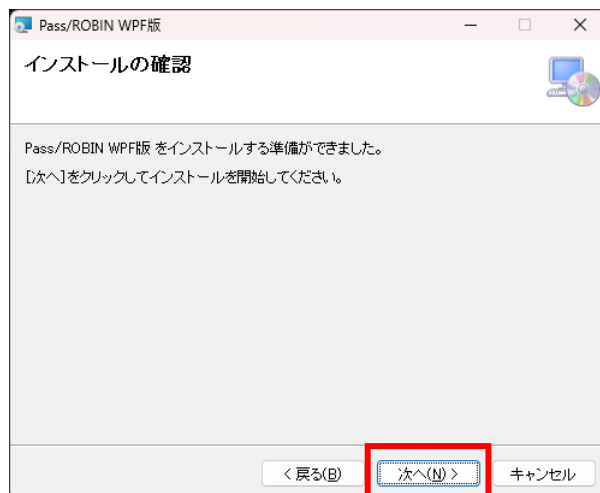


図 1-36 インストールの確認

- (7) インストールが完了すると、「閉じる」をクリックします。

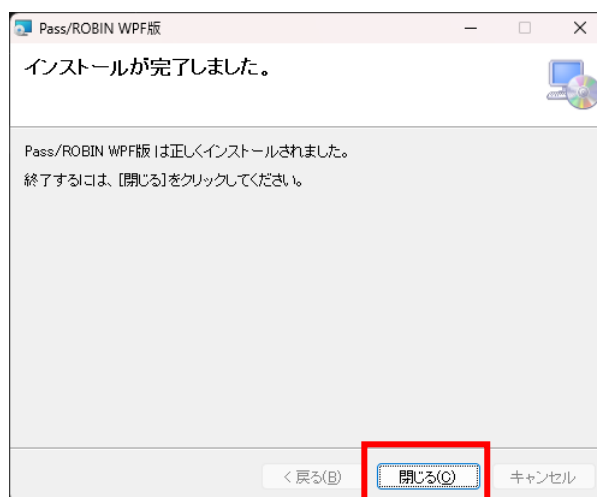


図 1-37 インストール完了

- (8) プロテクトキードライバをインストールします。「次へ」をクリックします。

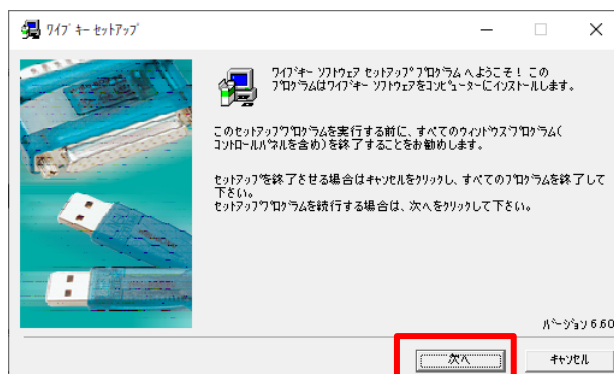


図 1-38 プロテクトキードライバのインストール

- (9) ソフトウェアライセンス条項に同意し、「次へ」をクリックします。

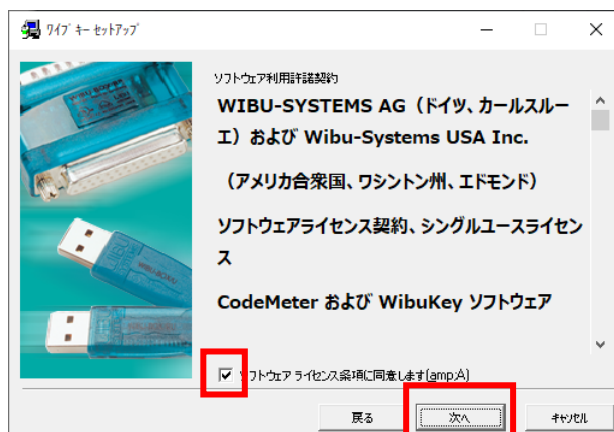


図 1-39 ソフトウェア利用許諾契約

(10) 「日本語」をチェックし、「次へ」をクリックします。



図 1-40 言語選択

(11) ネットワーク関連の機能は使用しませんので、チェックを外し「次へ」をクリックします。

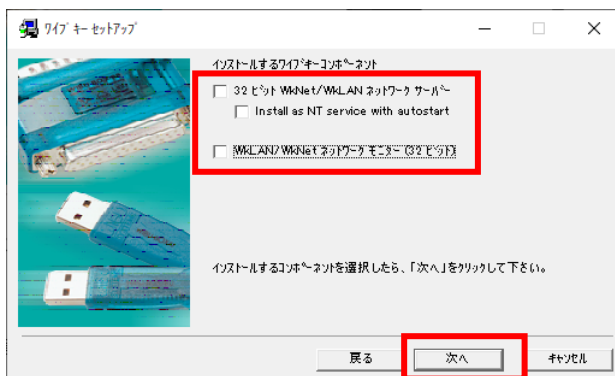


図 1-41 ネットワーク機能選択

(12) インストールを開始します。「次へ」をクリックします。

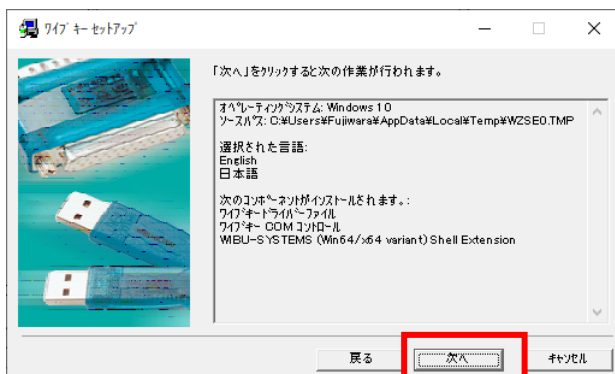


図 1-42 インストール確認

(13) インストールが完了すると、「次へ」をクリックします。

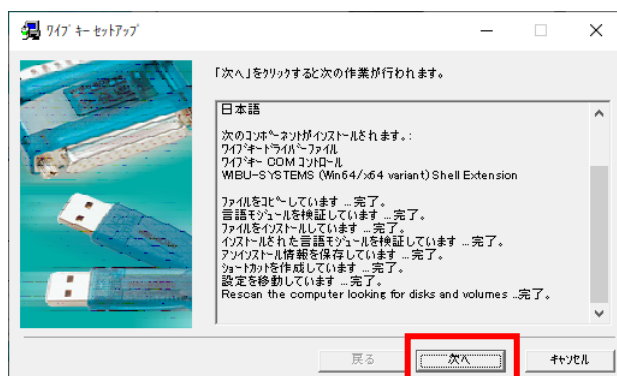


図 1-4 3 インストール完了

(14) 「はい、Readme text を参照します。」のチェックを外し、「終了」をクリックします。

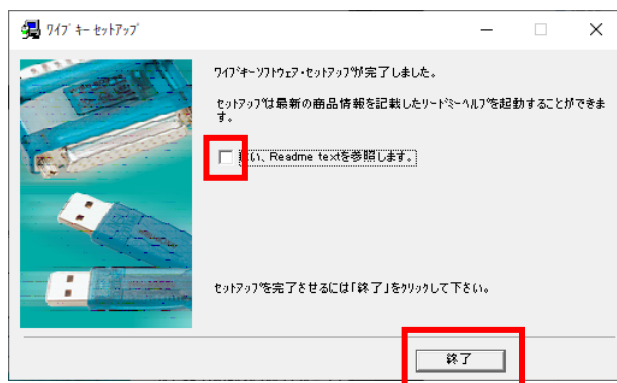


図 1-4 4 セットアップ完了

(15) インストール完了のメッセージがポップアップ表示されます。「OK」をクリックします。

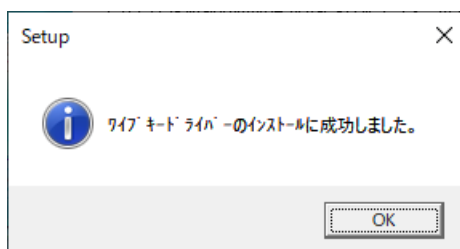


図 1-4 5 インストール完了メッセージ

1 - 5 Pass/ROBIN 環境設定手順

Pass/ROBIN を使用するための環境設定手順を説明します。

ARM 開発環境と SDK を以下にインストールした場合、自動的に環境設定が行われます。
この場合、環境設定画面は表示されません。

(環境変数の登録が完了したことを示すダイアログは表示されます。)

ARM 開発環境 : C:\Program Files (x86)\GNU Arm Embedded Toolchain\10.2021.10

SDK : C:\WinPass\RobinWpf\SDK

- (1) Pass/ROBIN のインストール完了後、環境設定画面を表示します。
環境設定を後で行う場合、キャンセルを選択して終了します。
もう一度環境設定を行う場合、環境設定実行ファイル(Setting.exe)を実行するか、
Windows のスタートメニューから「PassROBIN_WPF」→「Setting」を選択します。

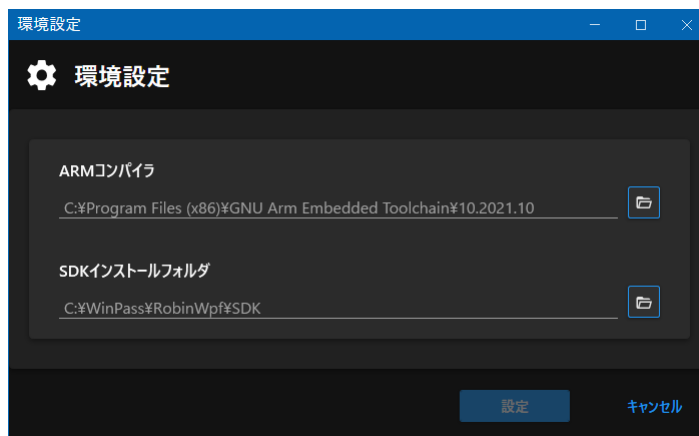


図 1 - 4 6 環境設定画面

- (2) 「ARM コンパイラ」に、ARM 開発環境のパスを設定します。
C:\Program Files (x86)\GNU Arm Embedded Toolchain\10 2021.10
- (3) 「SDK インストールフォルダ」に、SDK の展開先のパスを設定します。
C:\WinPass\RobinWpf\SDK

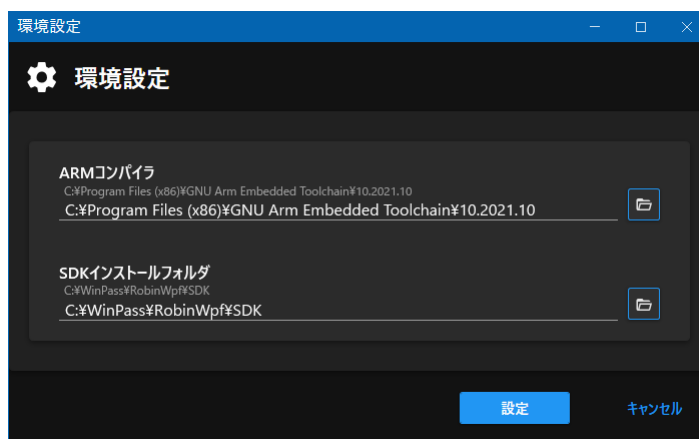


図 1 - 4 7 ARM コンパイラ、SDK インストールフォルダの設定

- (4) 「設定」ボタンをクリックすると、システムに Pass/ROBIN が使用する環境変数を登録します。このとき、環境変数の登録が完了したことを示すダイアログが表示されます。登録する環境変数の一覧は、下表のとおりです。

表 1-2 Pass/ROBIN 環境変数一覧

変数名	変数値	デフォルト設定での値
PASS_ROBIN_WPF	Pass/ROBIN インストールフォルダ	C:\WinPass\RobinWpf
ARMGCC_DIR	ARM コンパイラ	C:\Program Files (x86)\GNU Arm Embedded Toolchain\10.2021.10
ROBINSDKROOT_DIR	SDK インストールフォルダ	C:\WinPass\RobinWpf\SDK
Path	ARM コンパイラへのパス	C:\Program Files (x86)\GNU Arm Embedded Toolchain\10.2021.10\bin

- (5) PC を再起動します。
起動後、コマンドプロンプトで `set` と入力し、環境変数が登録できていることを確認します。環境変数を正しく登録できた場合、手順(6)～手順(8)は必要ありません。
- (6) Windows に administrator 権限を持つユーザでログインし、「コントロールパネル」－「システムとセキュリティ」－「システム」－「システムの詳細設定」と選択して、システムのプロパティを開きます。

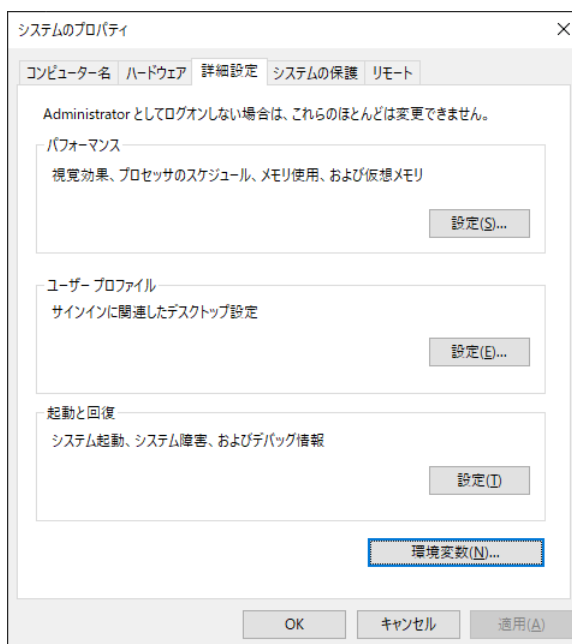


図 1-48 システムプロパティ

(7) 環境変数を選択し、環境変数エディタを開きます。

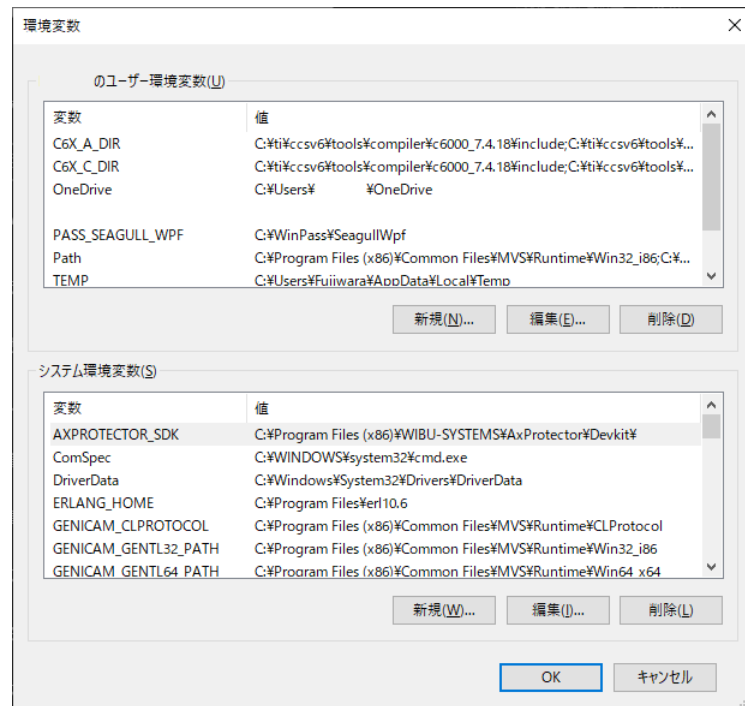


図 1-49 環境変数エディタ

(8) Pass/ROBIN で使用する環境変数を設定します。

- **PASS_ROBIN_WPF**
Pass/ROBIN インストールフォルダのパスをフルパスで指定します。
Pass/ROBIN を C:\WinPass\RobinWpf にインストールした場合、”C:\WinPass\RobinWpf”とします。
- **ARMGCC_DIR**
ARM コンパイラのパスをフルパスで指定します。
ARM 開発環境を C:\Program Files (x86)\GNU Arm Embedded Toolchain\10.2021.10 をインストールした場合、”C:\Program Files (x86)\GNU Arm Embedded Toolchain\10.2021.10”とします。
- **ROBINS DKROOT_DIR**
SDK の展開先フォルダをフルパスで指定します。
SDK を C:\WinPass\RobinWpf\SDK に展開した場合、”C:\WinPass\RobinWpf\SDK”とします。
- **PATH**
ARM コンパイラのパスをフルパスで指定します。
ARM 開発環境を C:\Program Files (x86)\GNU Arm Embedded Toolchain\10.2021.10 をインストールした場合、”C:\Program Files (x86)\GNU Arm Embedded Toolchain\10.2021.10\bin”とします。
なお、環境変数 PATH は OS を含む様々なアプリケーションが使用する変数のため、変数値が既に存在する場合は末尾に追加します。

1 - 6 Pass/ROBIN MATLAB 検索パス設定手順

Pass/ROBIN を使用するための MATLAB 検索パス設定手順を説明します。

(1) MATLAB を起動し、ホーム画面の「環境-パスの設定」でパス設定画面を開きます。
なお、MATLAB のバージョンおよびウインドウサイズによっては「環境-パスの設定」の表示位置が異なる場合があります。

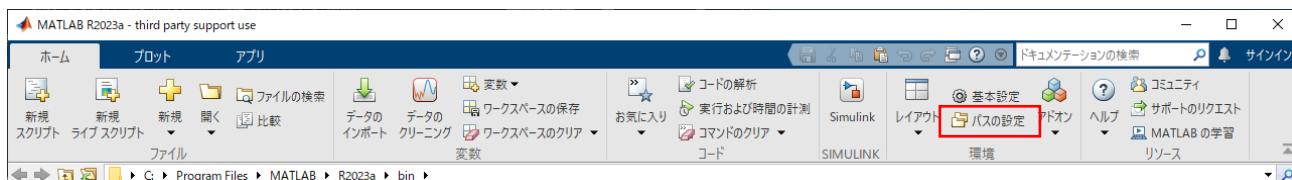


図 1-50 パス設定画面

(2) MATLAB 検索パスを Pass/ROBIN ライブラリを含むフォルダに設定します。
パス設定画面で「フォルダーを追加」を選択し、”Pass/ROBIN インストールフォルダ\matlab\rtw\c\pass”を追加します。

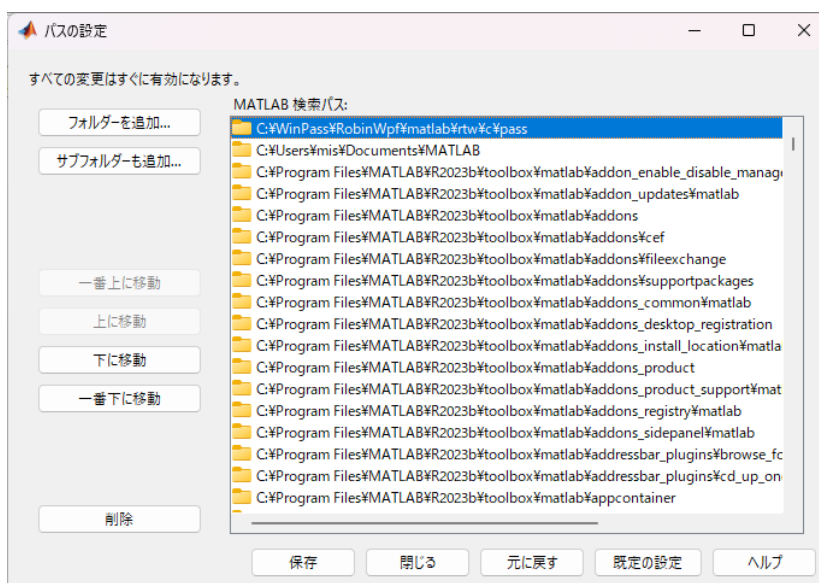


図 1-51 MATLAB 検索パス

(3) 設定の保存後、MATLAB コマンドウィンドウに passRobinWpf と入力します。
MATLAB パスが正常に登録できていれば、Simulink ライブラリと Pass/ROBIN ライブラリが呼び出されます。

2 アンインストール手順

アンインストール手順は、以下のとおりです。

- (1) MATLAB でパスの設定画面を開き、「1-6 Pass/ROBIN MATLAB 検索パス設定手順」で追加した検索パスを削除します。
- (2) MATLAB を終了します。
- (3) Windows のスタートメニューより、「コントロールパネル」－「プログラムのアンインストール」－「Pass/ROBIN WPF 版」を右クリックし、「アンインストール」を選択します。
- (4) アンインストール後、Pass/ROBIN インストールフォルダ内にはインストール後にユーザが作成したファイル（Simulink モデル、ARM プログラム、保存データ等）が削除されずに残ります。これらのファイルが不要な場合、エクスプローラ等で削除してください。

3 インストールファイル一覧

Pass/ROBIN インストーラがインストールするファイル一覧は、下表のとおりです。

表 3-1 Pass/ROBIN インストールファイル一覧

インストールフォルダ	フォルダ	内容
C:\WinPass\RobinWpf	Application	・ Pass/ROBIN の実行ファイル PassROBIN_WPF.exe ・ 環境設定実行ファイル Setting.exe ・ プロテクトキードライバインストーラ WkRuntime.exe ・ アプリケーションの実行に必要な DLL ファイル等
	lib	ARM 開発ライブラリ
	matlab	MATLAB C コード生成に必要なファイル
	robinsrc	Pass/ROBIN S-Function ソースファイル
	sample	Simulink モデル(サンプルモデル)