

スタートガイド

Panasonic A6B シリーズ

EtherCAT AC サーボドライバ

CiA402 CSP モード

86Duino Coding IDE 501

EtherCAT ライブラリ

(Version 1.0)

改訂履歴

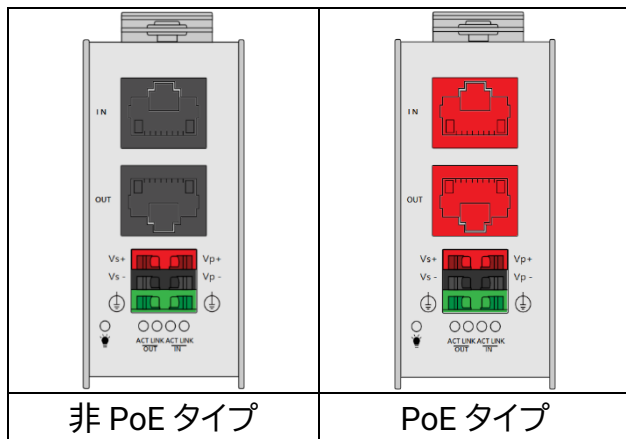
記述日	バージョン	備考
2025/1/21	Version1.0	New Release.

序文

本ガイドでは、QEC-M-01 (EtherCAT M デバイス) と Panasonic A6B Series (AC サーボドライバ)の CiA402 Profile Position(CSP)モードによる動作での使用方法を説明します。

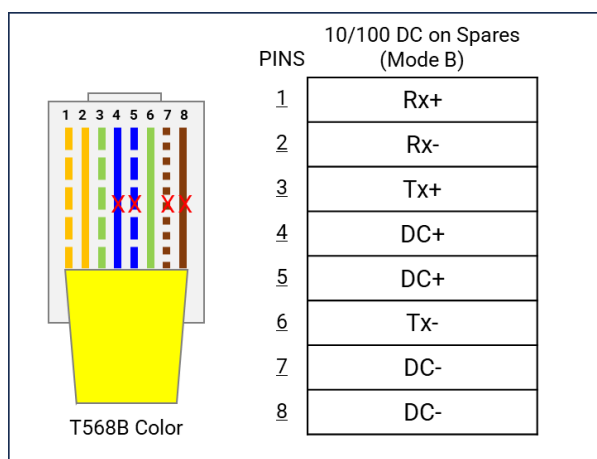
注意 QEC 機器の PoE (Power over Ethernet)について

QEC 製品のインストレーションでは、ユーザーは PoE と非 PoE を簡単に区別できます：
RJ45 ハウジングが赤色の場合は PoE タイプ、RJ45 ハウジングが黒色は非 PoE タイプです。



PoE(Power over Ethernet)は、ネットワーク経由で電力を供給する機能です。QEC には配線を減らすためオプションとして PoE 機能を用意しています。実際には PoE はシステム機器に基づいて選択されるため次の点に注意してください。

1. QEC の機能は EtherCAT P とは異なり互換性がありません。QEC の PoE 機能は PoE タイプ B に準拠しており、下記のようなピン配列になっています：



2. PoE デバイスと非 PoE デバイスを接続するときは、必ず EtherNet ケーブルのピン 4、5、7、8 を切断してください (例えば PoE 対応の QEC EtherCATM デバイスを他社の EtherCATSub デバイスに接続する場合)。
3. QEC PoE 電源は最大 24V/3A です。

1. ハードウェアの接続と配線

ここでは次のデバイスを使用します：

1. QEC-M-01 (EtherCAT M デバイス)
2. Panasonic A6B Series (AC サーボドライバ)
3. 24VDC 電源 & ヨーロッパタイプ端子ケーブル & LAN ケーブル
4. Panasonic A6B Series 用 AC 200-220V 電源



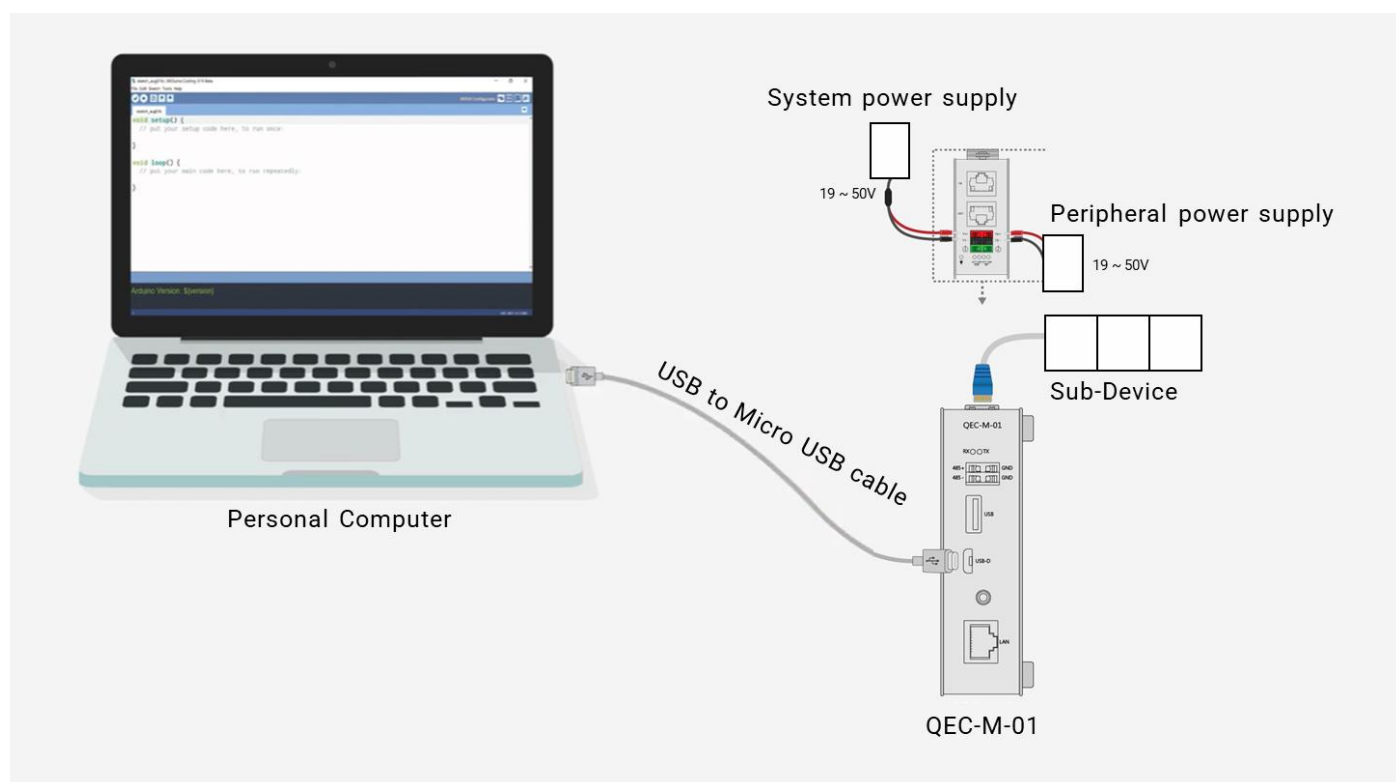
1.1 QEC-M-01

QEC EtherCAT M デバイス

1. 電源:

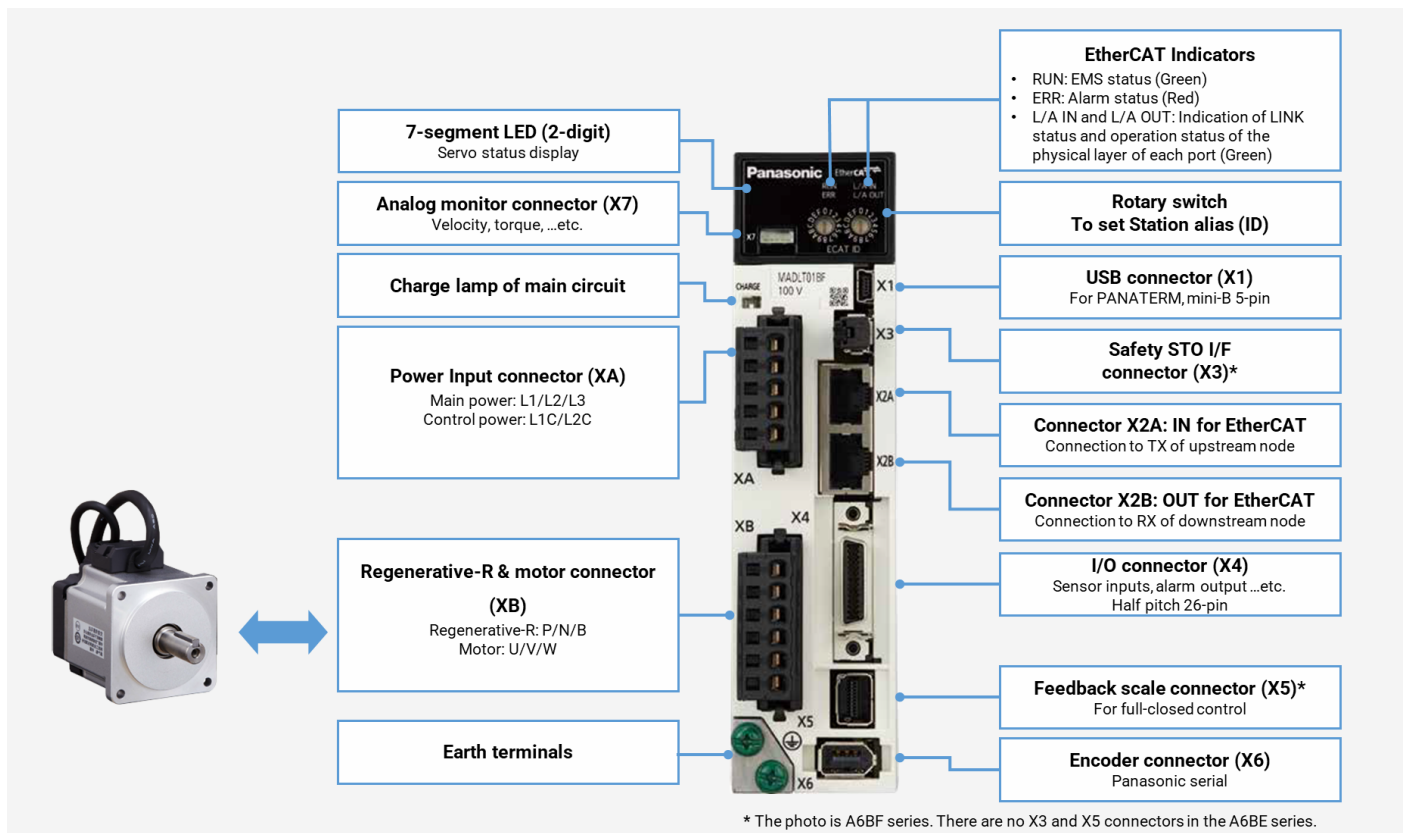
24V 電源を QEC EtherCAT M デバイスのヨーロッパタイプ端子 Vs+/Vs- および Vp+/Vp- に接続

2. EtherCAT 通信: EtherCAT 出力ポート (上側) から EtherCAT Sub デバイスの EtherCAT 入力ポートに RJ45 ケーブルで接続



1.2 Panasonic A6B シリーズ

Panasonic A6B シリーズは、高性能に統合された EtherCAT 対応 AC サーボドライバです。この図は、ドライバの配線ガイド例を示します。



1. EtherCAT 接続

- ネットワーク通信の EtherCAT 入力ポート(X2A)と EtherCAT 出力ポート(X2B)
- デバイス識別用のステーション ID ロータリースイッチ

2. 信号接続

- 入出力信号インターフェース(X4): デジタル入出力用
- エンコーダコネクタ(X6): モータ位置の追跡用
- 回生抵抗器およびモータコネクタ(XB): モータの電源および制御接続を提供します。

3. 電源供給

- 主電源入力(XA): モーターに電力を供給します
- 制御電源入力(XA1/XA2): ドライバの制御回路に電力を供給します

4. 安全性とステータス

- 安全 STO インターフェース(X3): 緊急停止のための安全トルクオフ(STO)機能をサポートします
- LED インジケータ: ドライバの状態、EtherCAT 通信の状態、およびエラー情報を表示します

5. その他の機能

- ロータリースイッチ: ステーションエイリアス(ID)の設定に使用します。
- USB インターフェース(x1): 設定および監視に使用します

配線図の概要

以下は、Panasonic A6B EtherCAT サーボドライバの一般的な配線例です。

1. EtherCAT M デバイス接続:

- QEC-M-01 の EtherCAT 出力ポートをサーボドライバの X2A(EtherCAT IN)ポートに接続します
- 追加のサブデバイスが必要な場合は、X2B(EtherCAT OUT)ポートを次のサブデバイスの EtherCAT 入力に接続します

2. モータ接続:

- モータの U、V、W 線を XB コネクタに接続します
- 回生抵抗を P、N、B 端子に接続します

3. 電源配線:

- 主電源を XA コネクタ(L1、L2、L3)に接続します
- 制御電源を XA1 および XA2 端子に接続します

4. 信号配線

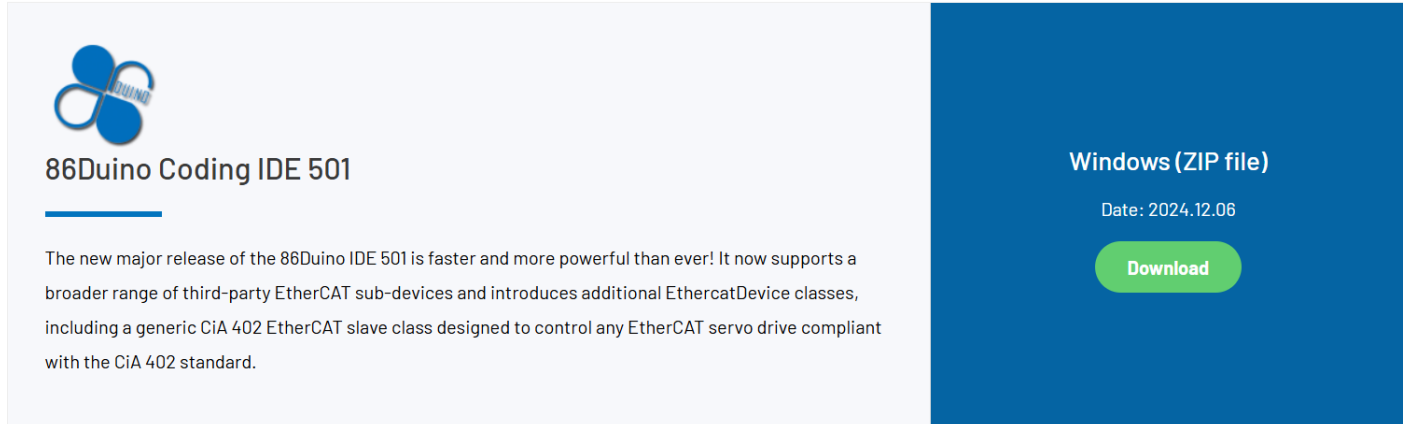
- センサやアラームなどの入出力信号を X4 コネクタに接続します
- エンコーダ信号線を X6 コネクタに接続します

5. 追加設定:

- ロータリースイッチを使用してステーションエイリアス(ID)を設定します
- X1 USB インターフェースを介してサーボドライバを設定します

2. ソフトウェア/開発環境

<https://www.qec.tw/software/>から 86duino IDE をダウンロードしてください。



86Duino Coding IDE 501

The new major release of the 86Duino IDE 501 is faster and more powerful than ever! It now supports a broader range of third-party EtherCAT sub-devices and introduces additional EthercatDevice classes, including a generic CiA 402 EtherCAT slave class designed to control any EtherCAT servo drive compliant with the CiA 402 standard.

Windows (ZIP file)
Date: 2024.12.06
[Download](#)

ダウンロード後、ダウンロードした zip ファイルを解凍してください。

追加のソフトウェアのインストールは必要ありません。86duino.exe をダブル・クリックして IDE を起動します。



***注:** Windows が警告を表示させた場合は、[詳細]を 1 回クリックし、[実行を続行]ボタンを 1 回クリックします。

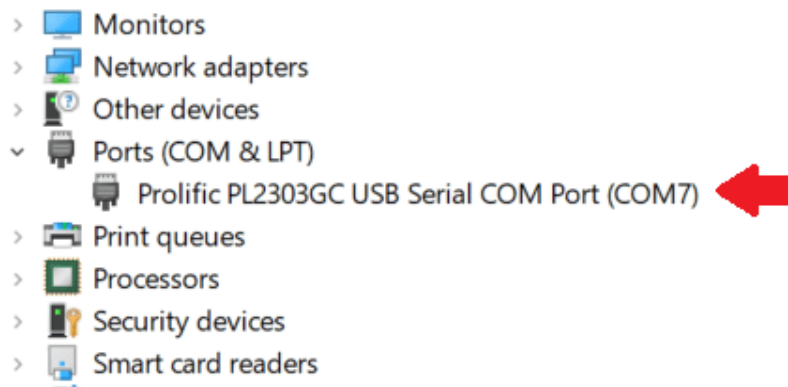
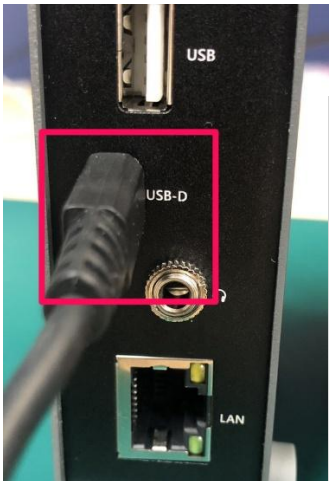
86Duino コーディング IDE 500+ は下図です。



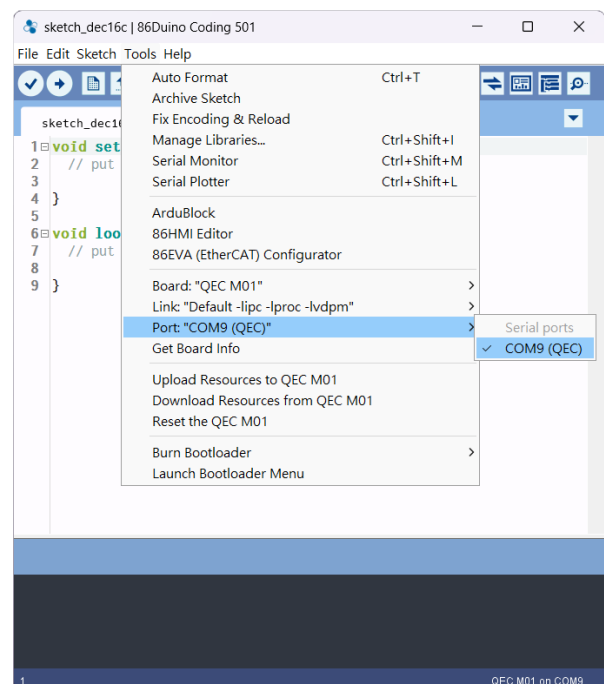
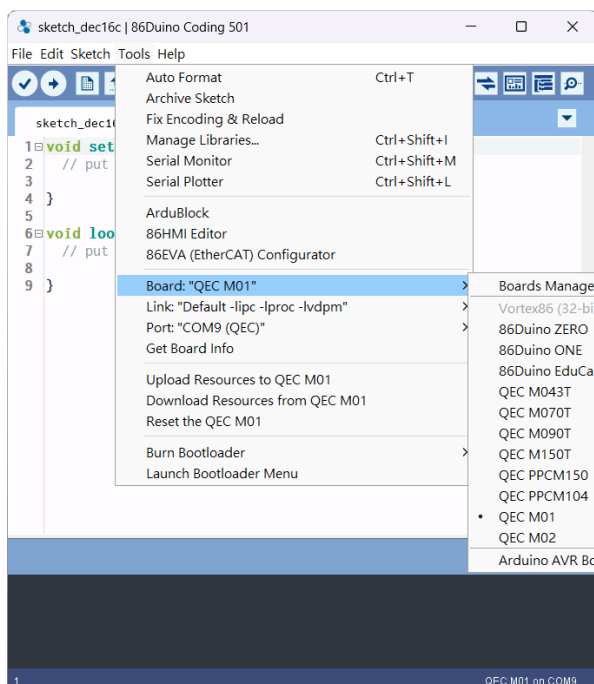
3. PC に接続して環境をセットアップする

以下の手順に従って開発環境をセットアップします:

1. Micro USB - USB ケーブルで QEC-M-01 を 86Duino IDE がインストールされた PC に接続します
2. QEC の電源を投入します。
3. PC で「デバイス・マネージャー」(Win+X キーを押した後のメニューで選択) -> 「ポート(COM および LPT)」を開き、ポートの内容を確認します。「Prolific PL2303GC USB Serial COM Port (COMx)」が検出されていることがわかります。検出されていない場合は、必要なドライバをインストールする必要があります(Windows PL2303 ドライバの場合は、[ここ](#)からダウンロードできます)



4. 86Duino IDE を開きます。
5. ボード(QEC M デバイス)の選択: IDE のメニューで、[Tools] > [Board] > [QEC-M-01] (または使用する QEC-M M デバイス型名) を選択します。
6. ポートの選択: IDE のメニューで、[Tools] > [Port] を選択し、デバイス・マネージャーで確認した QEC-MM デバイスに接続する USB ポートを選択します (この場合、COM9 (QEC))。



4. コードの記述

86Duino IDE の EtherCAT ライブラリを介して、Panasonic A6B (EtherCAT AC サーボドライバ)をコンフィギュレーションできます。

Arduino 開発環境(コーディング)には 2 つの主要な部分があります:初期化のための `setup()` とメイン・プログラムの `loop()`です。EtherCAT ネットワークを操作する前に、一度コンフィグレーションする必要があります。そのプロセスにより EtherCAT デバイスは Pre-OP から OP モードになります。

以下のプログラムは Panasonic A6B ドライバを CiA402 Cyclic-Synchronous Position モードに設定します。

- EtherCAT サイクルタイム: 1ms Cycle Time:
- EtherCAT モード: ECAT_SYNC
- EtherCAT Distributed Clock (DC) モード: Enabled.

`EthercatMaster` オブジェクト("master")は QEC-M-01 を意味し、`EthercatDevice_CiA402` オブジェクト("motor")は Panasonic A6B ドライバを意味します。

A. Set up()関数内の説明:

`setup()` 関数では、通信を初期化し、モーターを CiA402 CSP モードに設定します。下記の手順に従ってください。

1. Serial 通信の初期化
 - Serial 通信をボーレート 115200 でスタート
2. EtherCAT M デバイスのスタート
 - `begin()` 関数を使用して EtherCAT M デバイスをスタート、EtherCAT ステートマシンを PRE-OPERATIONAL 状態に切り替える
3. Cyclic Synchronous Position (CSP) モードの設定
 - `setCiA402Mode(CIA402_CSP_MODE)`を使用してモータを CSP モードに設定する
4. Distributed Clock (DC) モードの設定
 - `setDc()`関数を使用して A6 サーボドライバを DC モードに設定し、サイクルタイムを EtherCAT 通信と同期させる。
5. Cyclic Callback 関数の登録
 - `attachCyclicCallback()` 関数で"myCallback"を登録する。
6. EtherCAT M デバイスの開始
 - `start()` 関数を使用して、EtherCAT ステートマシンを OPERATIONAL 状態に切り替え、サイクル時間を 1ms に設定し、モードを ECAT_SYNC に設定します。
7. 現在位置の設定
 - 安全の為 enable の開始時に `setTargetPosition()` と `getPositionActualValue()` 関数を使い最初のターゲット位置を現在位置に設定します。

8. モータを Enable にする

- `enable()` 関数を使いモータを有効にして CIA402_OPERATION_ENABLED 状態に移行します。

B. Callback 関数内の説明:

`setTargetPosition()` 関数を使用して、ターゲット位置を現在の位置に設定し、1000 を加算します。

C. Loop 関数内の説明:

`loop()` 関数は、モーターの現在位置をシリアルモニターに表示します。

プログラム例:

```
#include "Ethercat.h"

EthercatMaster master;
EthercatDevice_CiA402 motor;

int32_t position = 0;

void myCallback() {
  if (motor.getCiA402State() != CIA402_OPERATION_ENABLED)
    return;
  motor.setTargetPosition(position += 1000);
}

void setup() {
  Serial.begin(115200);



  Serial.print("Begin: "); Serial.println(master.begin());
  Serial.print("Slave: "); Serial.println(motor.attach(0, master));
  motor.setCiA402Mode(CIA402_CSP_MODE);
  motor.setDc(1000000);

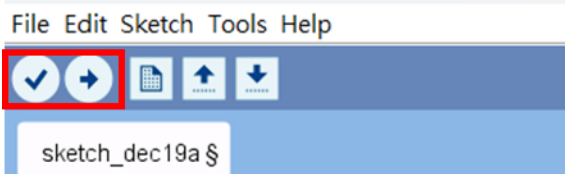
  master.attachCyclicCallback(myCallback);
  Serial.print("Start: "); Serial.println(master.start(1000000, ECAT_SYNC));

  motor.setTargetPosition(position = motor.getPositionActualValue());
  Serial.print("Enable: "); Serial.println(motor.enable());
}

void loop() {
```

```
Serial.print("Pos: "); Serial.println(motor.getPositionActualValue());  
// ...  
}
```

***注** コードを記述したならば、ツールバーの  をクリックしてコンパイルします。コンパイルが完了しエラーがないことを確認したら  をクリックしてプログラムをアップロードします。アップロードが完了するとプログラムが実行されます。



トラブルシューティング

QEC-M-01 cannot successfully upload code

コードのアップロードに成功しない場合は、86EVA を開いて QEC EtherCATM デバイスの環境に異常がないか確認してください。下図のようにブートローダ、EtherCAT ファームウェア、EtherCAT ツールを含む QEC EtherCATM デバイスの環境を更新してください。



アップデートの進め方を説明します:

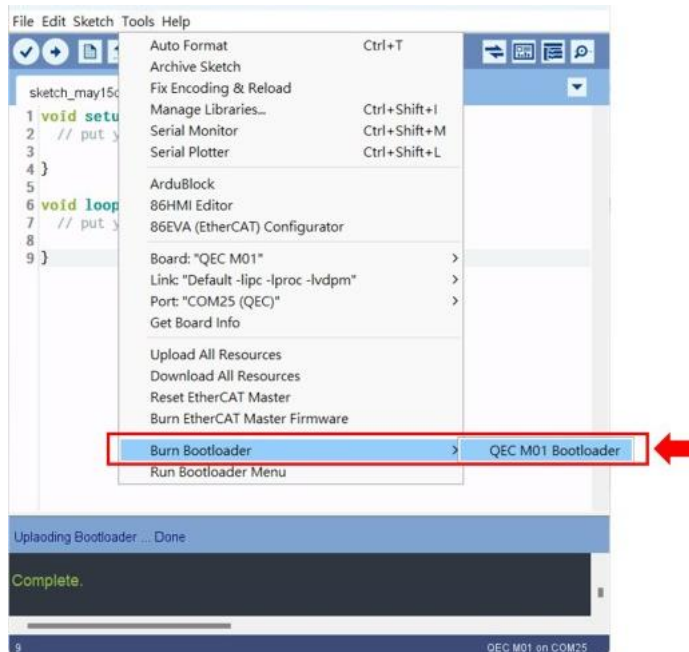
ステップ 1: QEC-M のセットアップ

1. 86Duino IDE 500 (または最新バージョン) をダウンロードしてインストールします: [Software](#) からダウンロードできます。
2. QEC-M を PC に接続: USB ケーブルを使用して QEC-M を PC に接続します。
3. 86Duino IDE を開く: インストールが完了したら、86Duino IDE ソフトウェアを開きます。
4. ボードの選択: IDE メニューから、**[Tools] > [Board] > [QEC-M-01]** (または使用中の QEC-M の型名) を選択します。
5. ポートの選択: IDE メニューから **[Tools] > [Port]** を選択し、QEC-M が接続されている USB ポートを選択します。

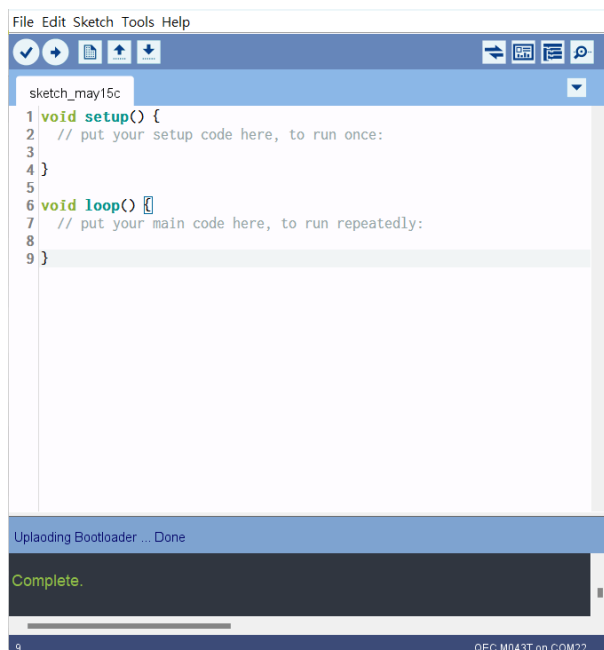
ステップ 2: 「Burn Bootloader」ボタンをクリック

QEC-M 製品に接続後、「Tools」>「Burn Bootloader」に移動します。現在選択されている QEC-M 名が表示されます。その上をクリックすると更新プロセスが開始されます。これには約 5~20 分かかります。

QEC-M-01:



ステップ 3: アップデートを完了



上記の手順を完了すると、QEC-M は最新の開発環境バージョンに正常に更新されます。

Warranty

This product is warranted to be in good working order for a period of one year from the date of purchase. Should this product fail to be in good working order at any time during this period, we will, at our option, replace or repair it at no additional charge except as set forth in the following terms. This warranty does not apply to products damaged by misuse, modifications, accident or disaster. Vendor assumes no liability for any damages, lost profits, lost savings or any other incidental or consequential damage resulting from the use, misuse of, originality to use this product. Vendor will not be liable for any claim made by any other related party. Return authorization must be obtained from the vendor before returned merchandise will be accepted. Authorization can be obtained by calling or faxing the vendor and requesting a Return Merchandise Authorization (RMA) number. Returned goods should always be accompanied by a clear problem description.

本書に記載されているブランド名および製品名は、各社の所有物および登録商標です。
本書に記載されている名称はすべて、識別目的のみに使用されます。

All Trademarks appearing in this manuscript are registered trademark of their respective owners. All Specifications are subject to change without notice.

©ICOP Technology Inc. 2025

日本語版資料は、英語版を翻訳したもので、内容に相違が生じる場合には原文を優先します。