



micro:bitで μT-Kernel 3.0を使う

2024年5月21日
パーソナルメディア株式会社

Agenda

- micro:bitとは
- micro:bitで μ T-Kernel 3.0を動かす
- micro:bit用 μ T-Kernel 3.0の開発環境
- Eclipseを使ったプログラム開発の実際
- 周辺デバイスを使う
- Q and A

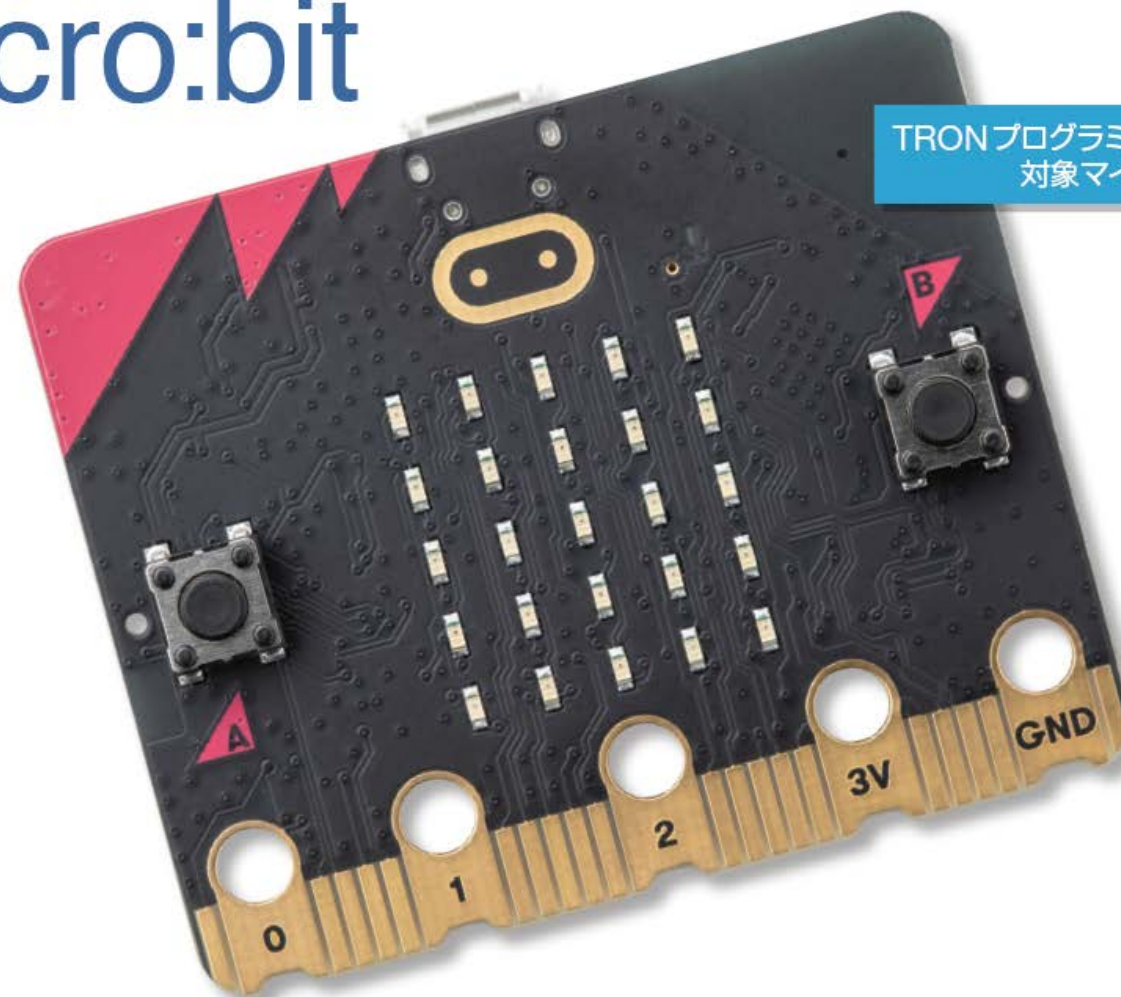


micro:bitとは

micro:bitとは

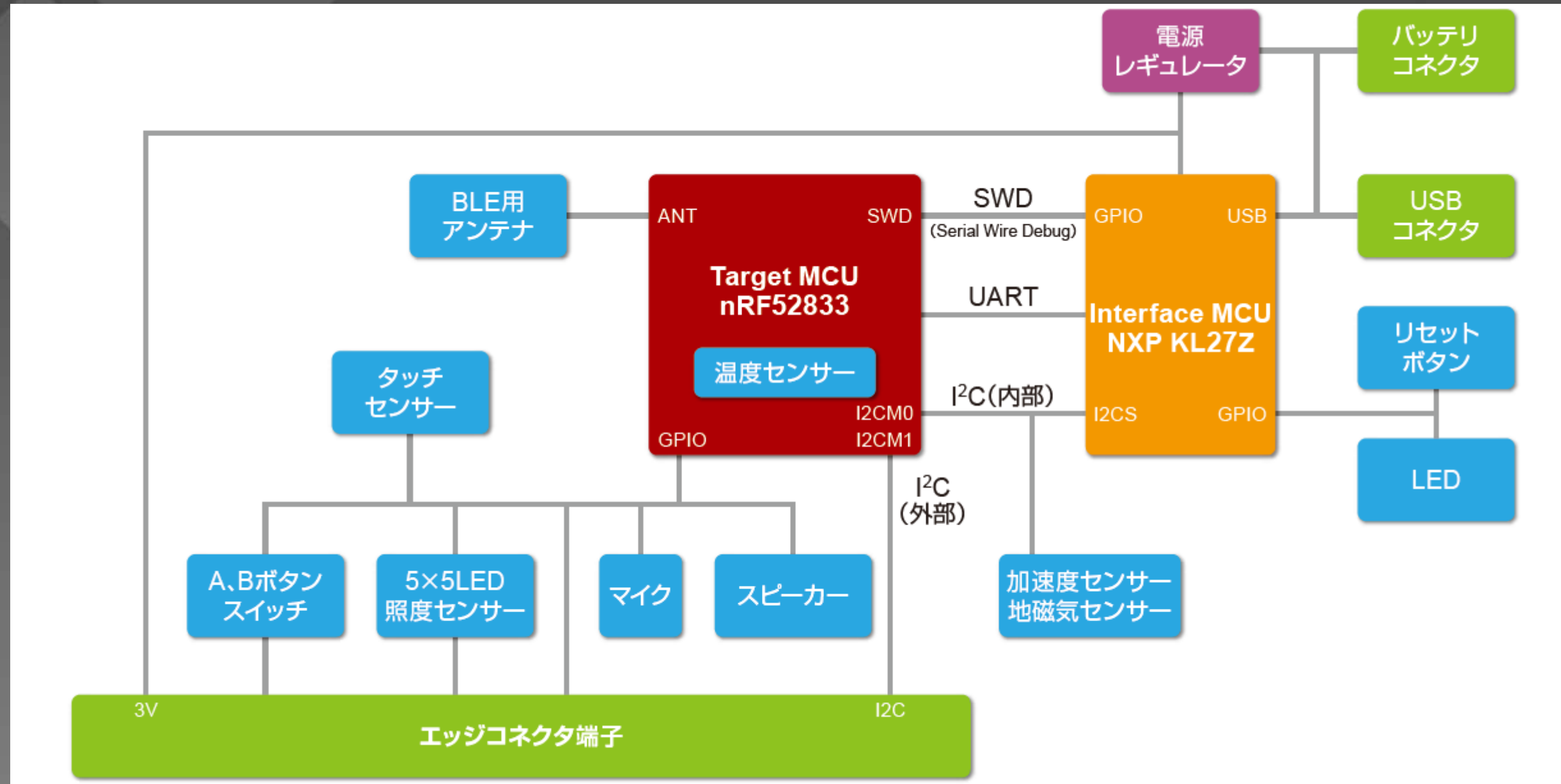
- 英国放送協会BBCが開発した超小型の教育用ボードコンピュータ
- CPUはNordicのnRF52833（Arm Cortex-M4コア）

micro:bit



micro:bitとは

■ 豊富な周辺デバイスを搭載



micro:bitとは

- 標準的にはビジュアル言語MakeCodeを使ってソフト開発





micro:bitで μT-Kernel 3.0を動かす

micro:bitでμT-Kernel 3.0を動かす

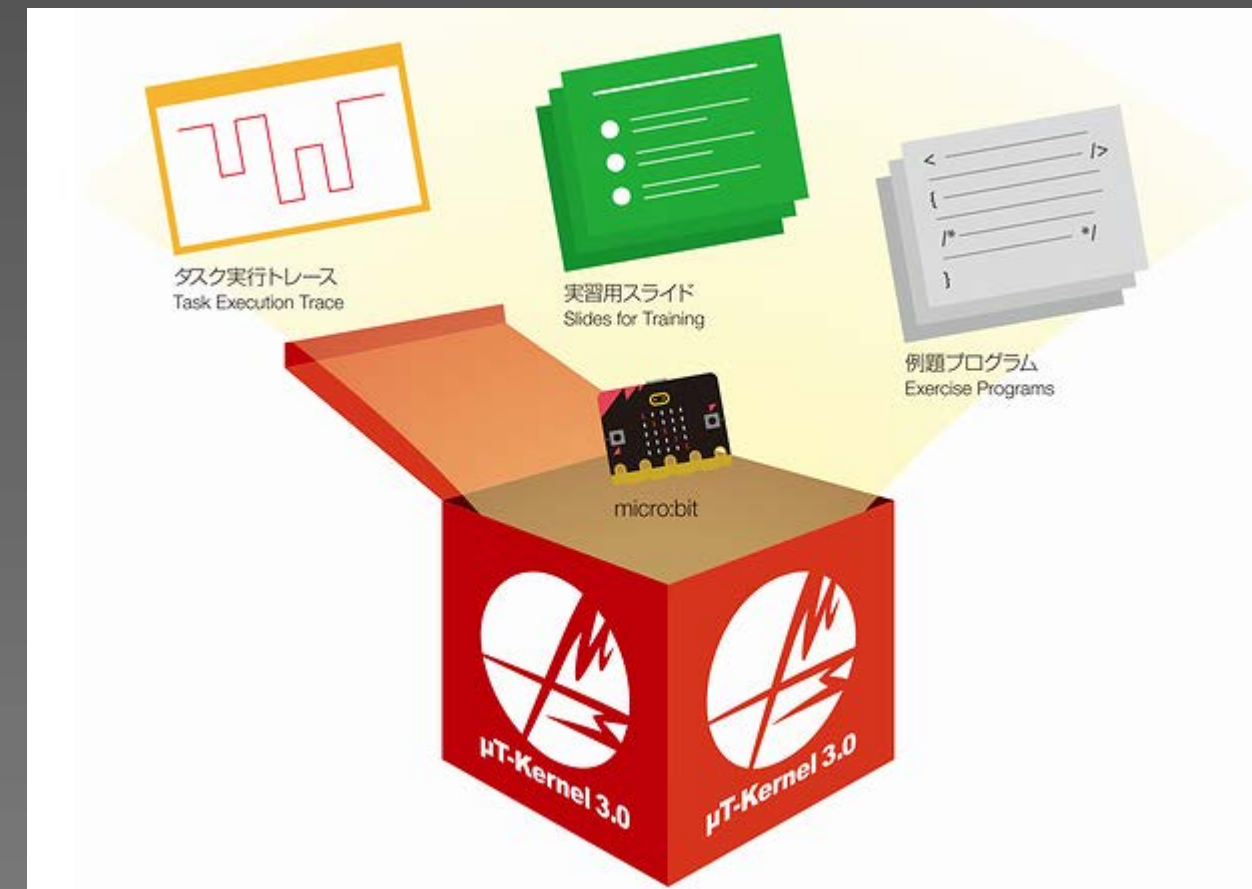
- 方法1: 「IoTエッジノード実践キット/micro:bit」を利用
- 方法2: TRONWARE記事「micro:bitでμT-Kernel 3.0を動かそう」
- 使用するμT-Kernel 3.0や開発環境、開発方法はどちらも同じ
 - トロンフォーラムのμT-Kernel 3.0をmicro:bitに移植したものを利用
 - 開発環境はEclipseを利用

micro:bitで μ T-Kernel 3.0を動かす

- 方法1: 「IoTエッジノード実践キット/micro:bit」を利用
http://www.t-engine4u.com/products/ioten_prackit.html



The screenshot shows the website www.t-engine4u.com with a navigation menu. The main content area features the title "micro:bitを使ってリアルタイムOSの動作を学ぶ学習キット" (Learning kit for learning the operation of real-time OS using micro:bit) and "IoTエッジノード実践キット micro:bit". Below the title is an image of the IoT Edge Node Practical Kit/micro:bit board. Two callout boxes are present: one on the left asking "タスクのスケジューリングとは?" (What is task scheduling?) and one on the right asking "セマフォによる排他制御とは?" (What is mutual exclusion control using semaphores?).

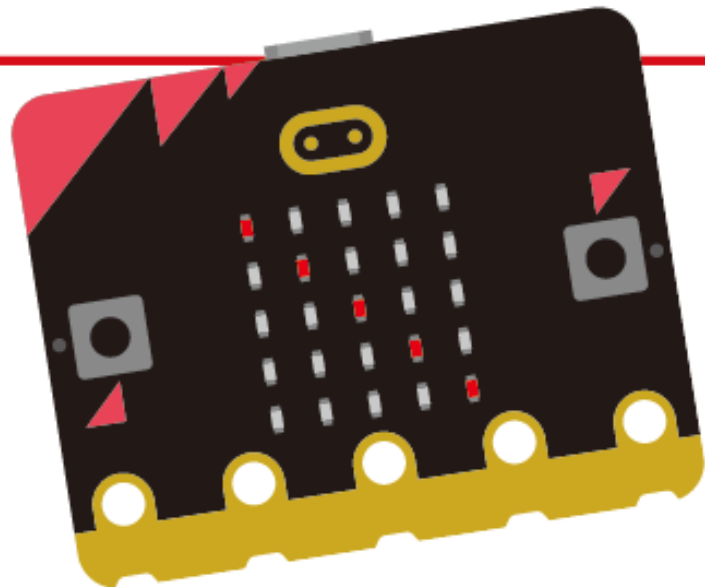


micro:bitでμT-Kernel 3.0を動かす

■ 方法2: TRONWARE記事「micro:bitでμT-Kernel 3.0を動かそう」

連載

 **micro:bit**でμT-Kernel 3.0を動かそう



 **micro:bit**で
μT-Kernel 3.0を動かそう

[第8回] ドレミファ音階の再生とティック時間

トロンフォーラム T3 WG

本連載では、小学生向けのプログラミング教育などに使われているBBC micro:bit (以下「micro:bit」)の上で動くようになったμT-Kernel 3.0をご紹介している。連載第8回の本号では、micro:bitから音を出してみよう。μT-Kernel 3.0の持つ時間関連の機能をいろいろ試しながら、ドレミファの音階を再生するところまで進めていく。

micro:bitでμT-Kernel 3.0を動かす

- 方法2: TRONWARE記事「micro:bitでμT-Kernel 3.0を動かそう」
 - TRONWARE VOL.200で連載記事の第1回-第5回を閲覧できる
 - https://www.personal-media.co.jp/book/tw/tw_index/366.html

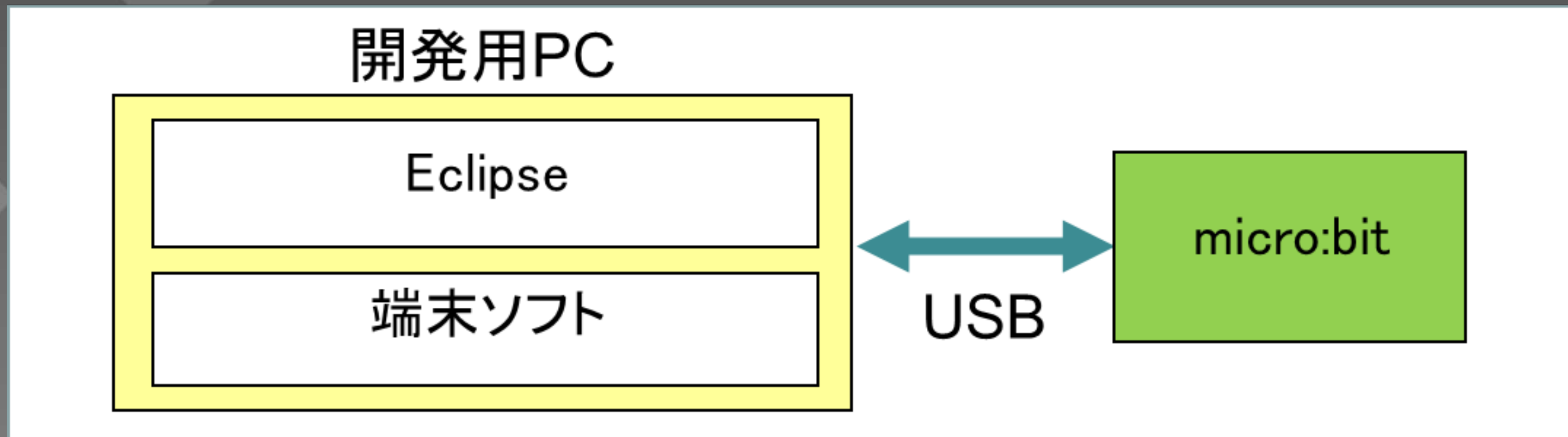




micro:bit用 μT-Kernel 3.0の開発環境

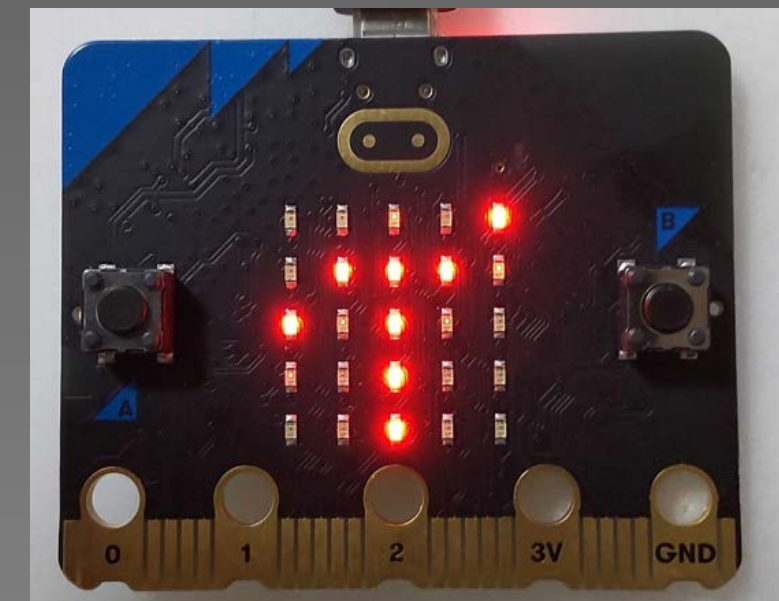
micro:bit用μT-Kernel 3.0の開発環境

- 統合開発環境Eclipseを使ったクロス開発
 - プログラムの作成、コンパイル、ビルド、実行、デバッグが可能
 - USBシリアル経由で開発用ホストの端末コンソールとの文字入出力が可能



micro:bit用 μ T-Kernel 3.0の開発環境

- アプリ開発には μ T-Kernel 3.0の開発言語であるC言語を利用
 - MakeCodeは使えない
- 周辺デバイスのハードウェア制御は自分自身で行う
 - 自分でドライバを開発するか、アプリから周辺デバイスのハードウェアを直接制御する
 - 以下のデバイスは、TRONWARE連載記事を参考にプログラミング可能
 - ✓ LEDマトリックス、ボタンスイッチ、
 - ✓ スピーカー、PWM、A/D変換、
 - ✓ I2C、加速度センサー





Eclipseを使った プログラム開発の実際

Eclipseを使ったプログラム開発の実際

- [Step-1] μ T-Kernel 3.0の開発用ツールの準備
 - (1) Eclipseのインストール
 - (2) GNU Arm Embedded Toolchainのインストール
 - (3) xPack Windows Build Toolsのインストール
 - (4) PythonとpyOCDのインストール
- [Step-2] μ T-Kernel 3.0のインポートとコンパイル
 - (5) micro:bit用 μ T-Kernel 3.0のソースコード入手と展開
 - (6) Eclipseに μ T-Kernel 3.0のプロジェクトを作成
 - (7) Eclipseの操作によるコンパイルとビルド

Eclipseを使ったプログラム開発の実際

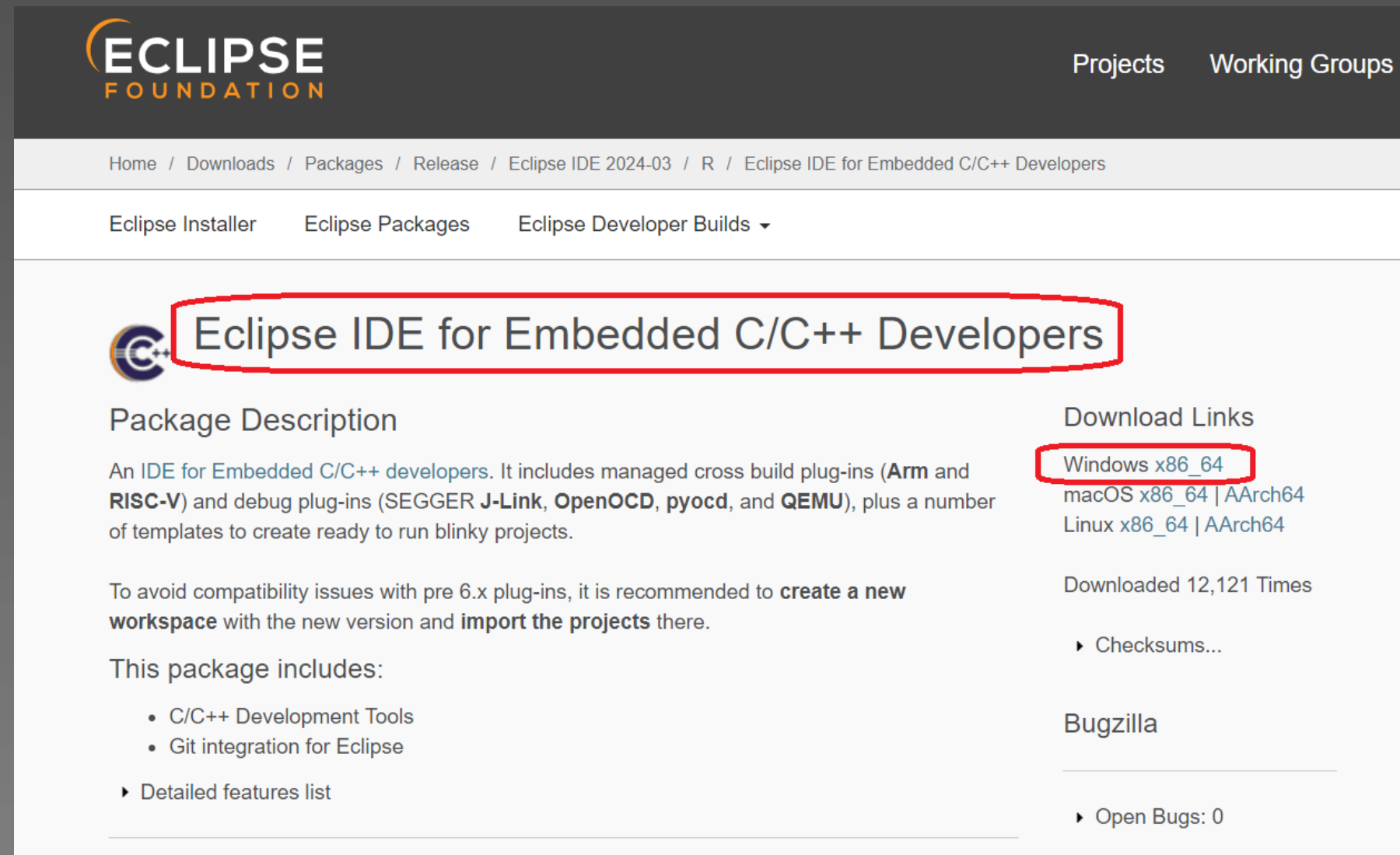
■ 使用する開発用ツールとソースコード

名称	説明	URL
Eclipse IDE for Embedded C/C++ Developers	GUI画面からソースコードの作成や編集、コンパイル、デバッグなどを行う統合開発環境	https://www.eclipse.org/downloads/packages/release/2024-03/r/eclipse-ide-embedded-cc-developers
GNU Arm Embedded Toolchain	Armマイコンを使った組込み開発用のGNU Cコンパイラおよび関連ツール一式	https://developer.arm.com/tools-and-software/open-source-software/developer-tools/gnu-toolchain/gnu-rm/downloads
xPack Windows Build Tools	プログラムの構築（ビルド）のためのWindows用のツール一式	https://github.com/xpack-dev-tools/windows-build-tools-xpack/releases
Python	プログラミング言語Pythonの言語処理系	https://www.python.org/downloads/
pyOCD	Python上で動作するArmマイコン用のデバッグツール	(Pythonからインストール)
micro:bit用μT-Kernel 3.0	ソースコード一式（パスワード付きZIPファイル）	https://www.personal-media.co.jp/book/tw/tw_index/362.html

Eclipseを使ったプログラム開発の実際

■ (1) Eclipseのインストール

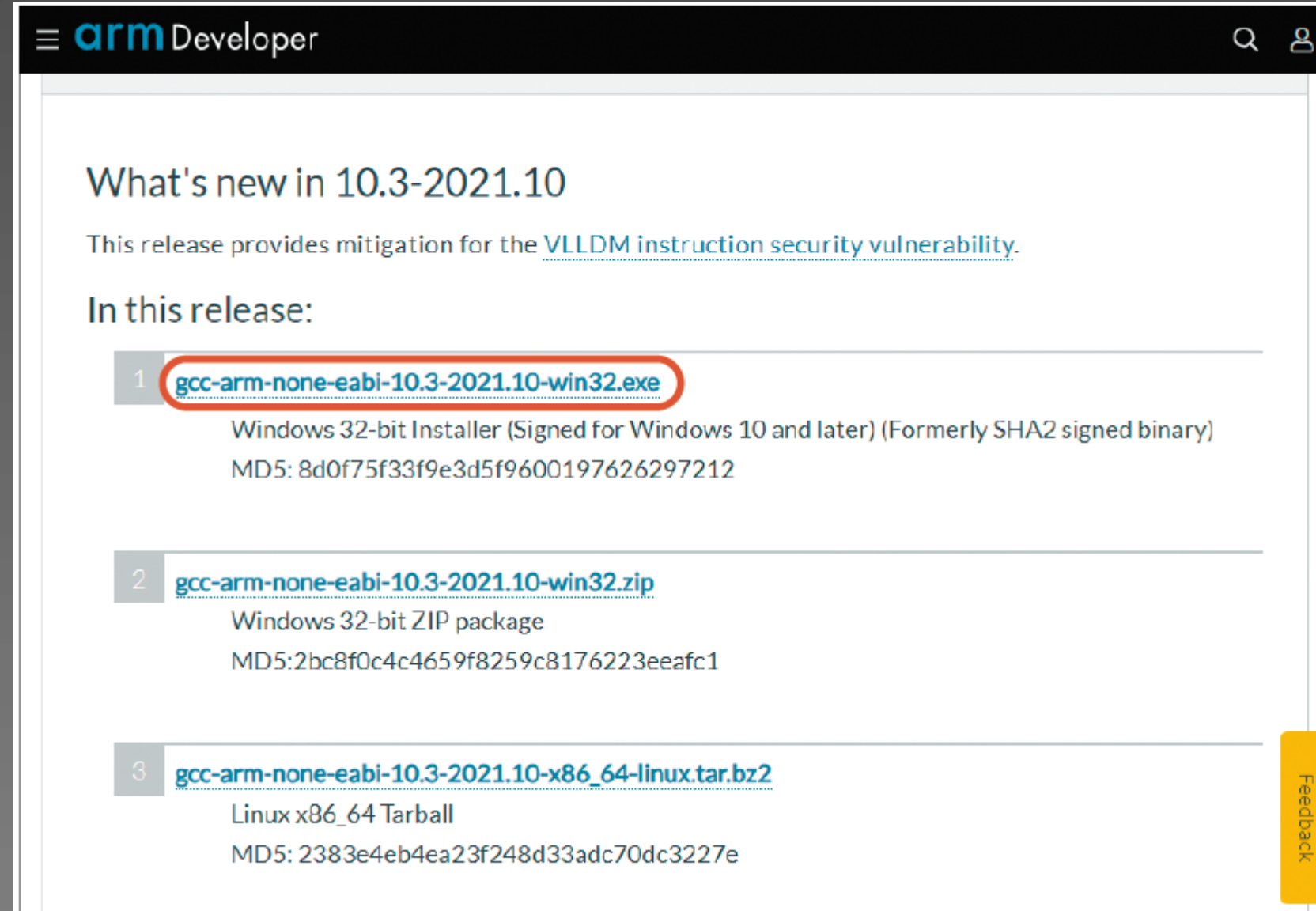
- GUI画面からソースコードの作成や編集、コンパイル、デバッグなどを行う統合開発環境



The screenshot shows the Eclipse Foundation website page for "Eclipse IDE for Embedded C/C++ Developers". The page title "Eclipse IDE for Embedded C/C++ Developers" is highlighted with a red rectangle. Below the title, the "Package Description" section states: "An IDE for Embedded C/C++ developers. It includes managed cross build plug-ins (Arm and RISC-V) and debug plug-ins (SEGGER J-Link, OpenOCD, pyocd, and QEMU), plus a number of templates to create ready to run blinky projects." The "Download Links" section lists: "Windows x86_64", "macOS x86_64 | AArch64", and "Linux x86_64 | AArch64". The "Windows x86_64" link is highlighted with a red rectangle. Below the download links, it says "Downloaded 12,121 Times" and "Checksums...". The "Bugzilla" section shows "Open Bugs: 0".

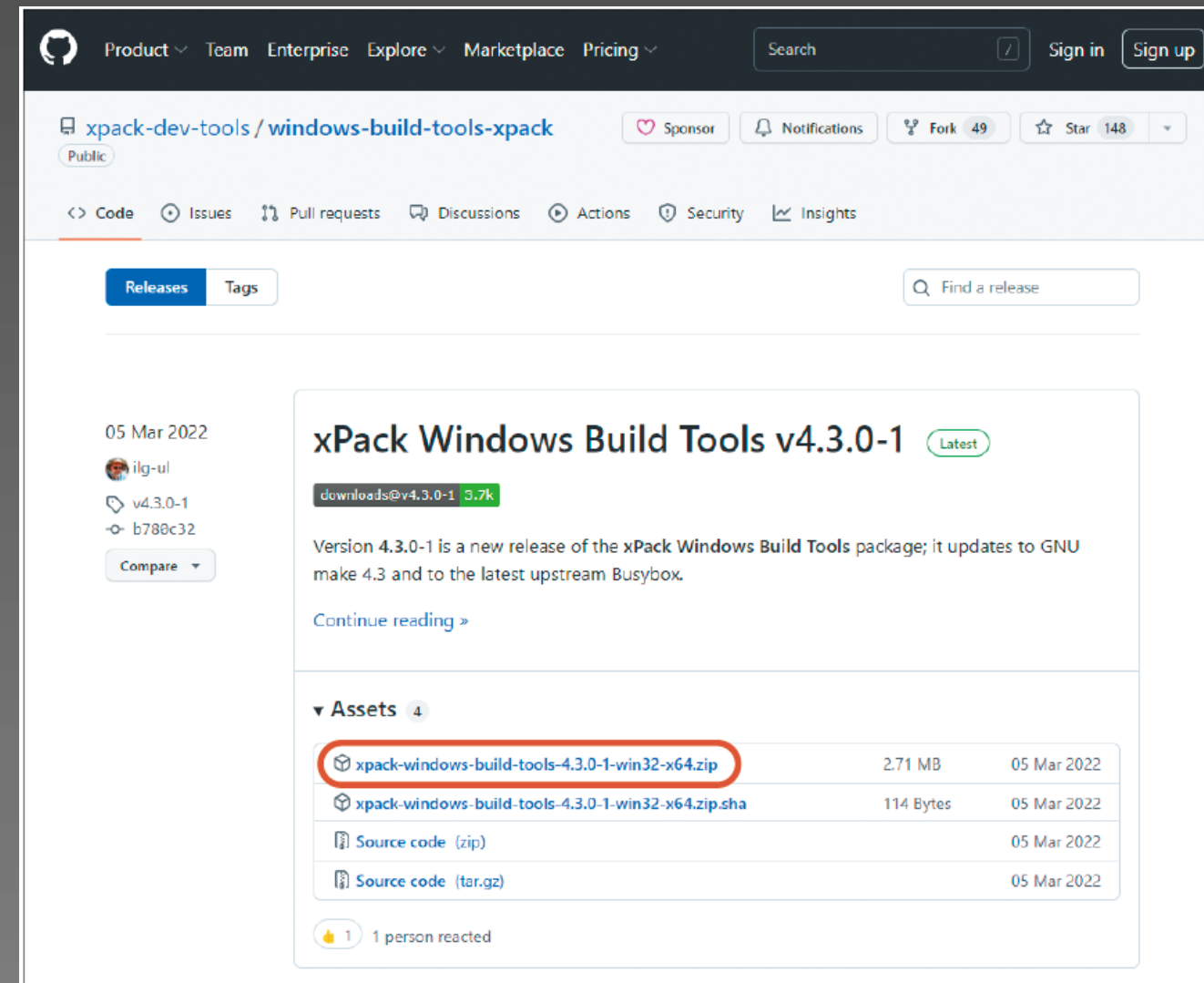
Eclipseを使ったプログラム開発の実際

- (2) GNU Arm Embedded Toolchainのインストール
 - Armマイコンを使った組み込み開発用のGNU Cコンパイラおよび関連ツール一式



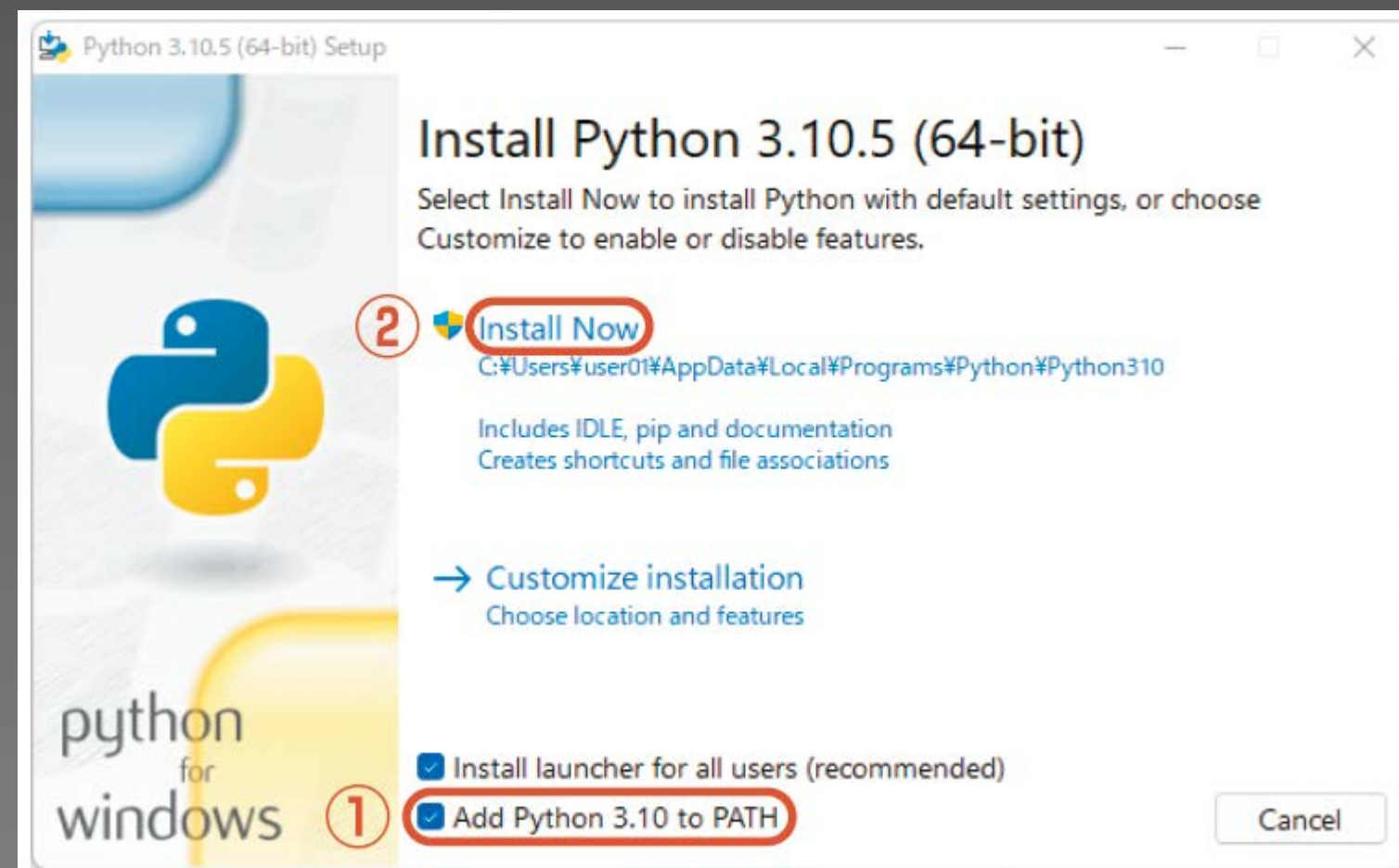
Eclipseを使ったプログラム開発の実際

- (3) xPack Windows Build Toolsのインストール
 - プログラムの構築（ビルド）のためのWindows用のツール一式



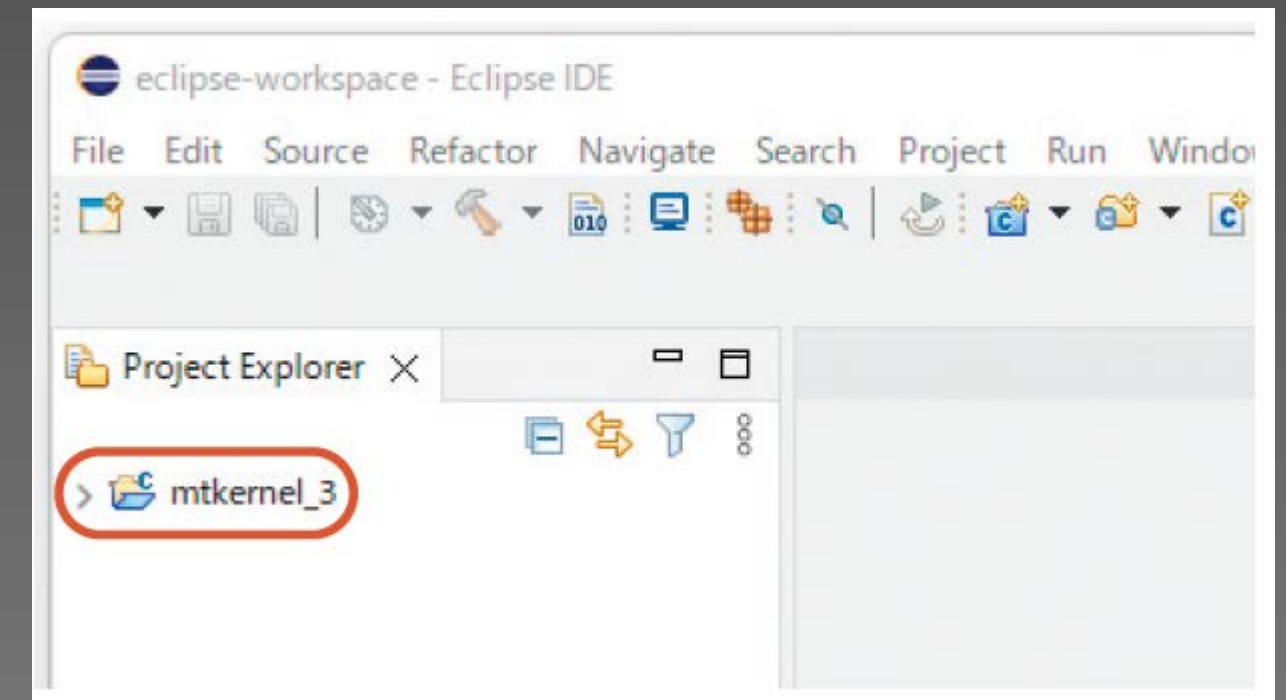
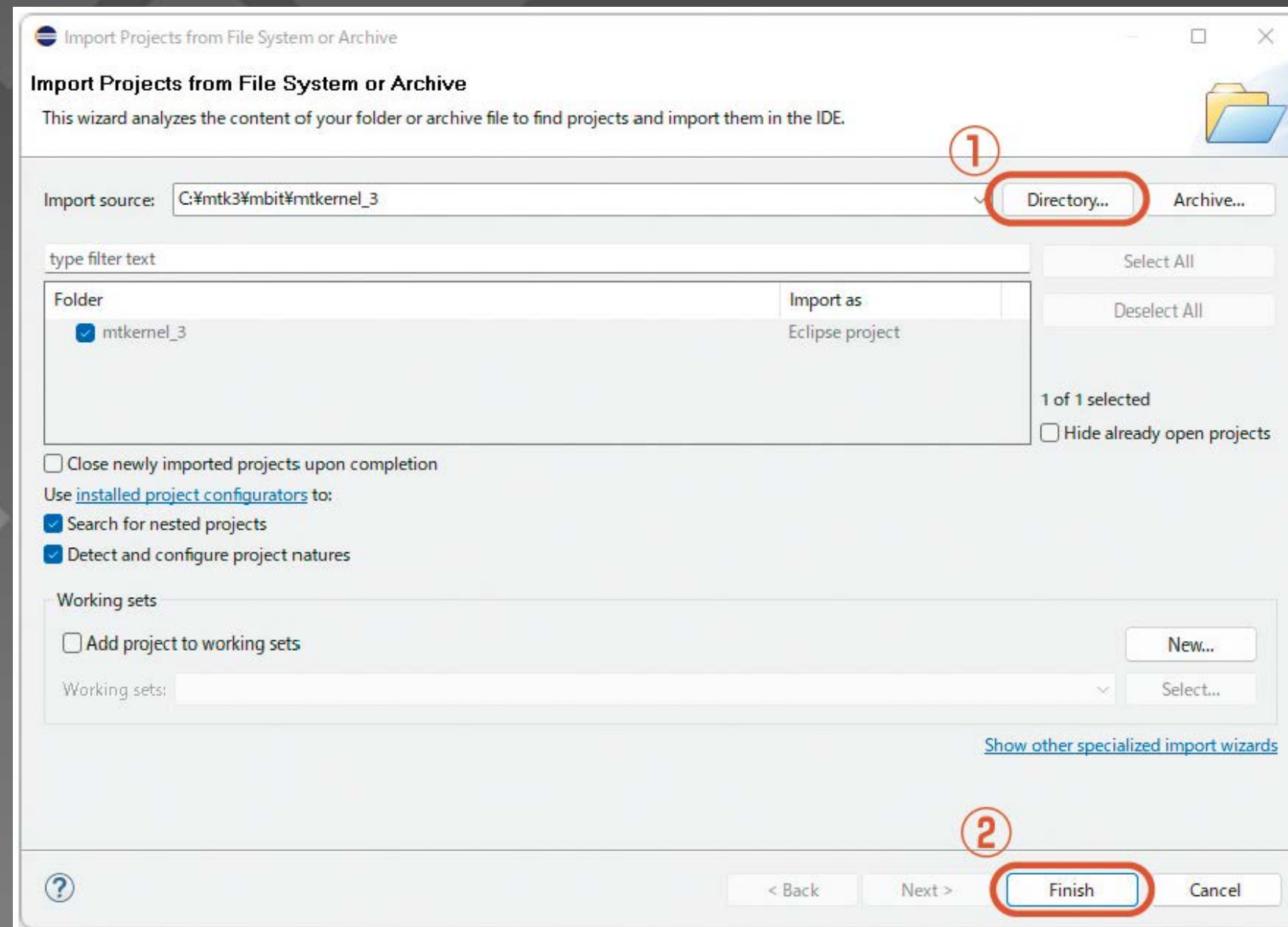
Eclipseを使ったプログラム開発の実際

- (4) PythonとpyOCDのインストール
 - Python上で動作するArmマイコン用のデバッグツール
 - Pythonを入れた後に “python -mpip install -U pyocd” を実行



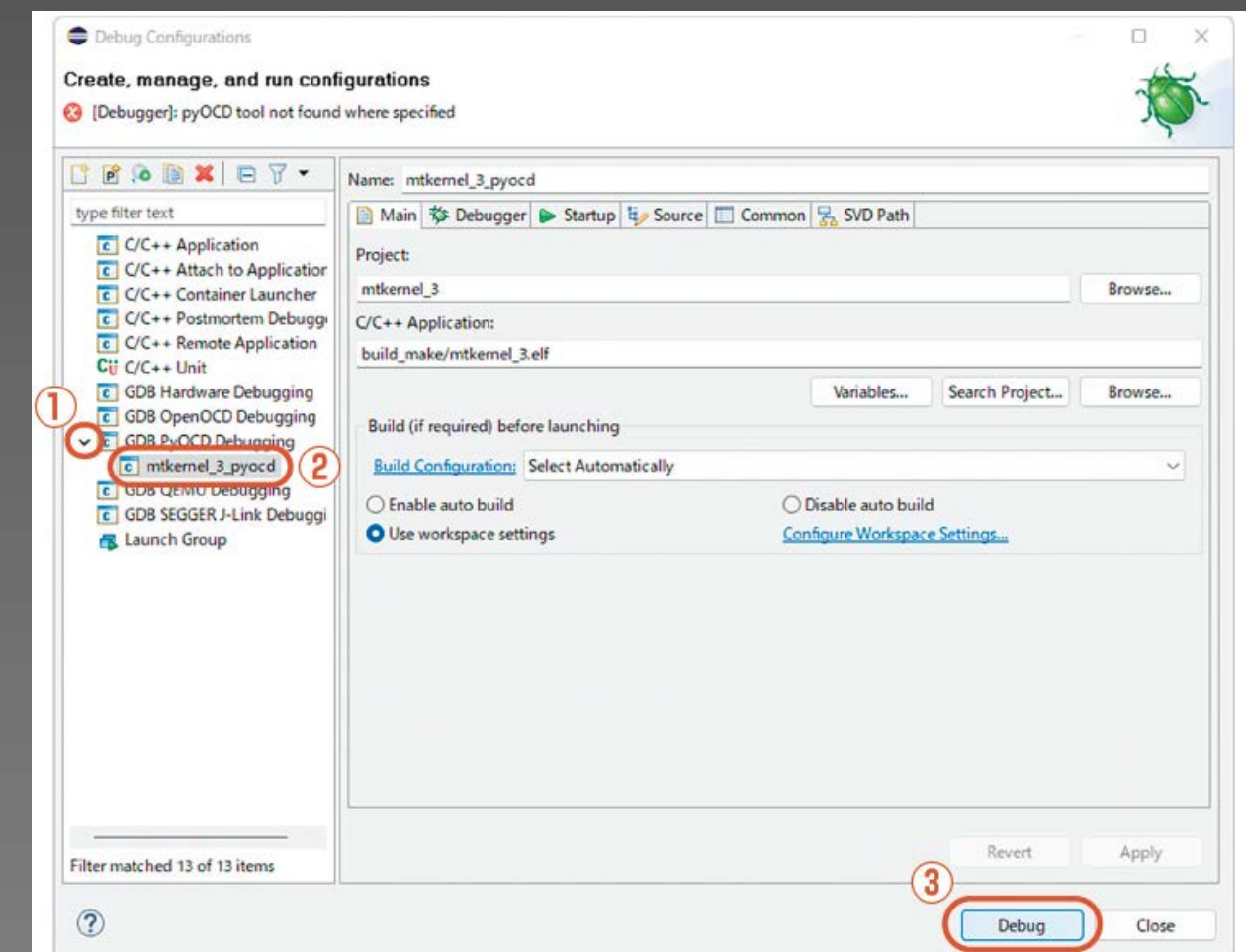
Eclipseを使ったプログラム開発の実際

■ [Step-2] μT-Kernel 3.0のインポートとコンパイル



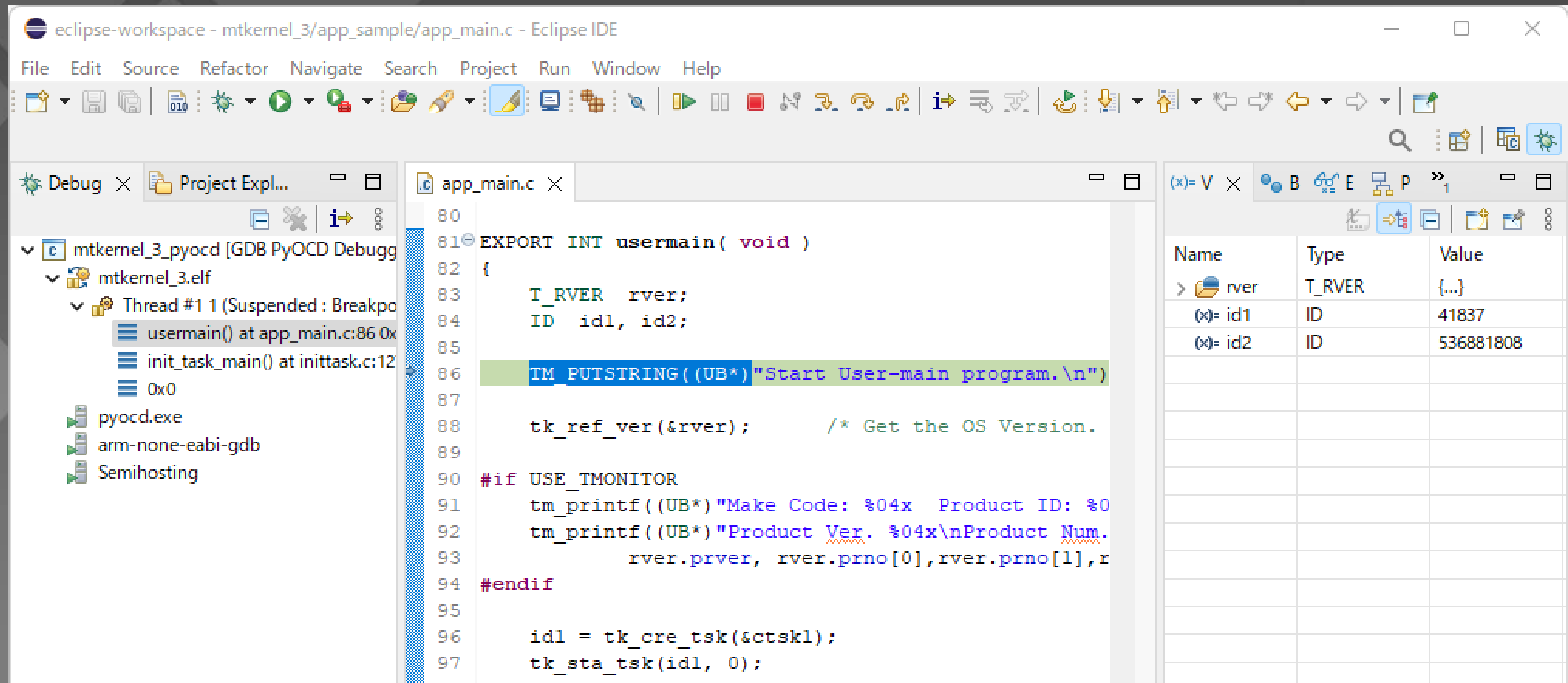
Eclipseを使ったプログラム開発の実際

- [Step-3] micro:bit実機へのμT-Kernel 3.0の転送と実行
 - (8) micro:bitのFlash ROMの消去
 - (9) 端末ソフトの起動とmicro:bit実機への接続
 - (10) デバッグ実行用のEclipseの設定
 - (11) micro:bit実機での実行



Eclipseを使ったプログラム開発の実際

- usermain() の先頭のブレークポイントでプログラムの実行が停止

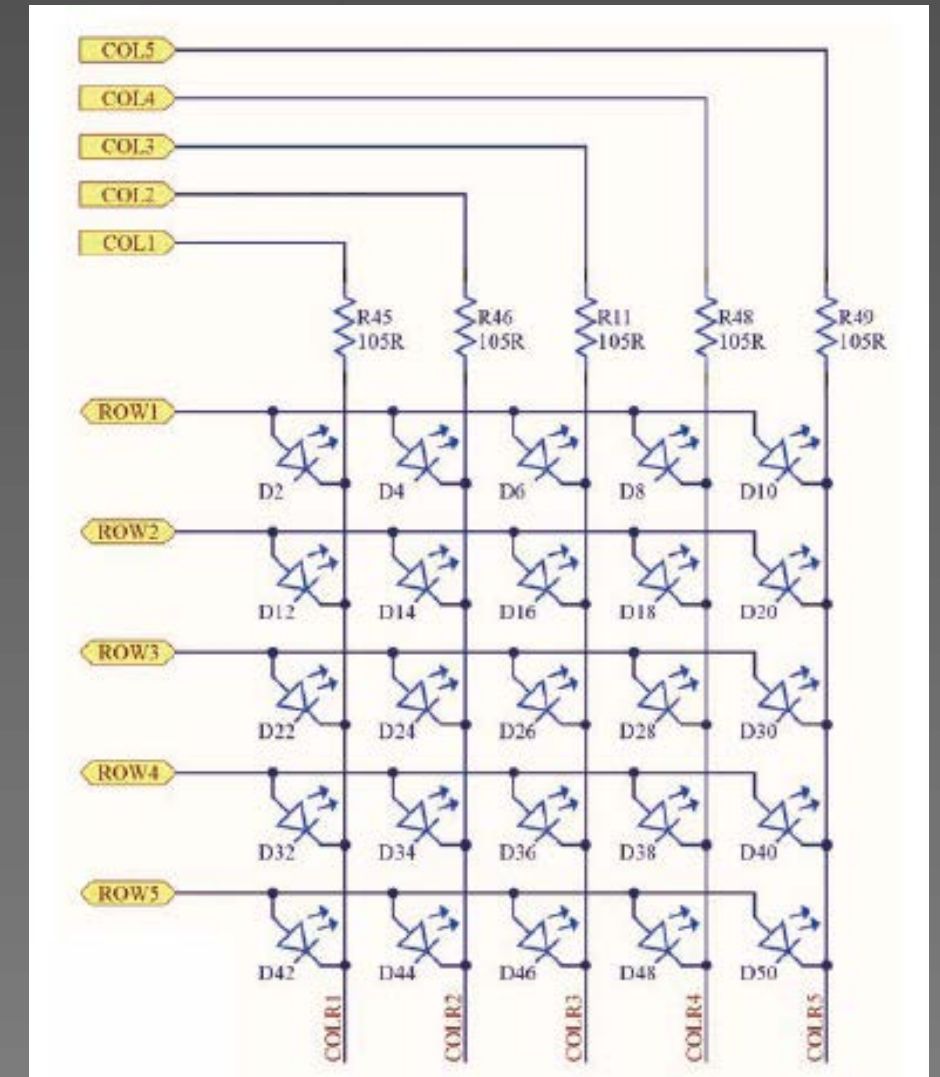
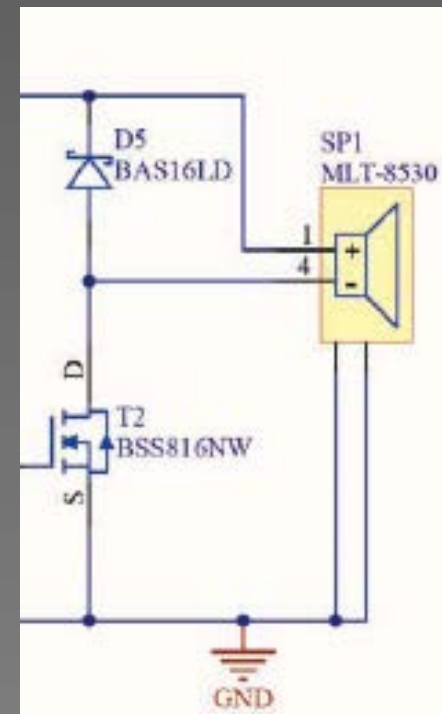
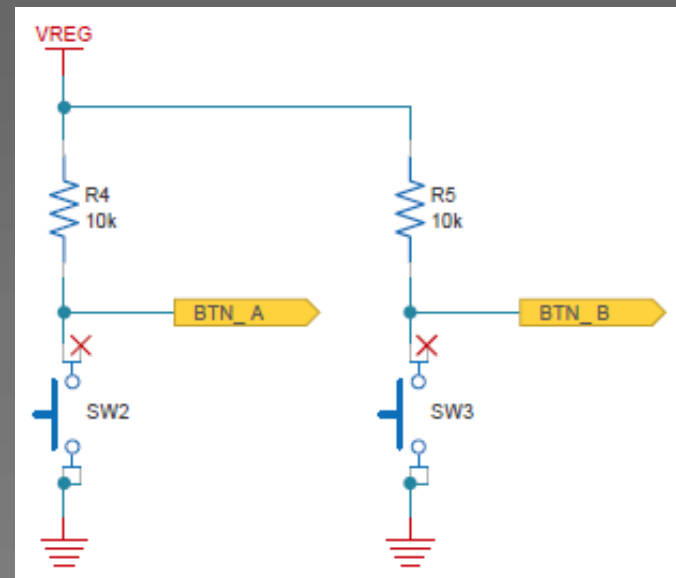




周辺デバイスを使う

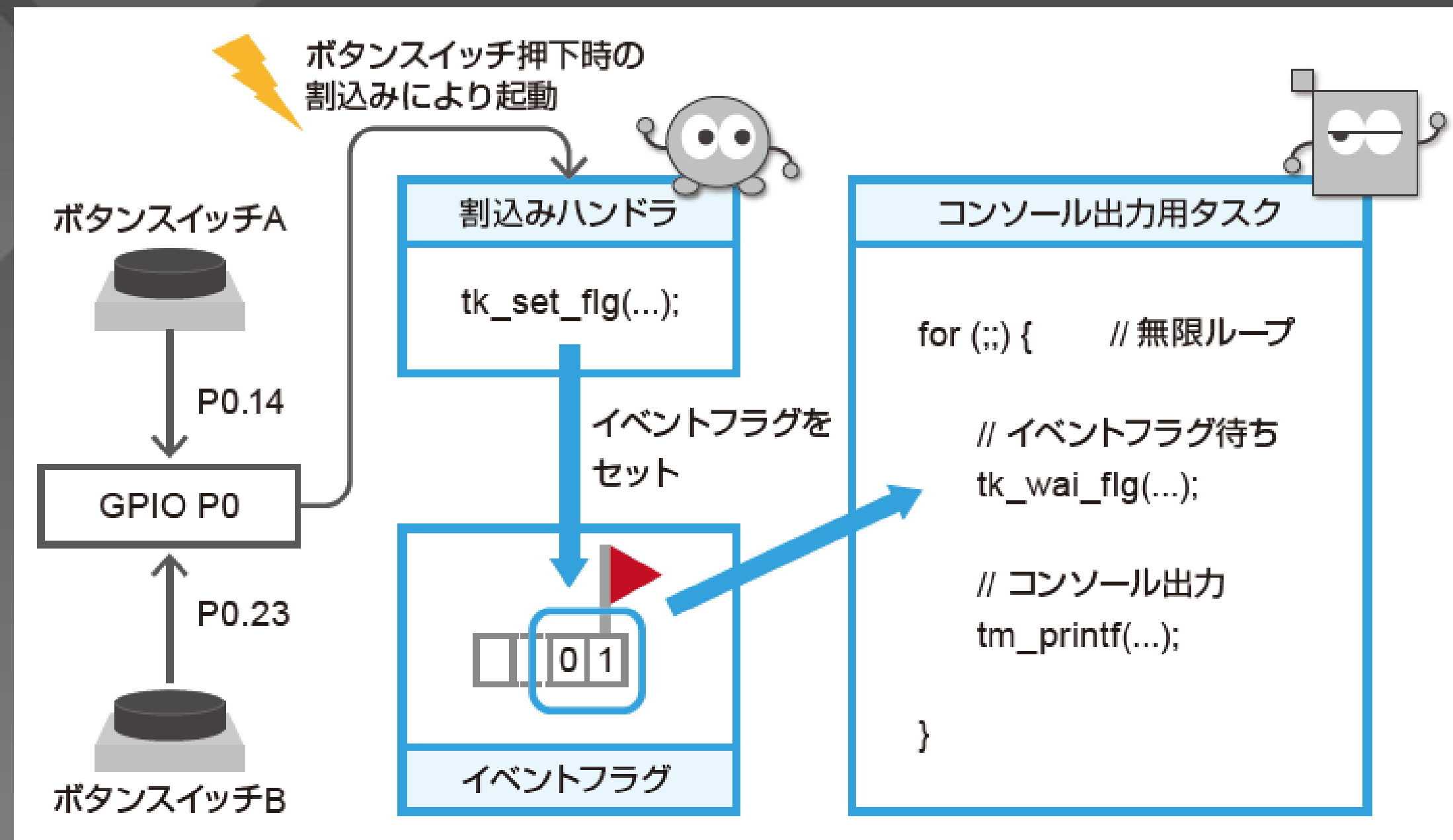
周辺デバイスを使う

- (1)GPIOとボタンスイッチ割込み
- (2)LEDマトリックスに「イ」を表示
- (3)PWMによる音階再生とLEDの調光制御
- (4)A/D変換(SAADC)とジョイスティック
- (5)I2Cと加速度センサー



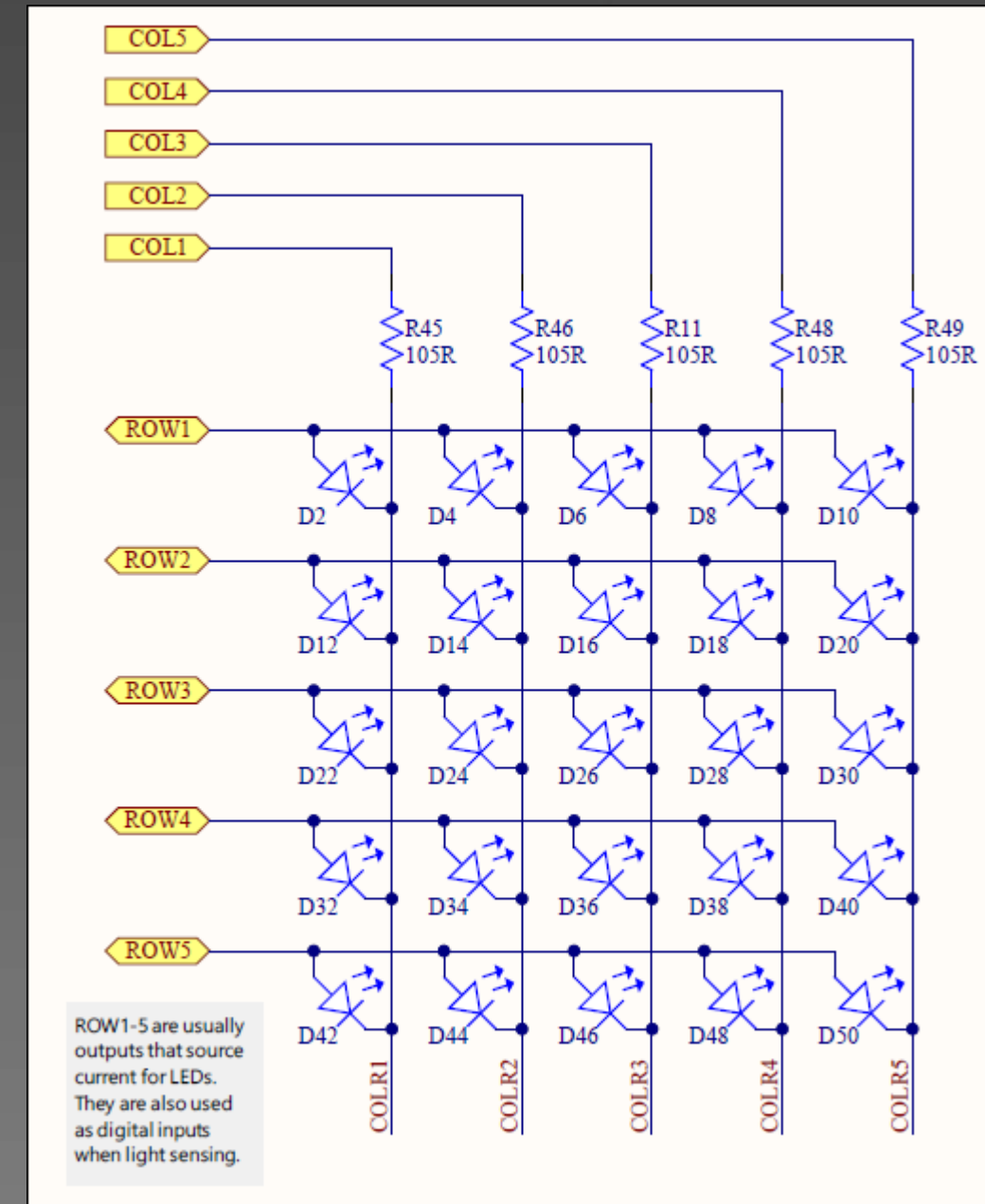
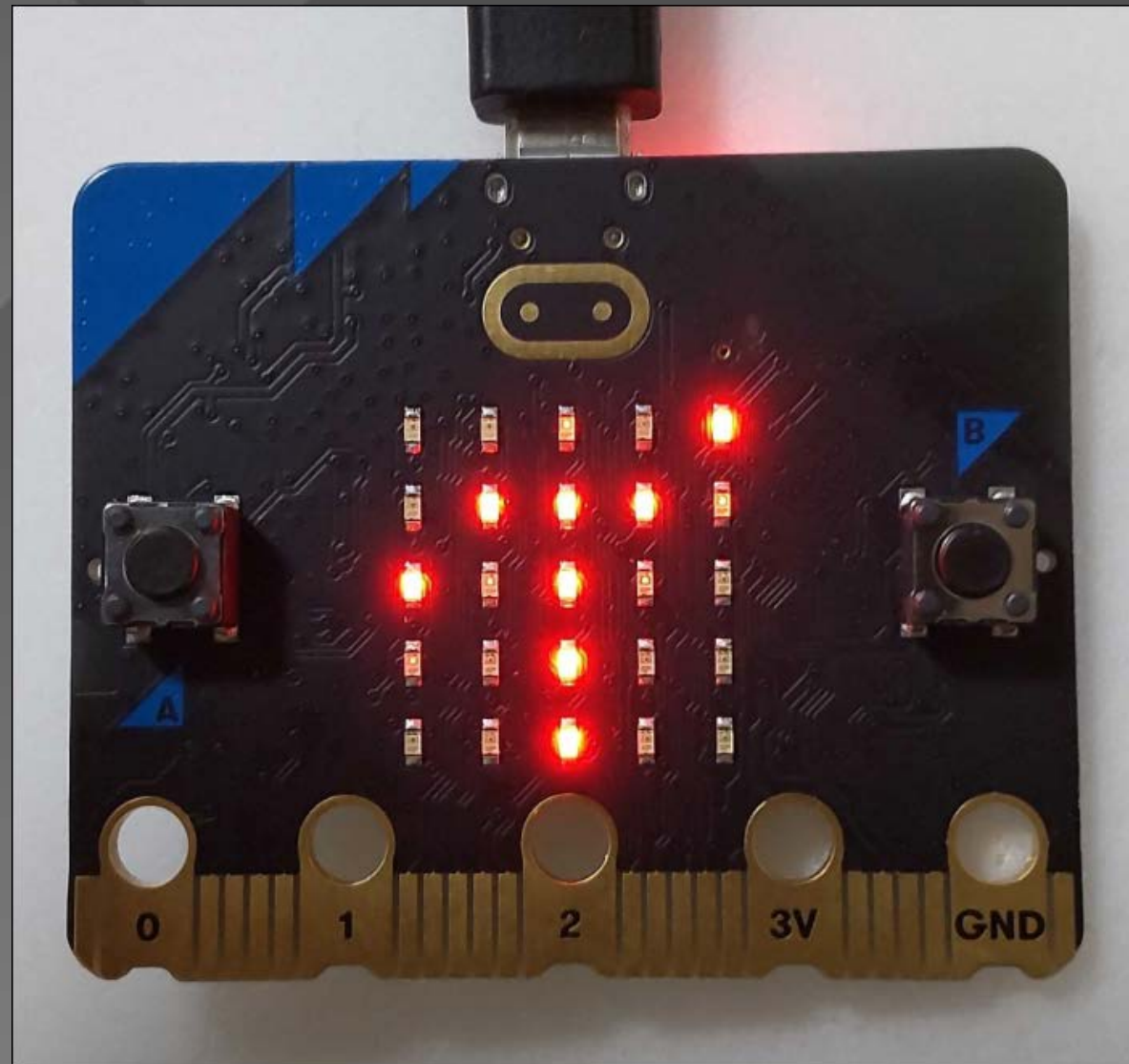
周辺デバイスを使う

■ (1)GPIOとボタンスイッチ割込み



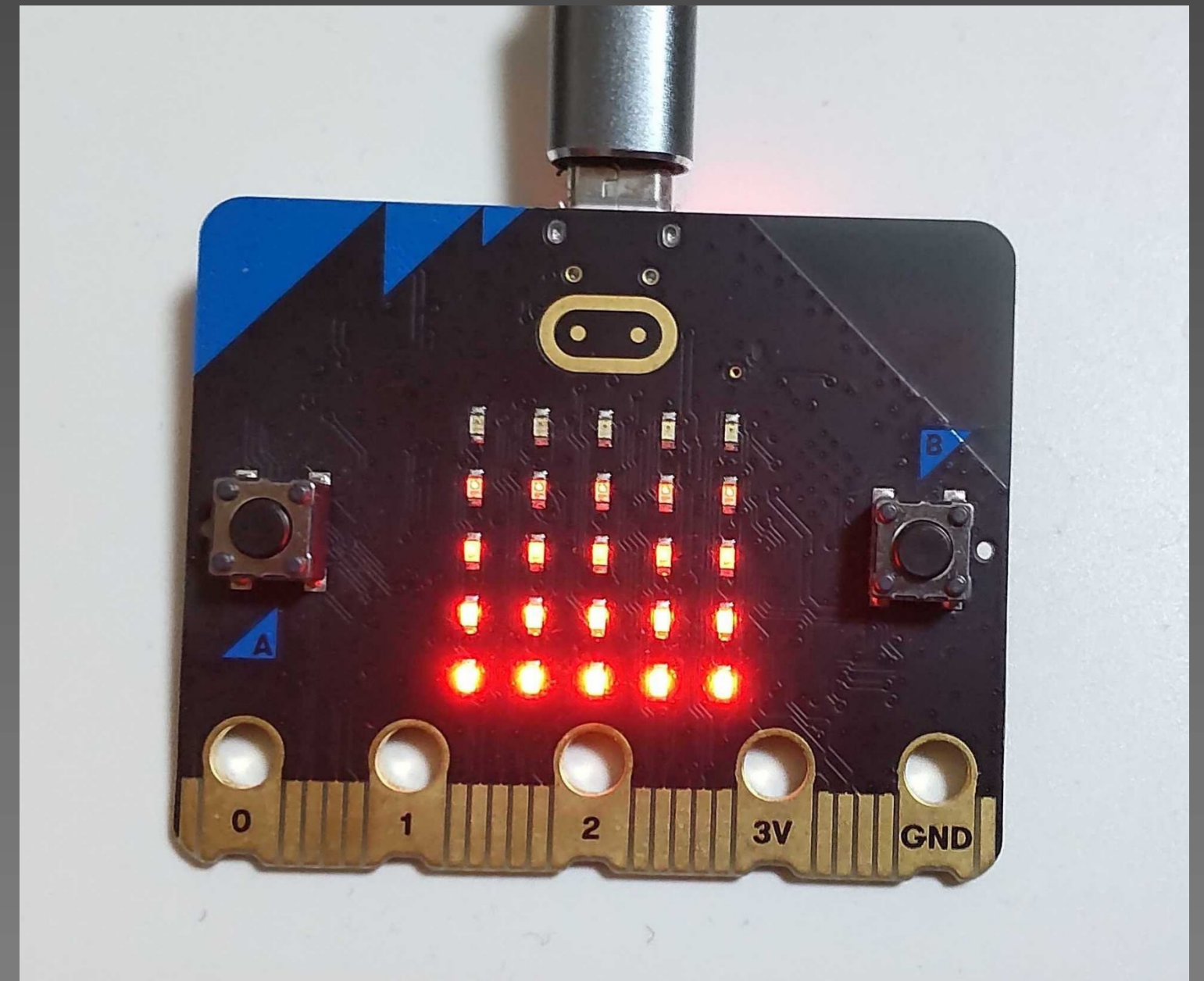
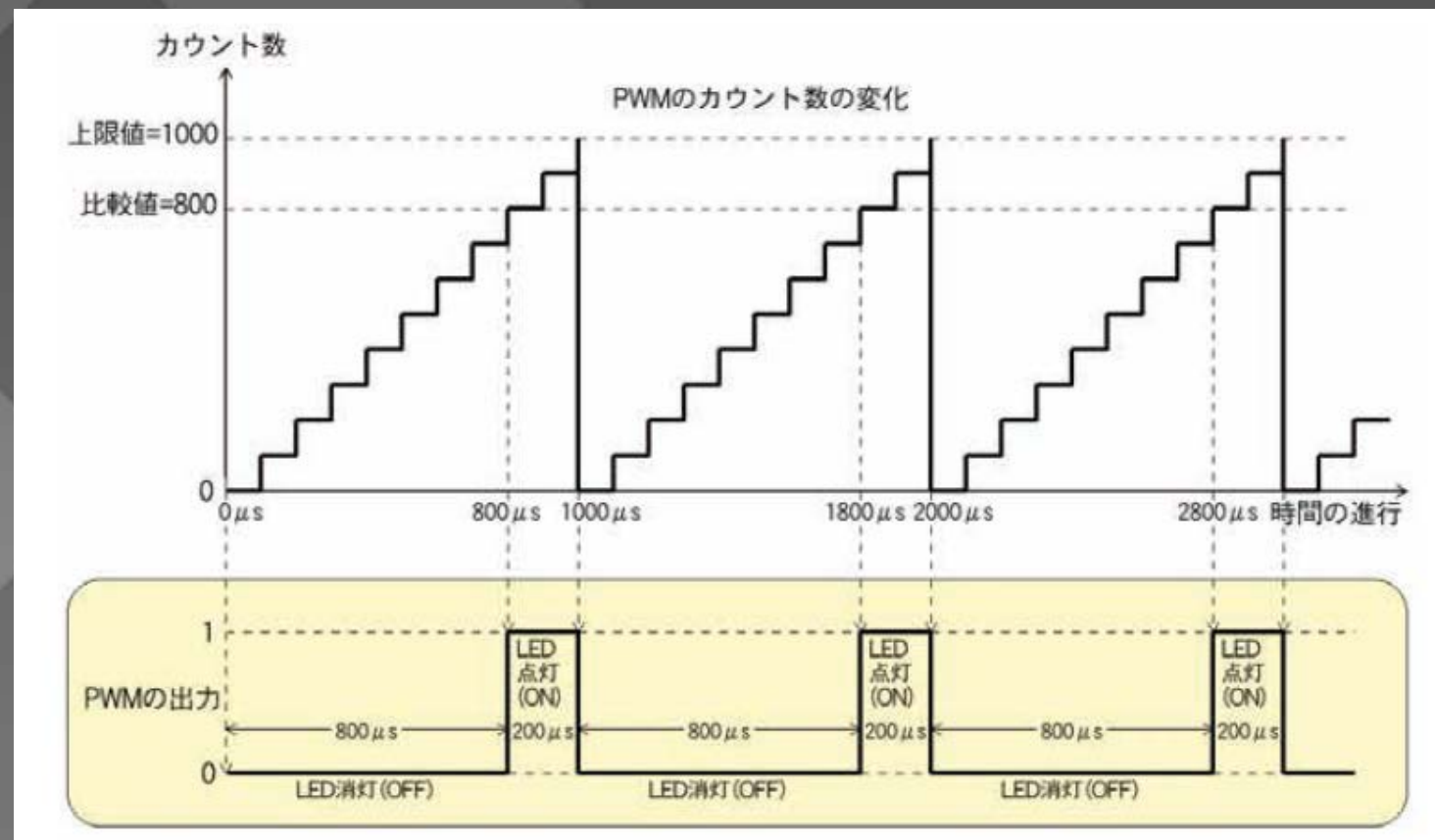
周辺デバイスを使う

■ (2)LEDマトリックスに「イ」を表示



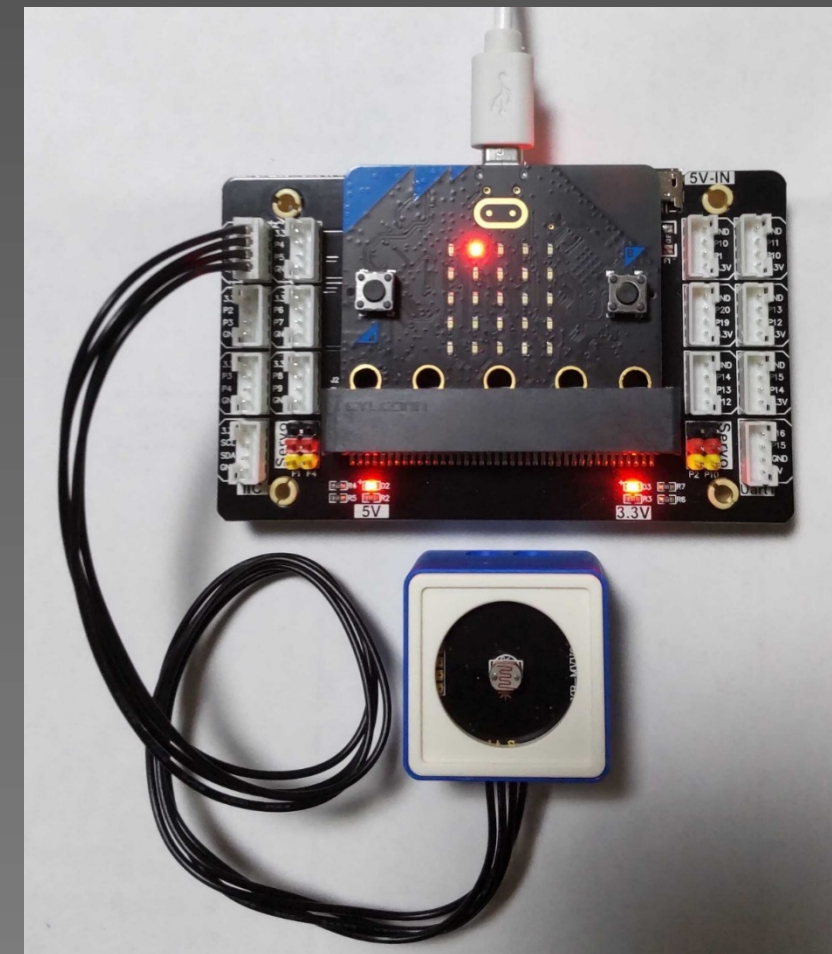
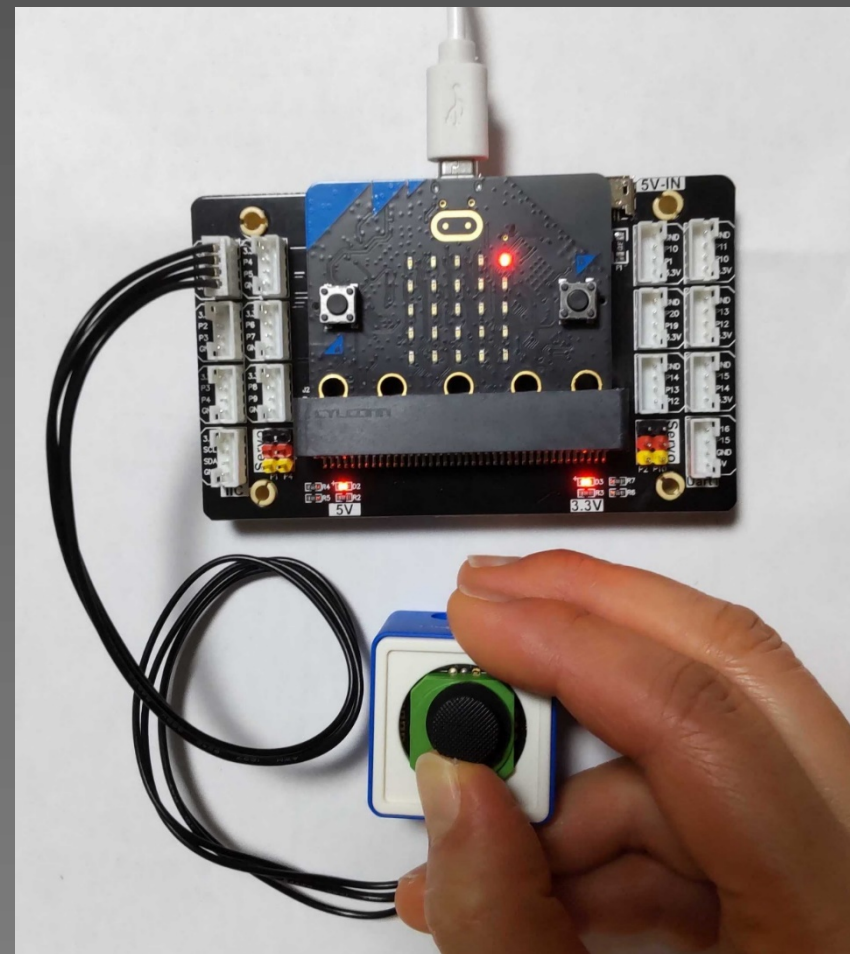
周辺デバイスを使う

■ (3)PWMによる音階再生とLEDの調光制御



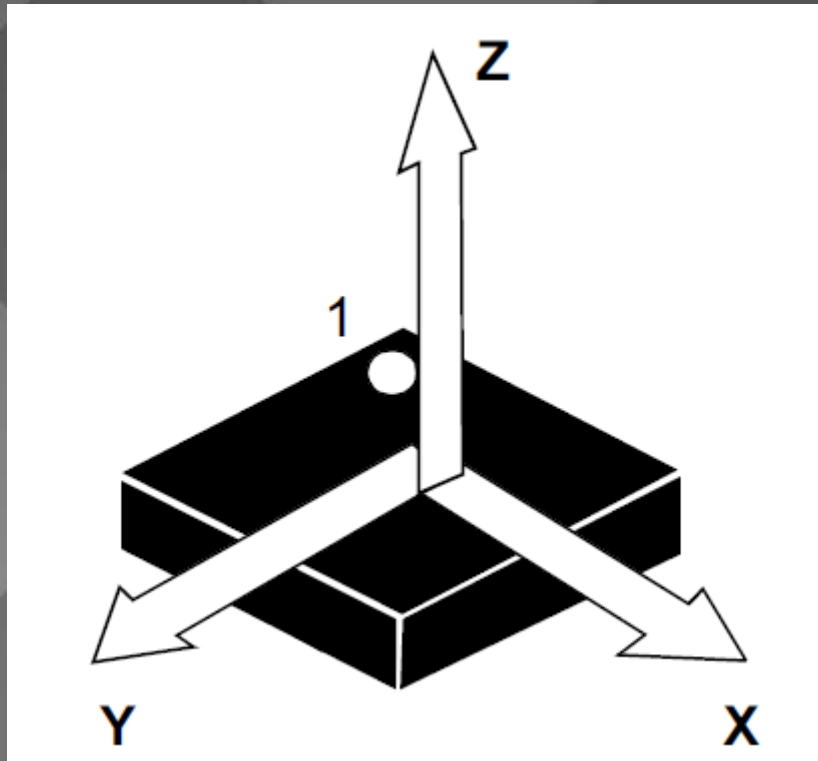
周辺デバイスを使う

- (4)A/D変換(SAADC)とジョイスティック
 - イフティニー ストアの「ワールド オブ モジュール センサーキット」を利用
 - ✓ <https://store.iftiny.com/products/yahboom-world-of-module-programmable-sensor-kit-for-microbit-v2>



周辺デバイスを使う

■ (5)I2Cと加速度センサー



microT-Kernel Version 3.00

WHO_AM_I_A = 0x33

CTRL_REG1_A = 0x57

Acc: x,y,z= -10, -6, 237

Acc: x,y,z= -8, -4, 247

Acc: x,y,z= -14, -11, 240

Acc: x,y,z= 36, 2, 241

Acc: x,y,z= 97, 3, 224

Acc: x,y,z= 127, 17, 208

Acc: x,y,z= 144, 6, 201

Acc: x,y,z= 11, 12, 259

Acc: x,y,z=-112, -5, 214

Acc: x,y,z=-165, 2, 192

← ボードが水平 (micro:bit の文字が上)

← ボードの左側を下に傾ける

← ボードの右側を下に傾ける

周辺デバイスを使う

- 紹介したサンプルプログラムはTRONWARE連載記事の付録
 - 以下の各ページの「ダウンロード」のタブを参照
- (1)GPIOとボタンスイッチ割込み (VOL.201)
 - 本文記事中のリストを参照
- (2)LEDマトリックスに「イ」を表示 (VOL.202)
 - https://www.personal-media.co.jp/book/tw/tw_index/379.html
- (3)PWMによる音階再生とLEDの調光制御 (VOL.205)
 - https://www.personal-media.co.jp/book/tw/tw_index/382.html
- (4)A/D変換(SAADC)とジョイスティック (VOL.206)
 - https://www.personal-media.co.jp/book/tw/tw_index/383.html
- (5)I2Cと加速度センサー (VOL.207,6月15日公開予定)
 - https://www.personal-media.co.jp/book/tw/tw_index/384.html



Q and A

Q and A

- (1)serial,servo,ble などを実現させるための方法が知りたいです。
- (2)ベースとなる「μT-Kernel3.0のソースコード」はTRONWARE連載記事のモノ(*1)でよいか？
 - (*1)TRONWARE VOL.196でDLサービスされた[362_mbit_mtk3.zip]
- そのほか、参加者の皆様からのご質問



www.tron.org
www.t-engine4u.com