



micro:bitで μT-Kernel 3.0を使う 2026

2026年2月5日
パーソナルメディア株式会社

Agenda

- micro:bitとは
- micro:bitでμT-Kernel 3.0を動かす
- micro:bit用μT-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
- コンテスト応募とAI活用のヒント
- Q and A

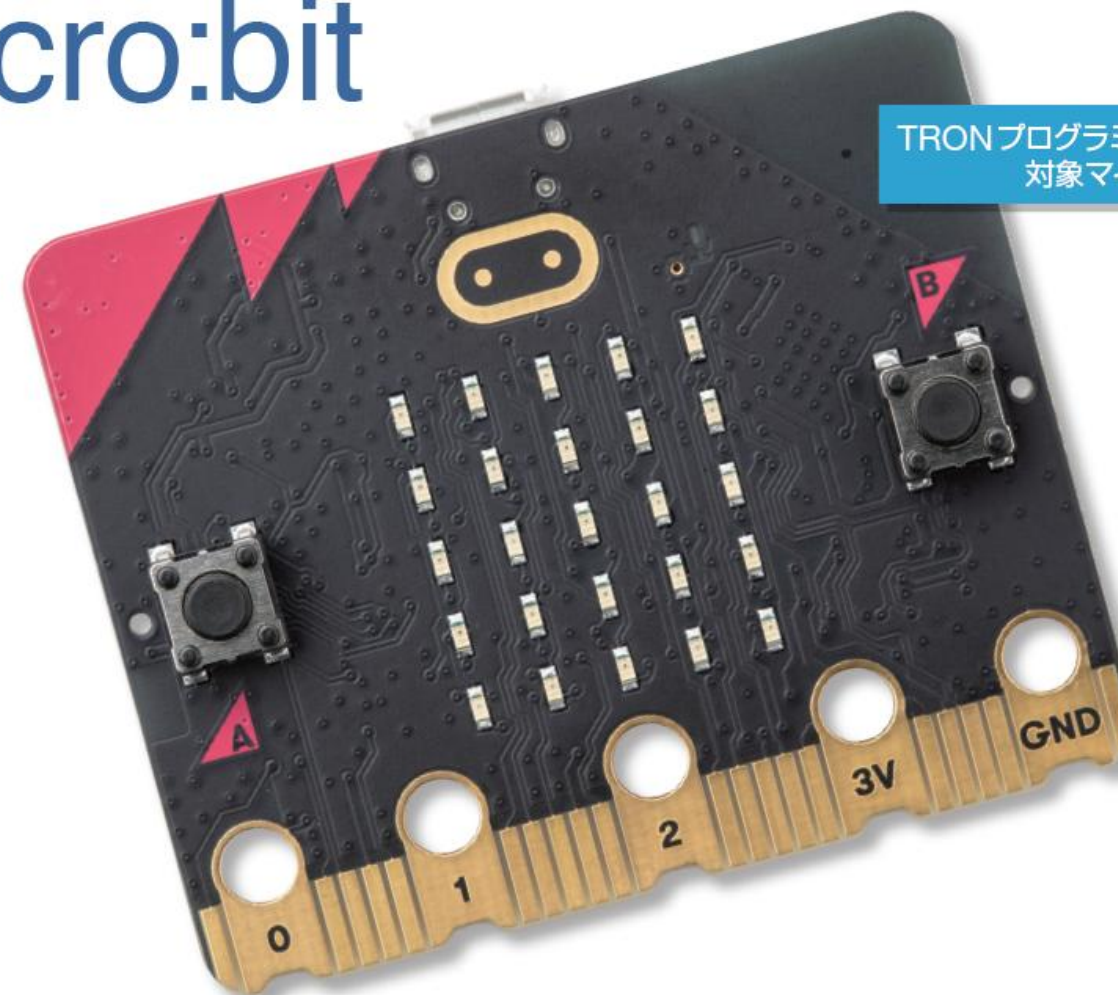


micro:bitとは

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）

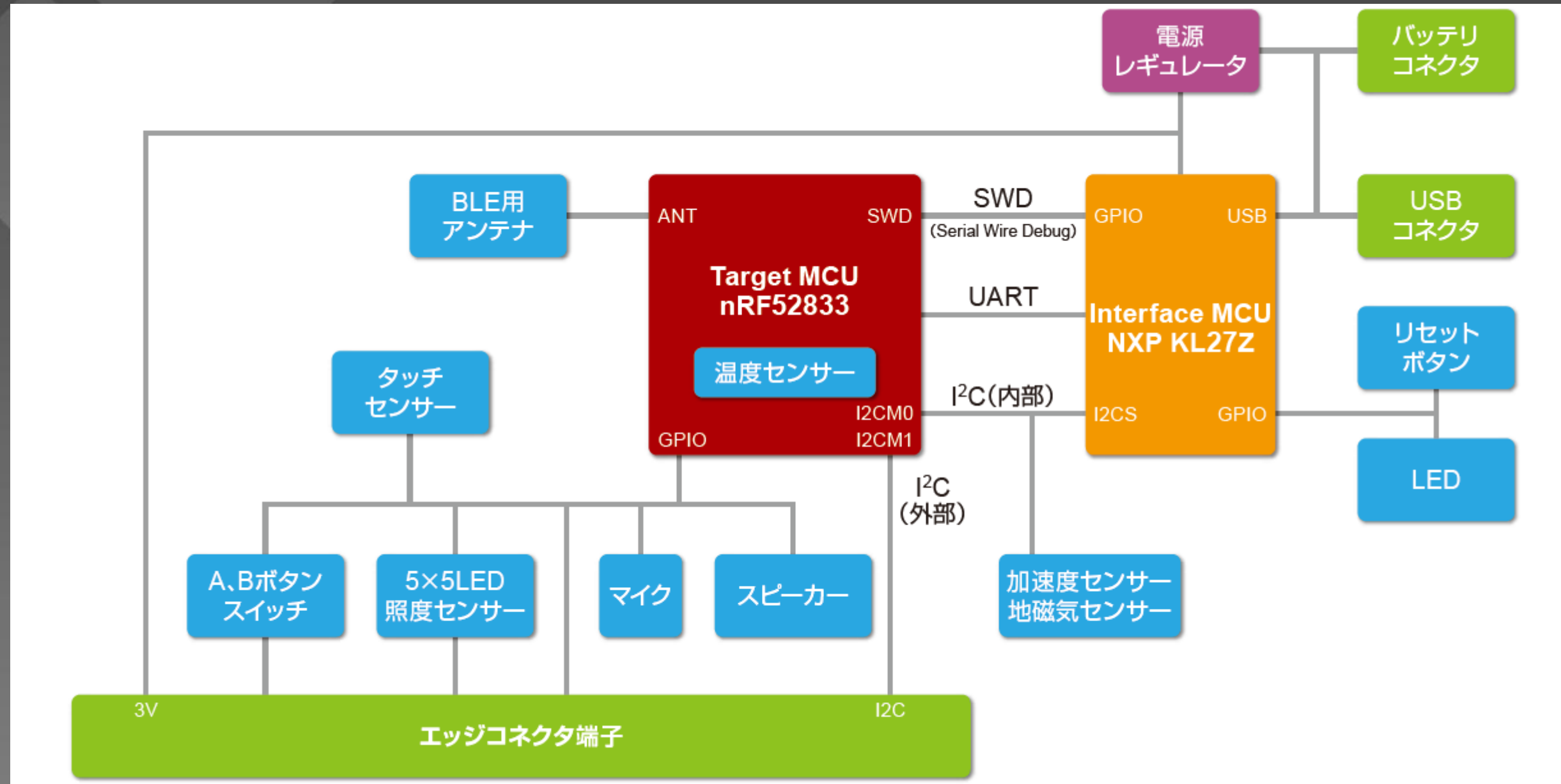
micro:bit



TRONプログラミングコンテスト
対象マイコン

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

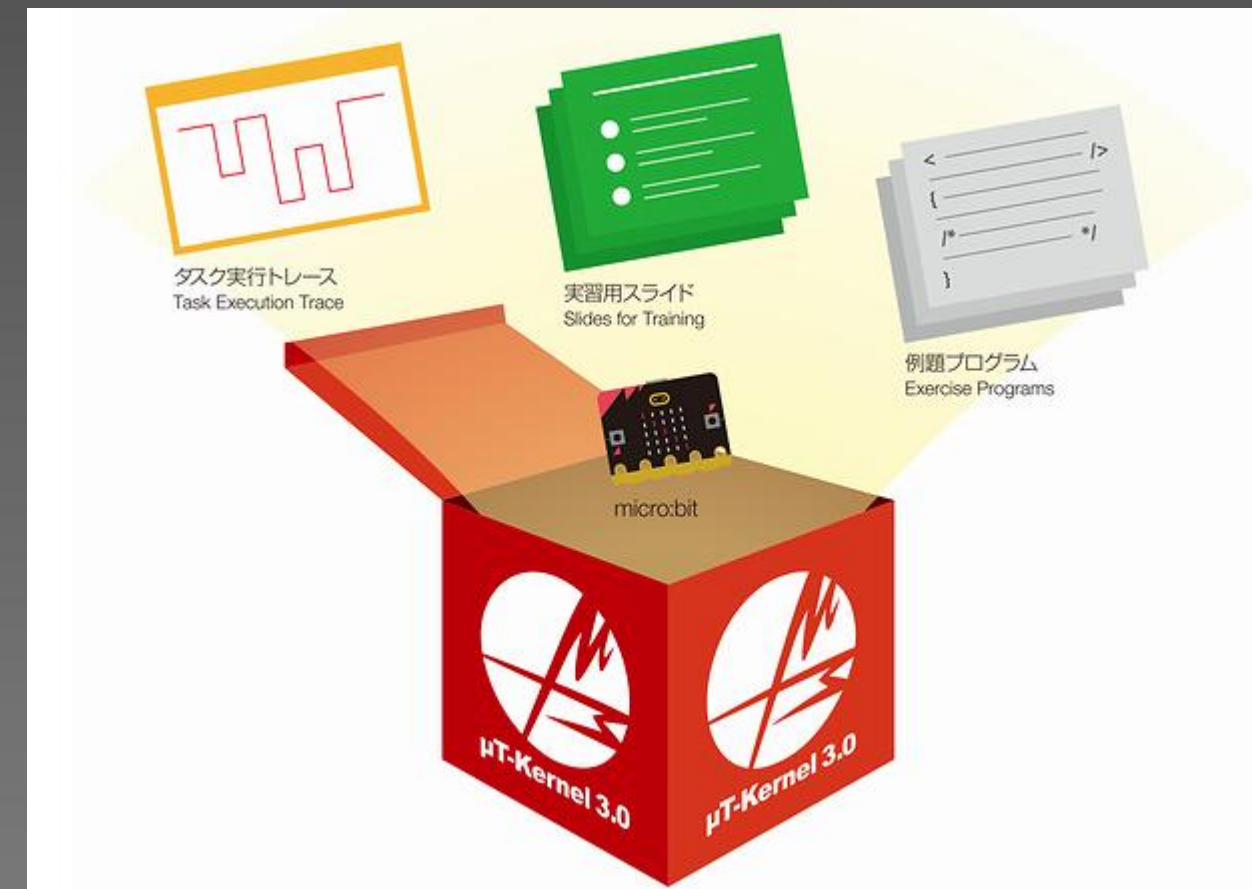
- 方法1: 「IoTエッジノード実践キット/micro:bit」を利用
- 方法2: TRONWARE記事「micro:bitでμT-Kernel 3.0を動かそう」
- 方法3: 上記の「方法2: TRONWARE記事」をウェブサイトで閲覧
 - パーソナルメディアのサイトから公開中
 - <https://www.t-engine4u.com/info/mbit/index.html>
- 使用するμT-Kernel 3.0や開発環境、開発方法はいずれも同じ
 - トロンフォーラムのμT-Kernel 3.0をmicro:bitに移植したものを利用
 - 開発環境はEclipseを利用

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 including Home, ソリューション, 製品, セミナー, 情報/書籍, サポート, and English. 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" and "IoTエッジノード実践キット micro:bit". A large image of the micro:bit board is shown with two callout bubbles: "タスクのスケジューリングとは?" (What is task scheduling?) and "セマフォによる排他制御とは?" (What is mutual exclusion control using semaphores?). The left sidebar lists various products, with "IoTエッジノード 実践キット/micro:bit" highlighted.



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3: Web版「micro:bitでμT-Kernel 3.0を動かそう」

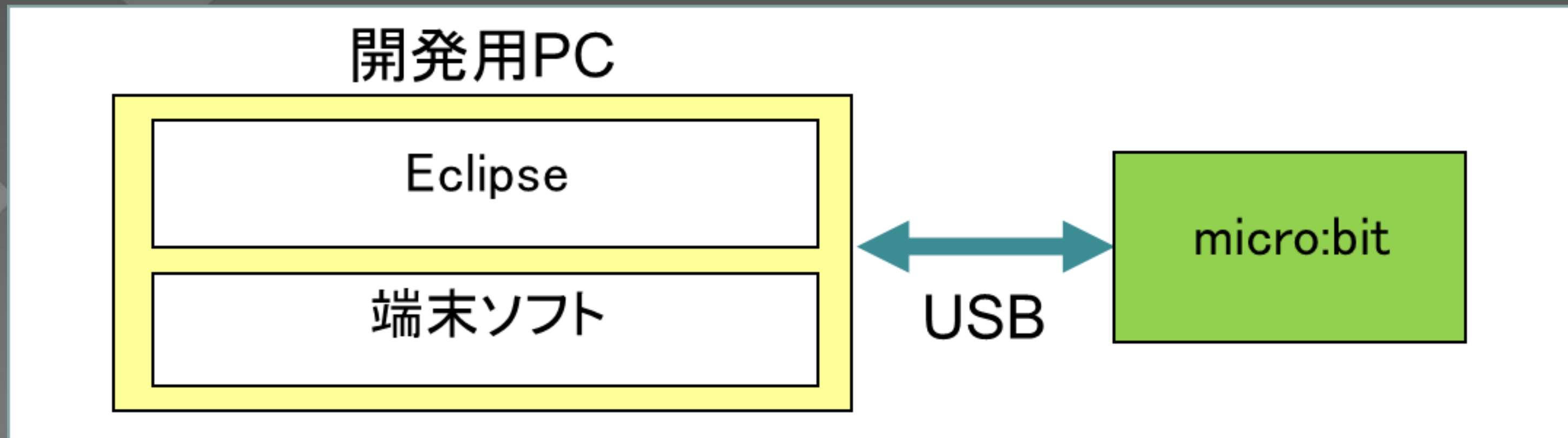




micro:bit用 μT-Kernel 3.0の開発環境

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μT-Kernel 3.0の開発環境

- アプリ開発にはμT-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
 - ✓ 2台のmicro:bit同士でシリアル通信



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] μT-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] μT-Kernel 3.0のインポートとコンパイル
 - (5) micro:bit用μT-Kernel 3.0のソースコード入手と展開
 - (6) EclipseにμT-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

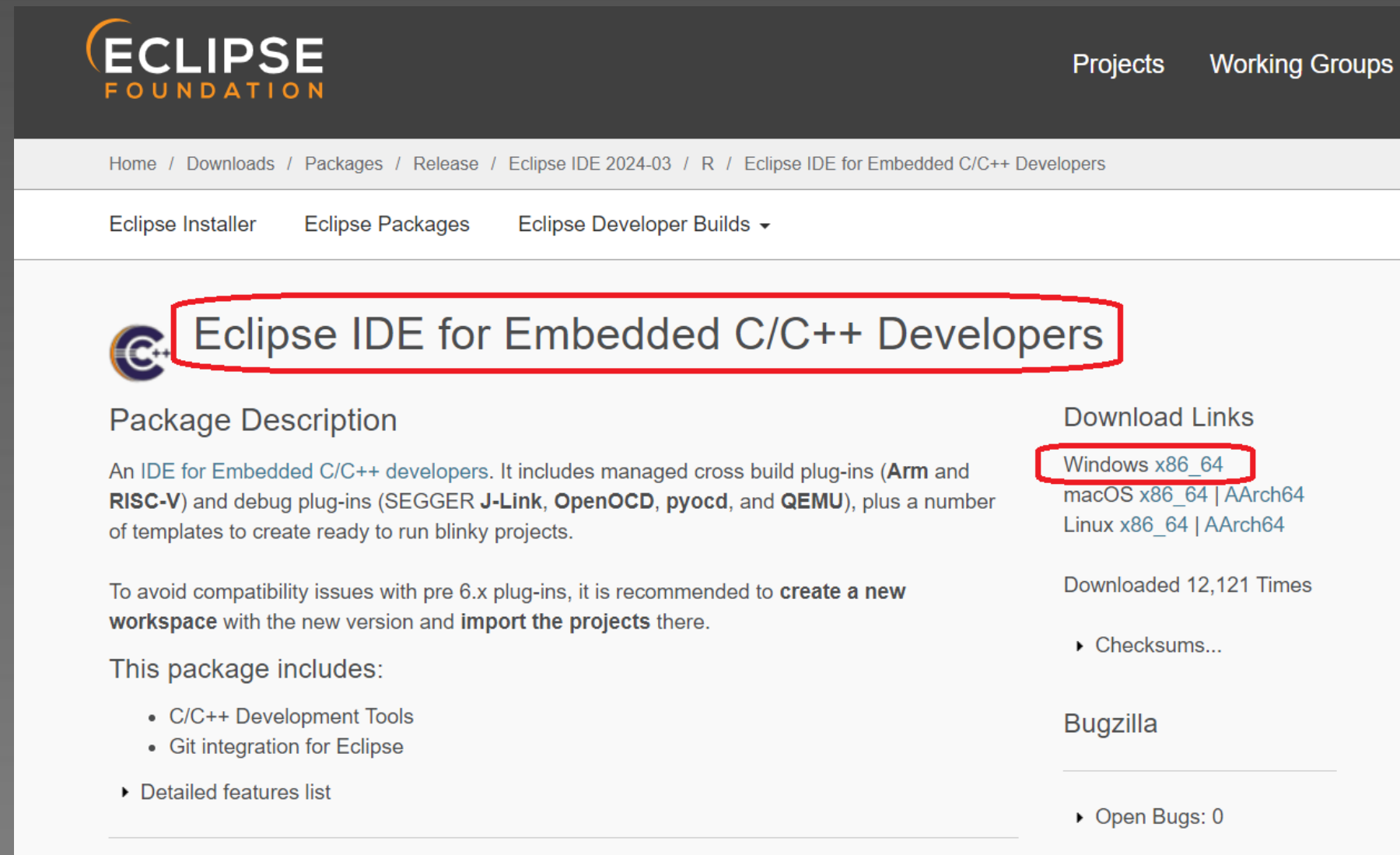
Eclipseを使ったプログラム開発の実際

■ 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 is "Eclipse IDE for Embedded C/C++ Developers", which is highlighted with a red rectangle. Below the title, there is a "Package Description" section and a "Download Links" section. The "Download Links" section lists three options: "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...". There is also a "Bugzilla" section with "Open Bugs: 0".

ECLIPSE FOUNDATION Projects Working Groups

Home / Downloads / Packages / Release / Eclipse IDE 2024-03 / R / Eclipse IDE for Embedded C/C++ Developers

Eclipse Installer Eclipse Packages Eclipse Developer Builds ▾

Eclipse IDE for Embedded C/C++ Developers

Package Description

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.

To avoid compatibility issues with pre 6.x plug-ins, it is recommended to **create a new workspace** with the new version and **import the projects** there.

This package includes:

- C/C++ Development Tools
- Git integration for Eclipse

▸ Detailed features list

Download Links

Windows x86_64
macOS x86_64 | AArch64
Linux x86_64 | AArch64

Downloaded 12,121 Times

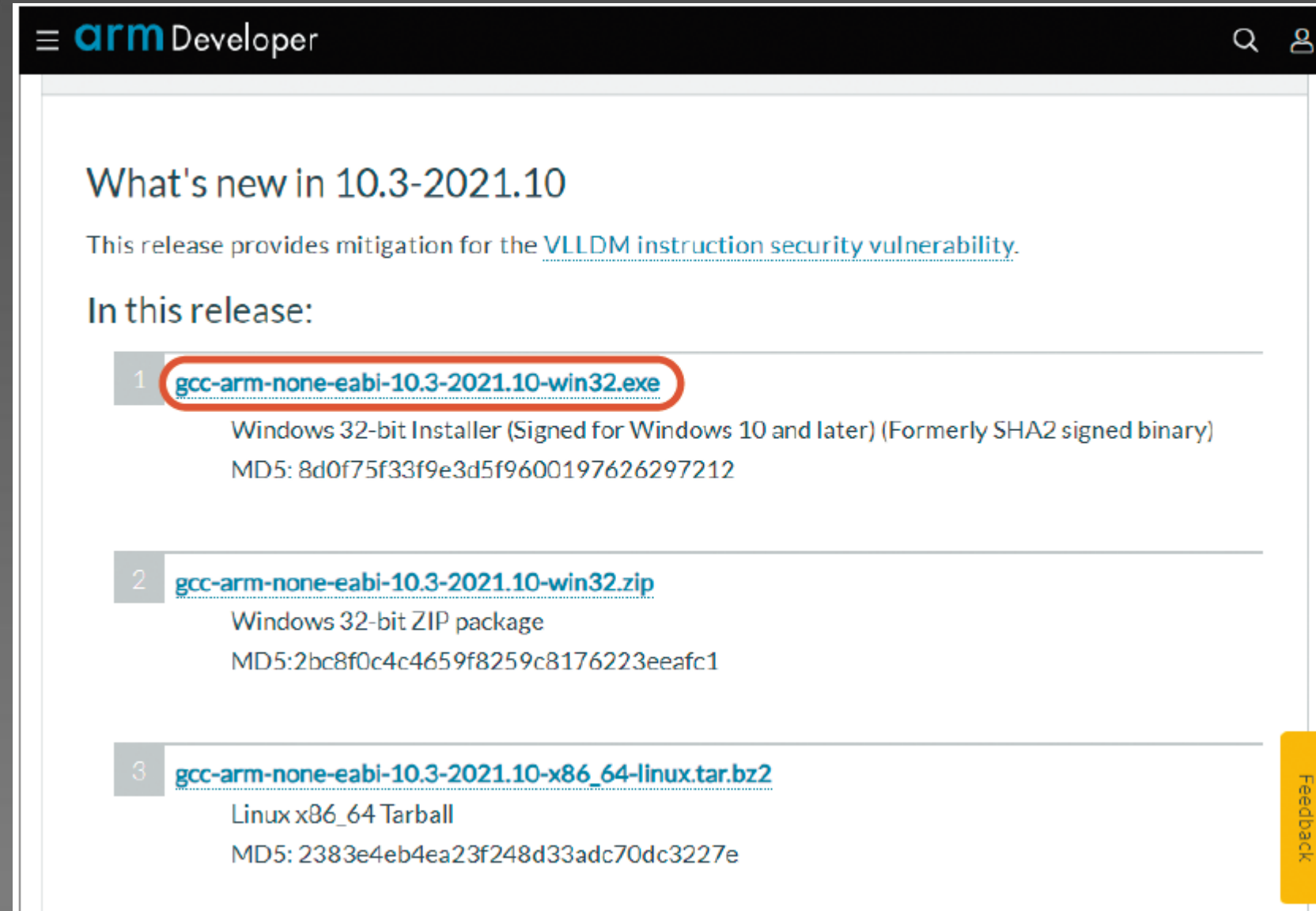
▸ Checksums...

Bugzilla

▸ Open Bugs: 0

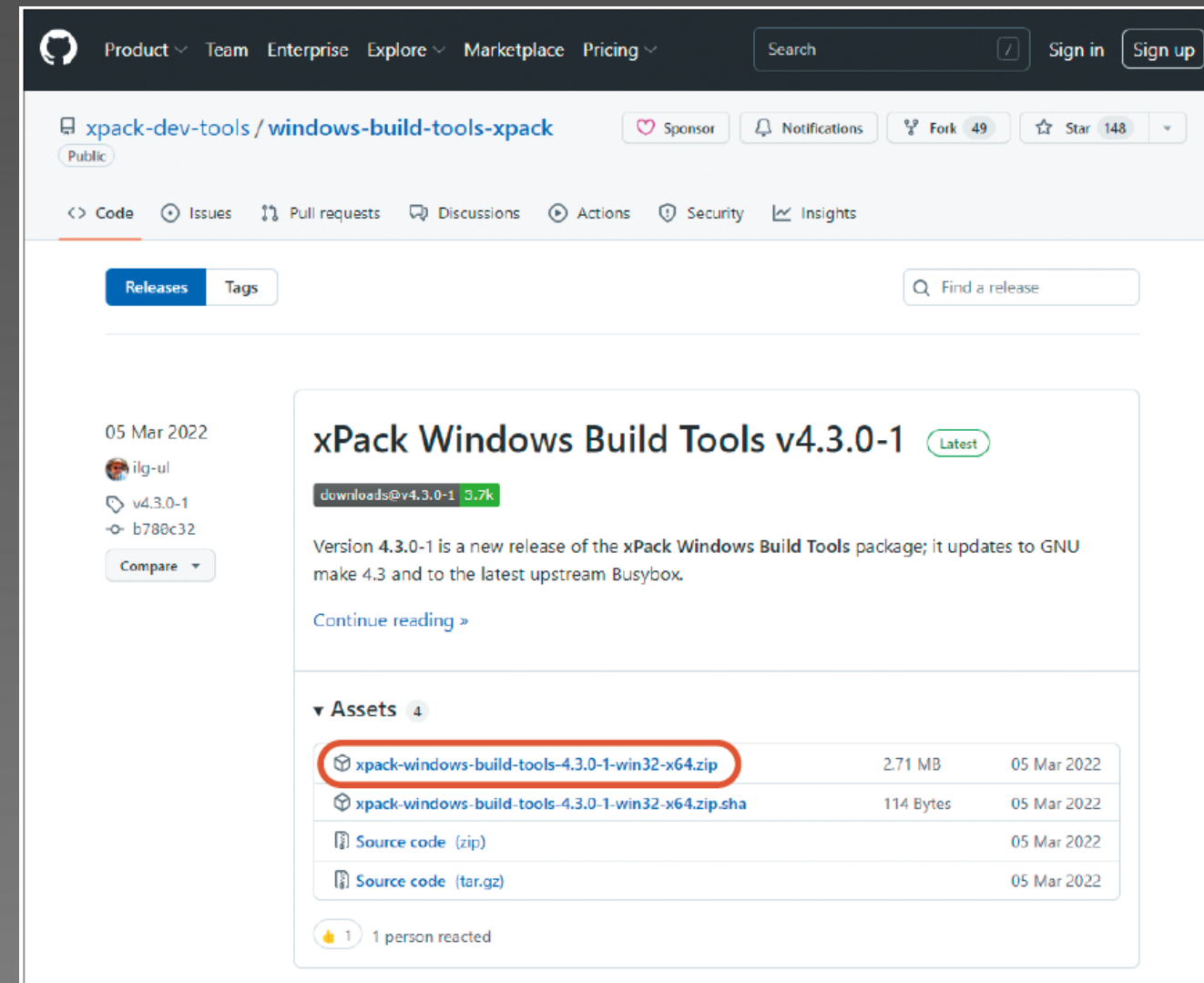
Eclipseを使ったプログラム開発の実際

- (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



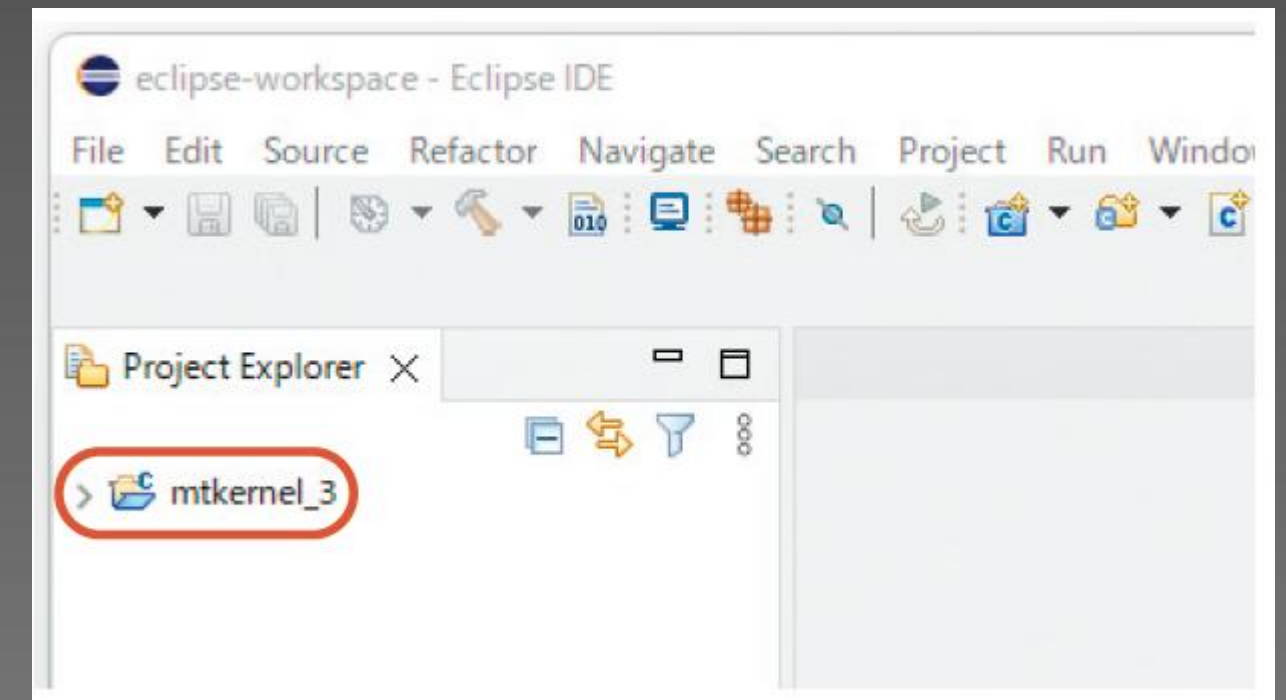
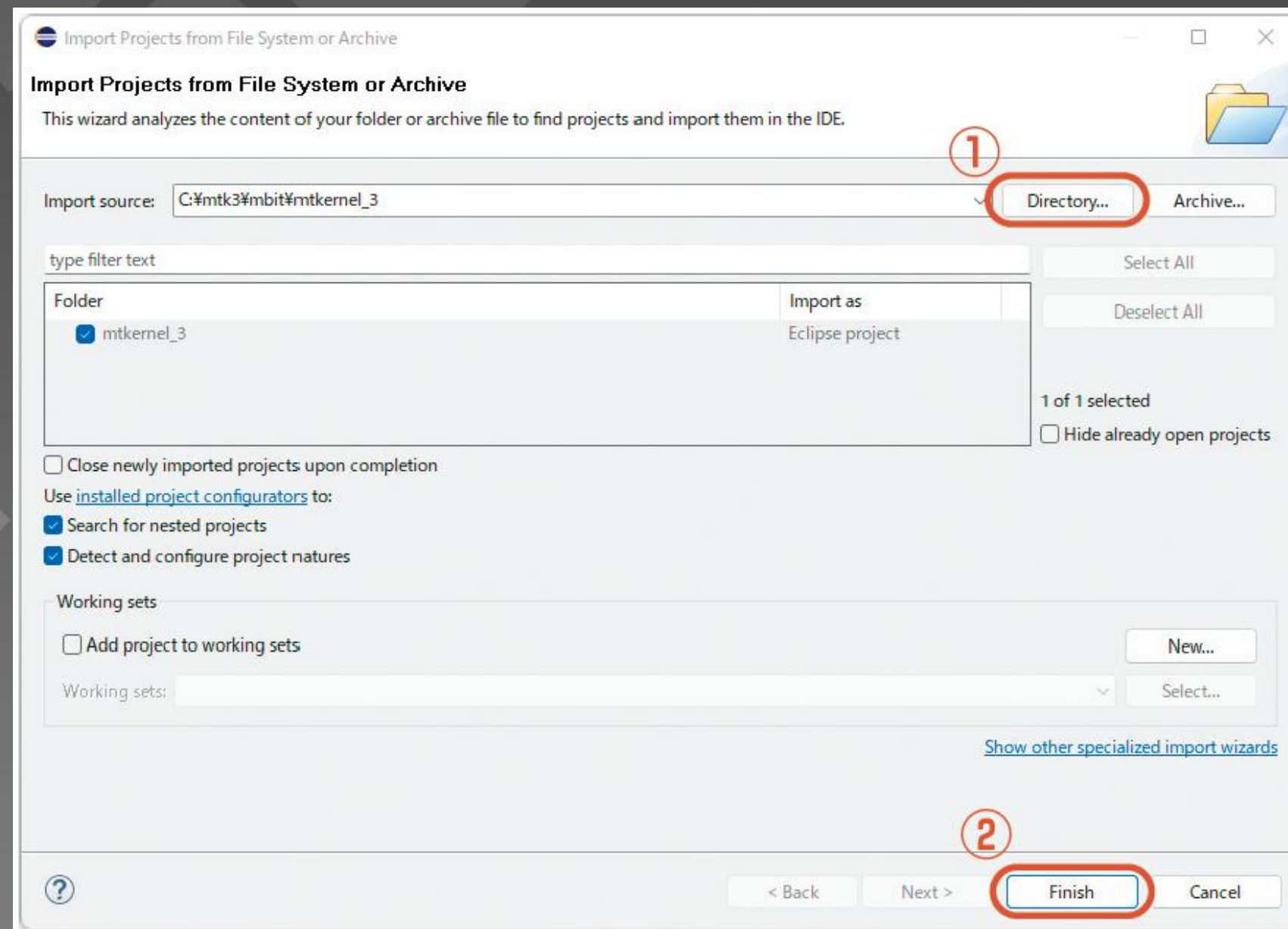
Eclipseを使ったプログラム開発の実際

- (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



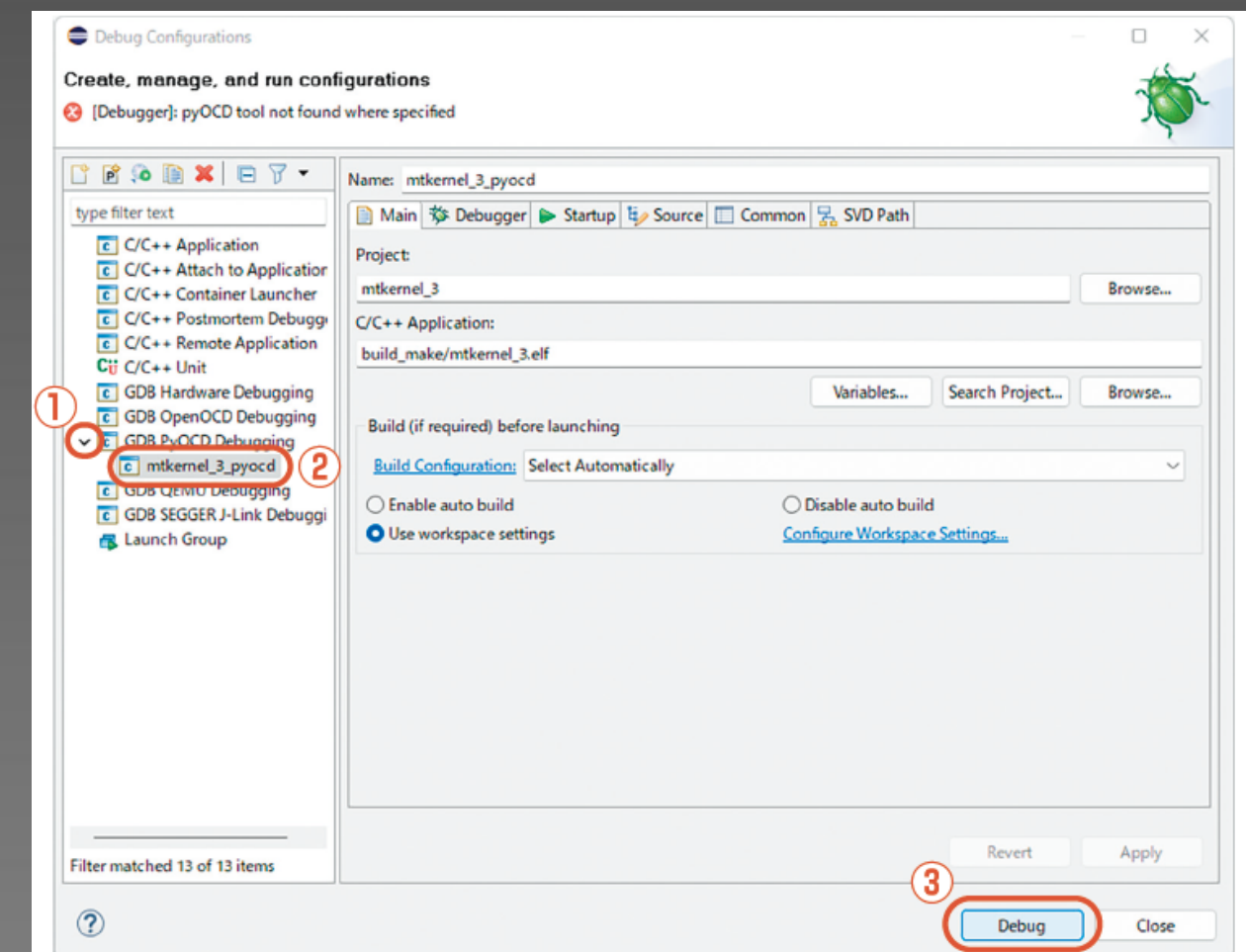
Eclipseを使ったプログラム開発の実際

■ [Step-2] μ T-Kernel 3.0のインポートとコンパイル



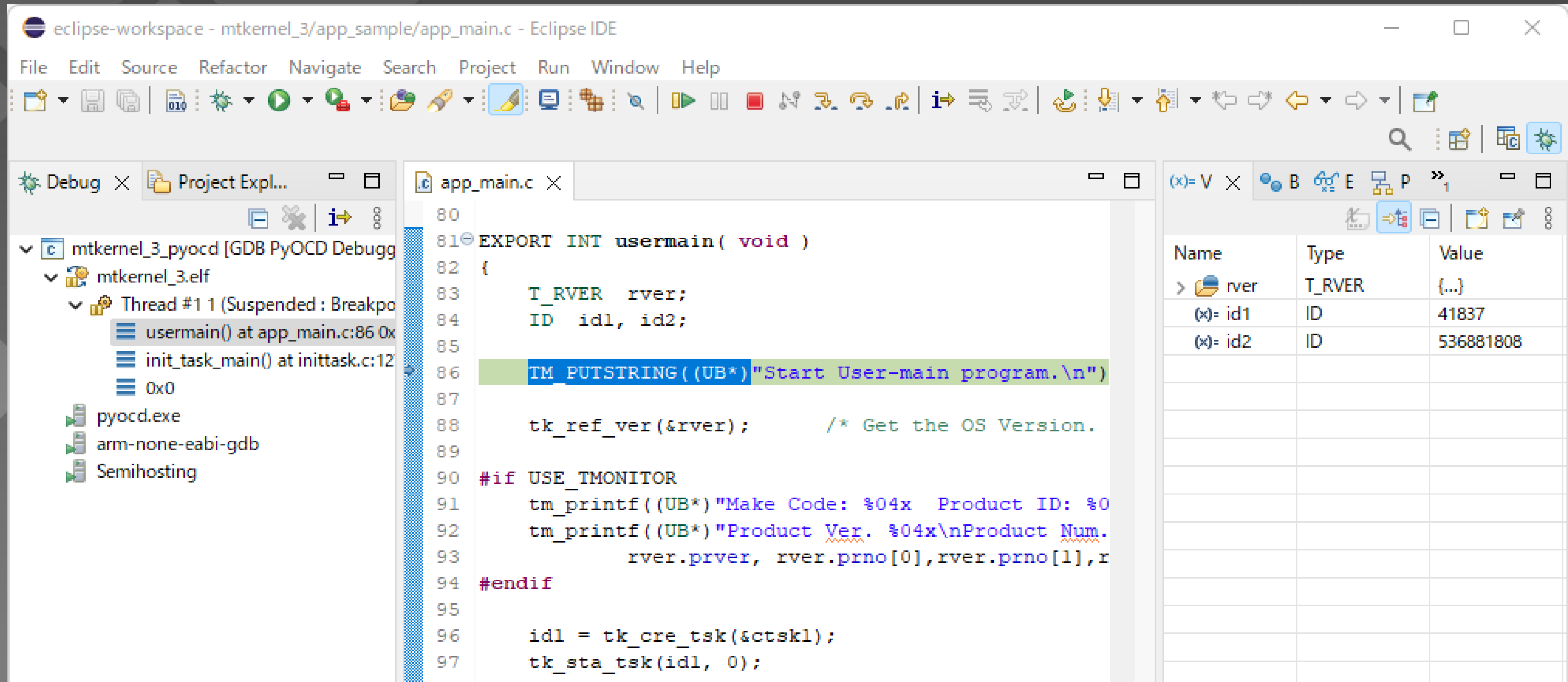
Eclipseを使ったプログラム開発の実際

- [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



The screenshot shows the Eclipse IDE interface with the following components:

- Project Explorer:** Shows the project structure with 'mtkernel_3_pyocd' and 'mtkernel_3.elf'.
- Source Editor:** Displays the C code for 'app_main.c'. The function 'usermain()' is visible, starting with a breakpoint at line 81.
- Debug Console:** Shows the output of the program: 'Start User-main program.\n'.
- Variables View:** Displays the current state of variables: 'rver' (T_RVER), 'id1' (ID, 41837), and 'id2' (ID, 536881808).

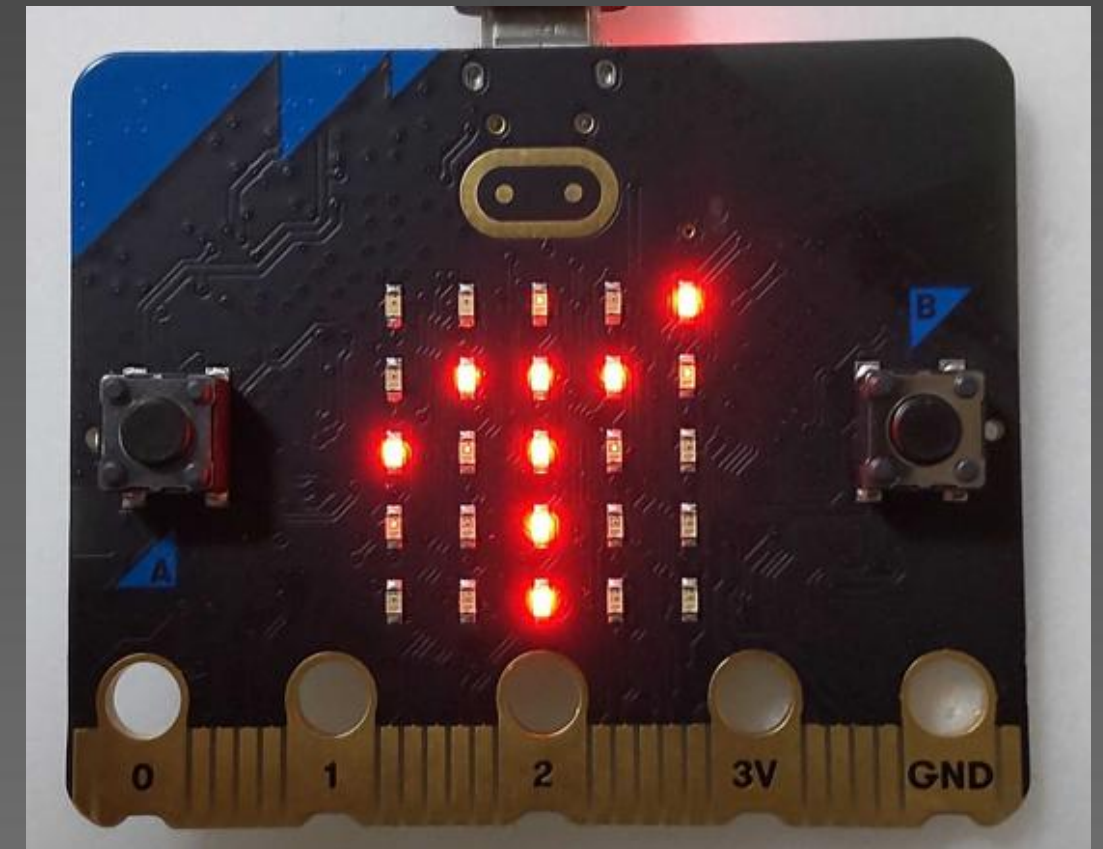
```
80
81 EXPORT INT usermain( void )
82 {
83     T_RVER rver;
84     ID id1, id2;
85
86     TM_PUTSTRING((UB*)"Start User-main program.\n")
87
88     tk_ref_ver(&rver); /* Get the OS Version.
89
90 #if USE_TMONITOR
91     tm_printf((UB*)"Make Code: %04x Product ID: %0
92     tm_printf((UB*)"Product Ver. %04x\nProduct Num.
93         rver.prver, rver.prno[0], rver.prno[1], r
94 #endif
95
96     id1 = tk_cre_tsk(&ctsk1);
97     tk_sta_tsk(id1, 0);
98 }
```



周辺デバイスを使う

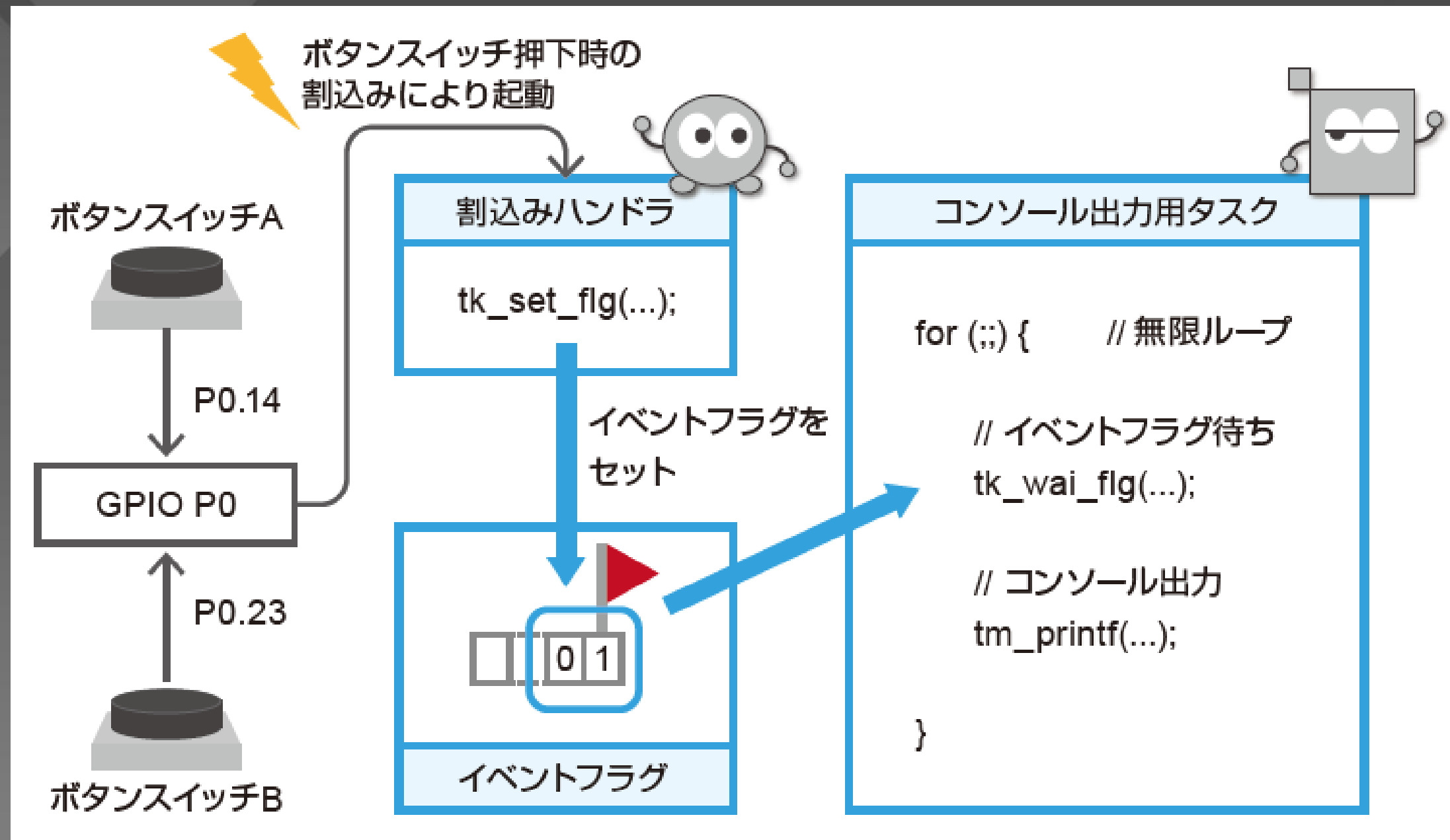
周辺デバイスを使う

- (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変換とジョイスティック
- (5)I2Cと加速度センサー
- (6)別のmicro:bitとシリアル接続



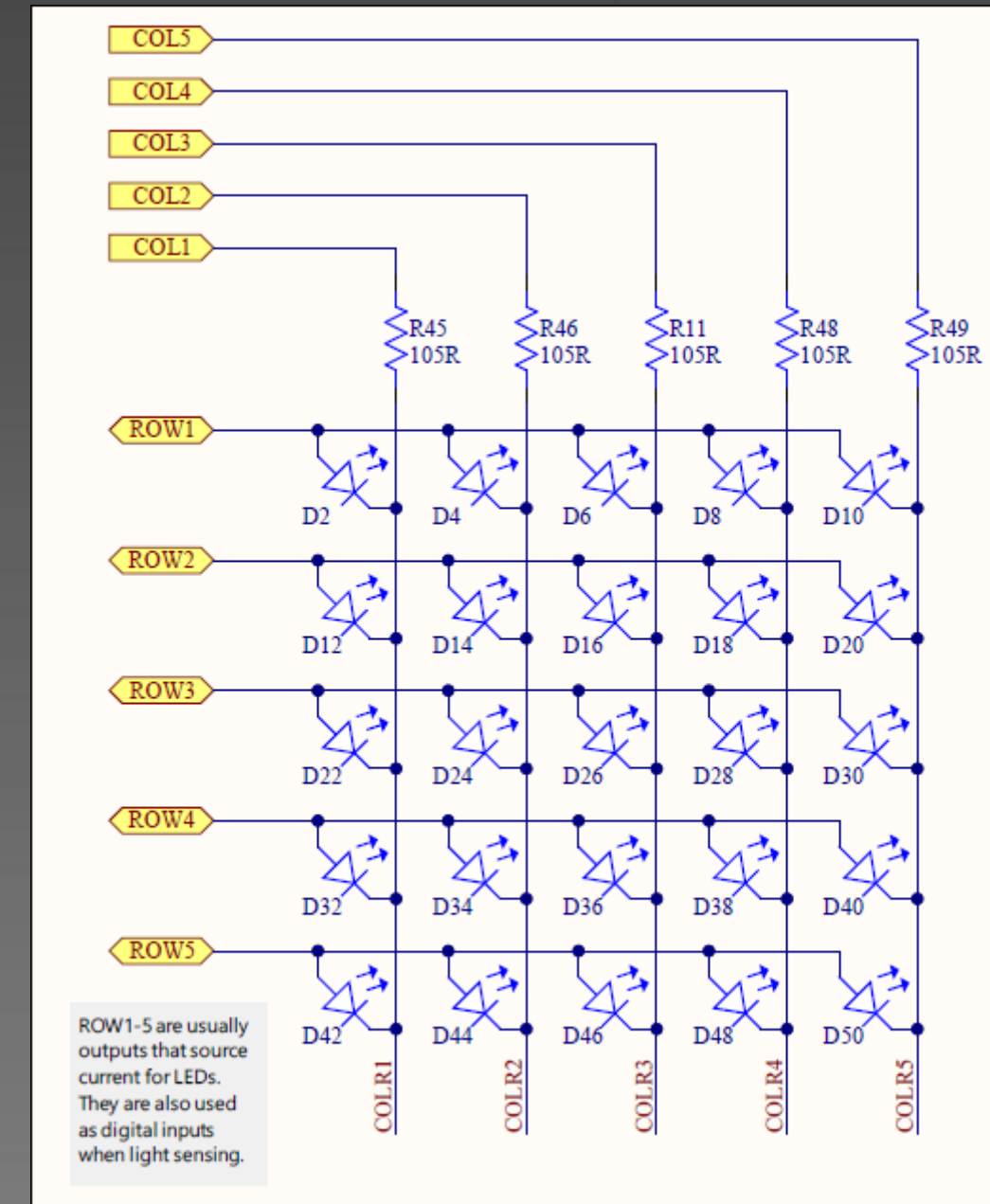
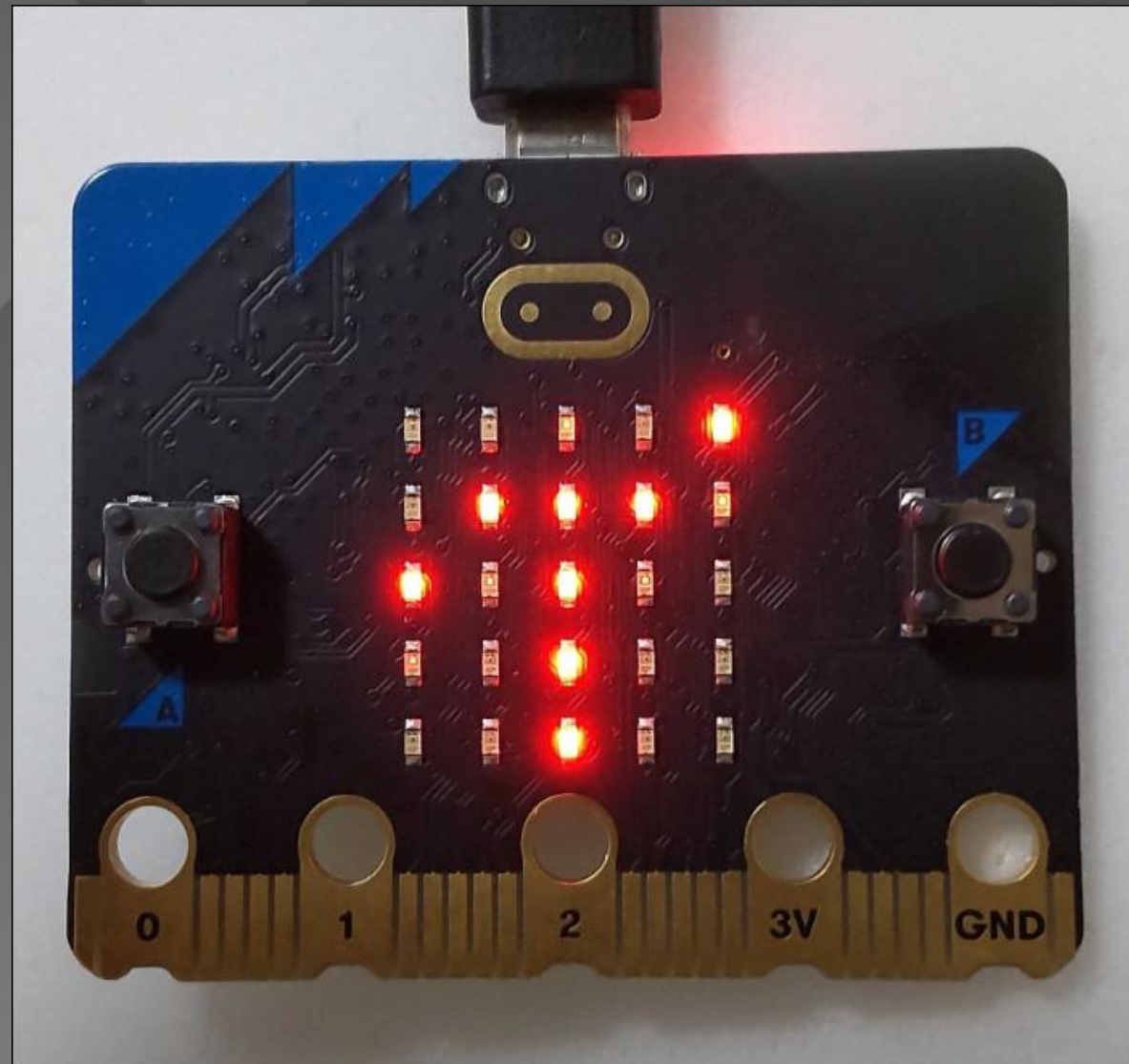
周辺デバイスを使う

■ (1)GPIOとボタンスイッチ割込み



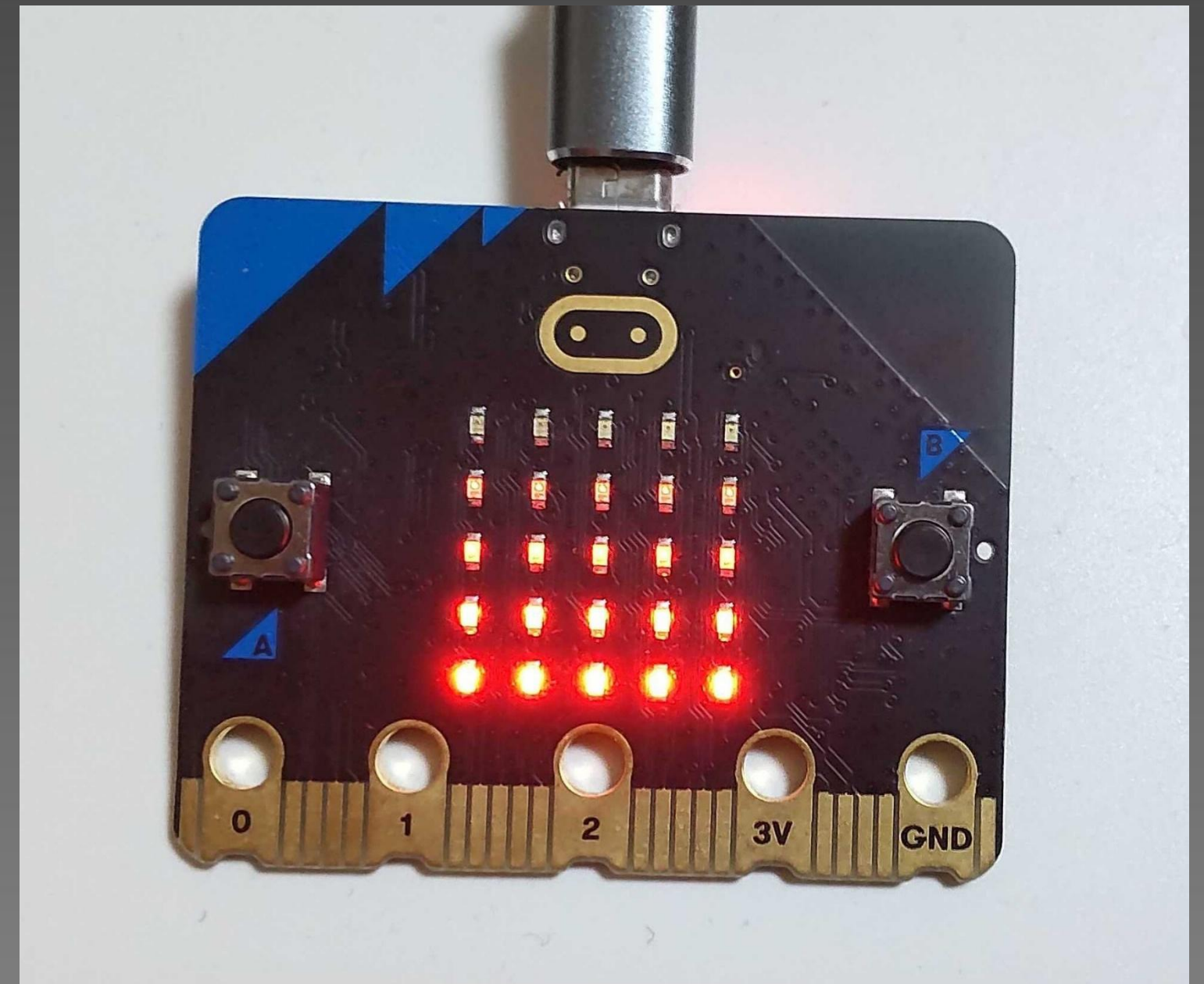
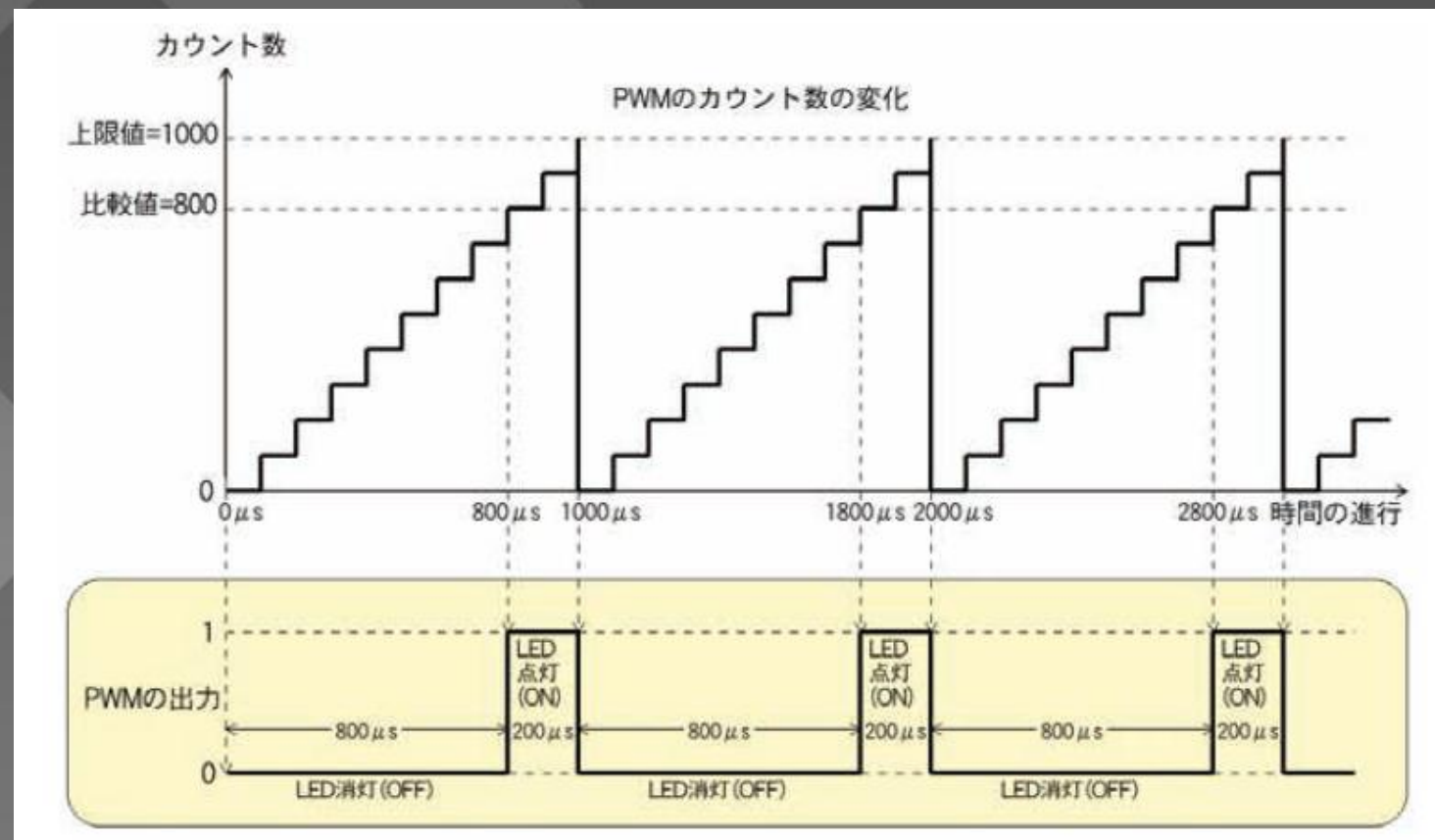
周辺デバイスを使う

■ (2)LEDマトリックスに「イ」を表示



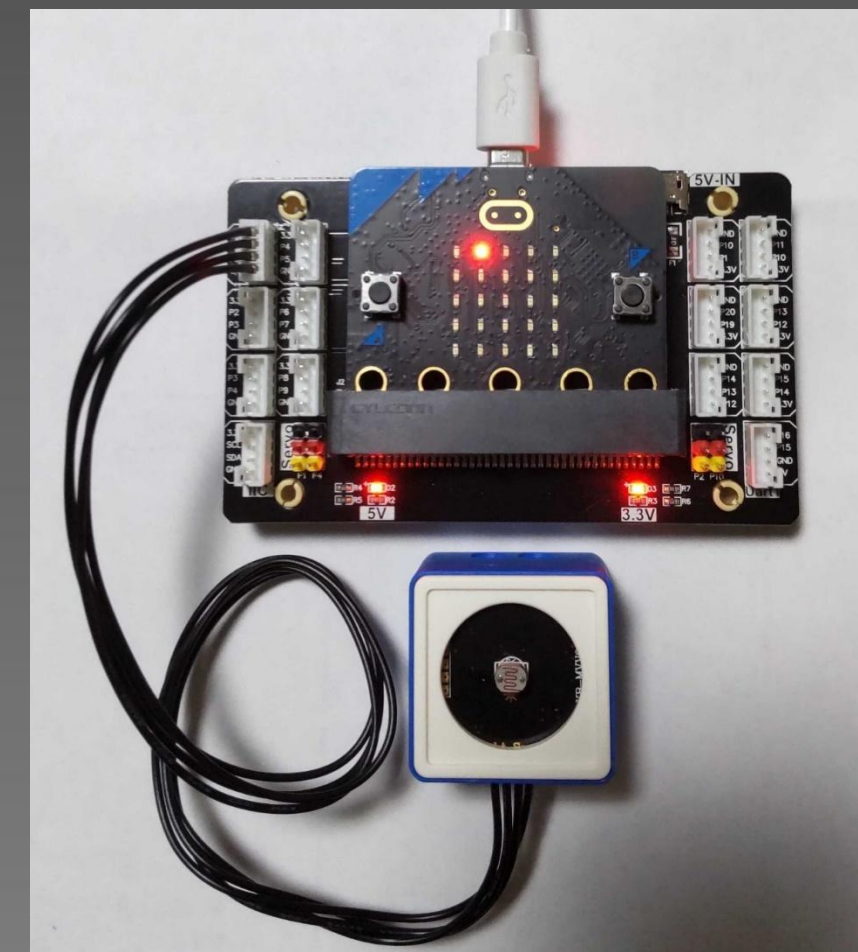
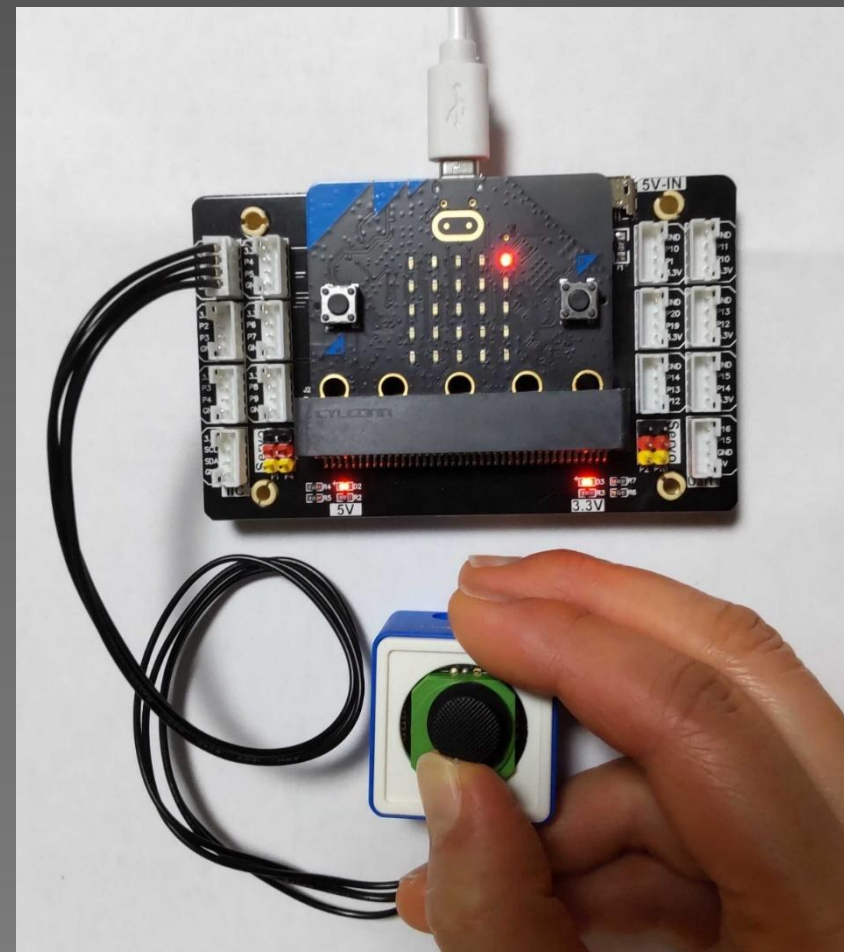
周辺デバイスを使う

■ (3)PWMによる音階再生とLEDの調光制御



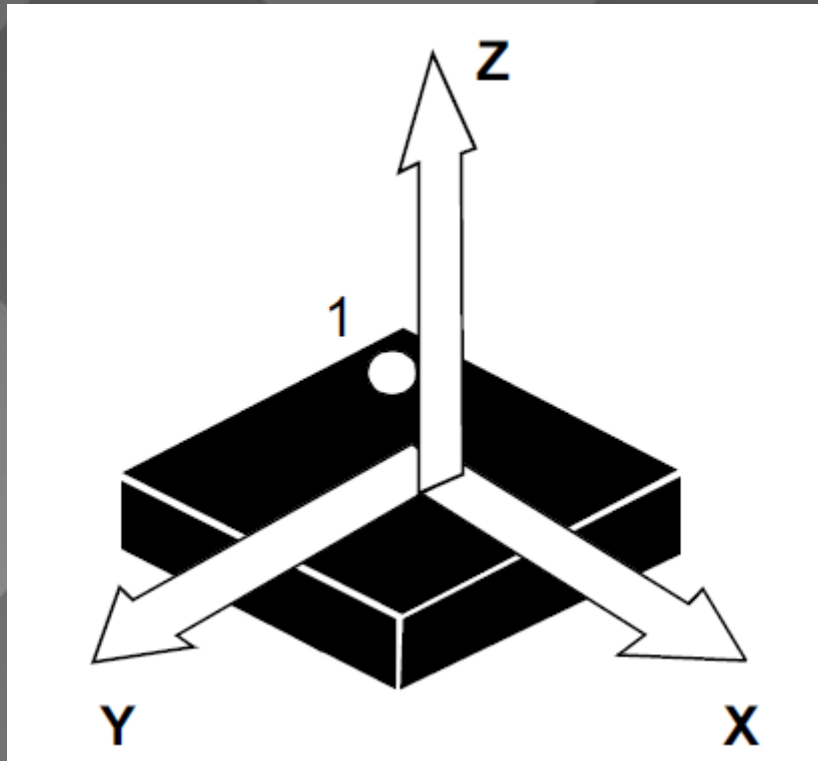
周辺デバイスを使う

- (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

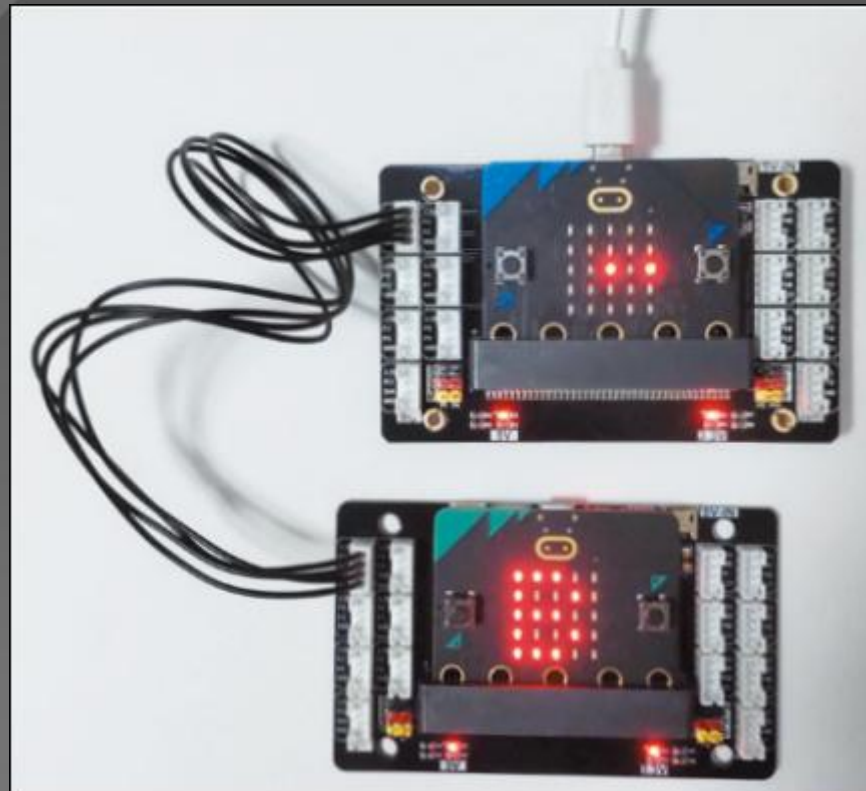
← ボードが水平 (micro:bit の文字が上)

← 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

- (6)別のmicro:bitとシリアル接続
 - もう1台のmicro:bitではMakeCodeを使うことも可能

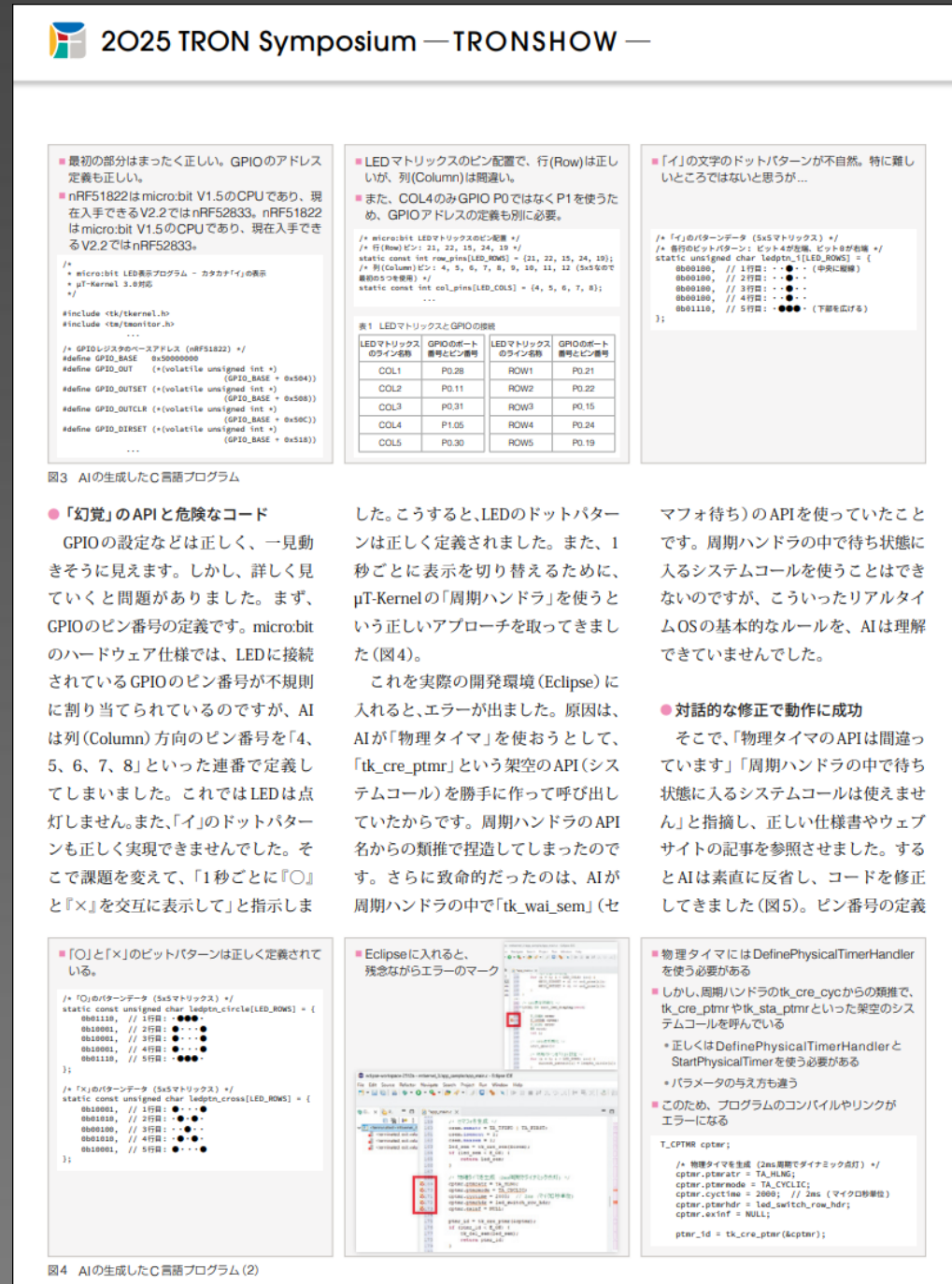


周辺デバイスを使う

- 紹介したサンプルプログラムはTRONWAREの連載記事で掲載、Web版もあり
- (1)GPIOとボタンスイッチ割込み (VOL.201)
 - <https://www.t-engine4u.com/info/mbit/5.html>, <https://www.t-engine4u.com/info/mbit/6.html>
- (2)LEDマトリックスに「イ」を表示 (VOL.202)
 - <https://www.t-engine4u.com/info/mbit/7.html>
- (3)PWMによる音階再生とLEDの調光制御 (VOL.205)
 - <https://www.t-engine4u.com/info/mbit/10.html>
- (4)A/D変換とジョイスティック (VOL.206)
 - <https://www.t-engine4u.com/info/mbit/11.html>
- (5)I2Cと加速度センサー (VOL.207)
 - <https://www.t-engine4u.com/info/mbit/12.html>
- (6)別のmicro:bitとシリアル接続 (VOL.208)
 - <https://www.t-engine4u.com/info/mbit/13.html>

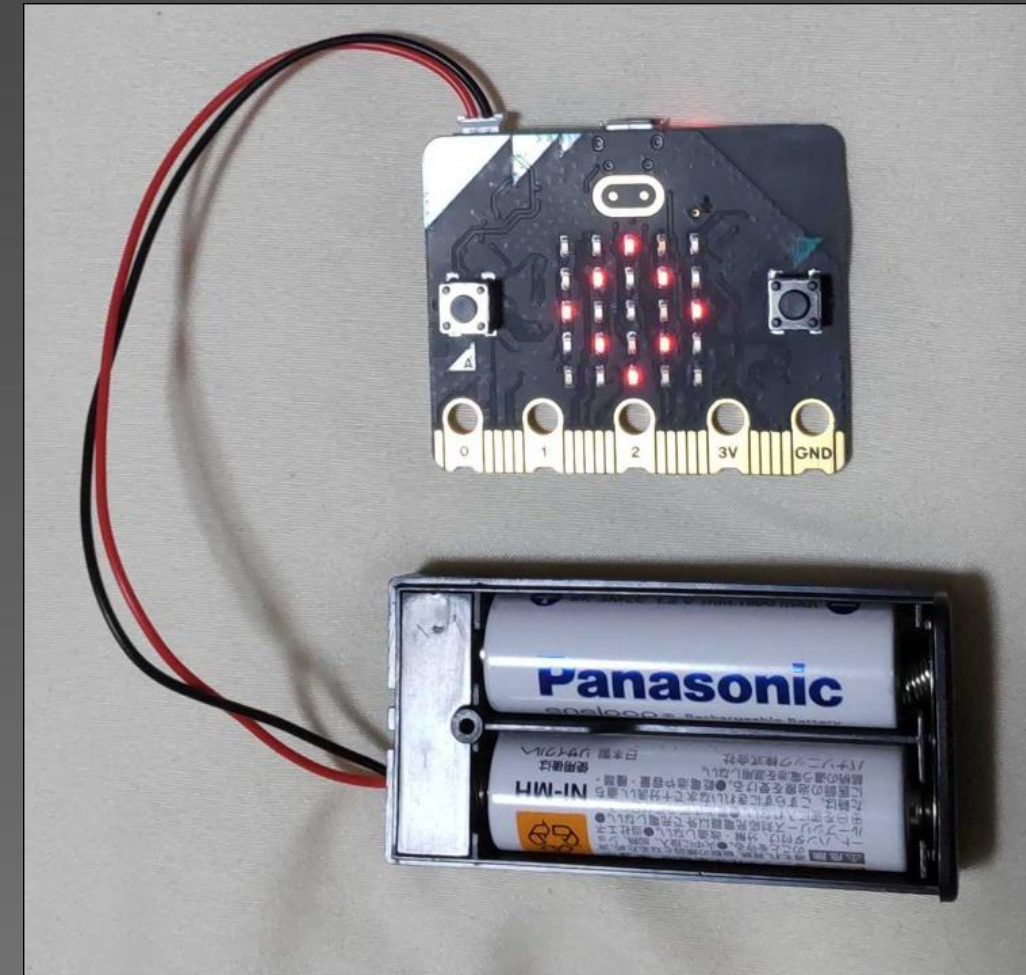
micro:bit + μ T-Kernelのプログラム開発方法：番外編

- 生成AIに適切なプロンプトを与えると、ある程度のプログラムはできる
- プロンプト例
 - micro:bitで動く μ T-Kernel 3.0を使って、LEDマトリックスに「イ」を表示するC言語のプログラムを書いてください。
- TRONWARE VOL.217 (最新号) で紹介



コンテスト応募とAI活用のヒント

- micro:bitの特性を活かす
 - ボードのコンパクト性
 - 電池駆動の携帯性
 - 豊富なセンサー類
- たとえば、以下を組み合わせたミニ楽器
 - (3) PWMによる音階再生
 - (5) 加速度センサー
- AIの活用例
 - micro:bitでも実行可能な超コンパクトなAI機能、フレームワーク、機械学習ソフトウェア等の利用
 - クラウド上のAIと連携
 - 開発環境や開発ツールとしてAIを活用
- とはいえ、上記にとらわれない広範囲なアイデアを募集



Q and A

- 参加者の皆様からのご質問



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