Confidential

# VS CodeによるZephyr開発 [Part3] AIアプリ評価編

## Nuvoton NuMicro® M55M1シリーズ

11/09/2026

Since 1844 Marubun

デバイス・ソリューション企画部

© 2026 MARUBUN CORPORATION

# ご利用上の注意および免責事項

丸文株式会社（以下、当社）より資料を入手されたお客様は、下記の使用上の注意をご一読のうえご使用ください。
お客様は本資料のご使用にあたり、下記の内容に従うことに合意したものとします。

- 本資料中に記載の技術、アプリケーション、その他設計に関する助言及び情報、並びに本資料に関して別途ご提供する各種サービスは、製品を組み込んだアプリケーションの開発者に役立つことを目的としてご提供するものです。
- 本資料は非売品であり、許可無く転売することや無断複製することを禁じます。
- 本資料は予告なく変更することがあります。
- 本資料は作成時の情報にもとづき作成しておりますが、もととなる情報が更新された場合でも本資料には反映されていない場合があります。
- 本資料の内容とメーカ資料の内容に相違がある場合は、メーカ資料の内容が優先されます。
- 本資料は製品・ツールを利用する際の補助的なものとして作成しています。製品・ツールをご使用になる場合はメーカ資料もあわせてご確認ください。
- 本資料はお客様に製品・ツールをご使用いただくための参考資料であり、本資料で取り扱っている内容（回路、技術、プログラム、測定データ、数値等）に関しては参考情報となりますので、貴社にて十分な検証を行ったうえ、ご使用ください。
- 本資料で取り扱っている内容（回路、技術、プログラム、測定データ、数値等）に関して運用した結果の損害、第三者の知的財産権、その他権利に関する侵害に関し、当社は責任を負いません。
- 本資料を非居住者に提供する場合は、外為法および国内外の輸出関連法令等を遵守し、必要な手続きをおとりください。但し、居住者、非居住者のいずれの場合であっても、本資料を大量破壊兵器等の開発等に使用したり、そのおそれがある第三者には提供しないでください。また、本資料をその他軍事用途の目的に使用する非居住者、あるいは、そのおそれがある非居住者にも提供しないでください。なお、本資料の提供に関するご相談等は、弊社営業窓口までお問い合わせください。
- 本資料の作成には万全を期していますが、万一誤り、記載漏れなどお気づきの点がありましたら、当社担当者までご一報いただけますでしょうか。

2018年2月 丸文株式会社

Marubun Since 1844

© 2026 MARUBUN CORPORATION

Confidential

# アジェンダ

- 導入&セットアップ
- プロジェクト作成
- ビルド&Flash書き込み
- 動作確認
- デバッグ環境

Since 1844 Marubun

© 2026 MARUBUN CORPORATION

# はじめに

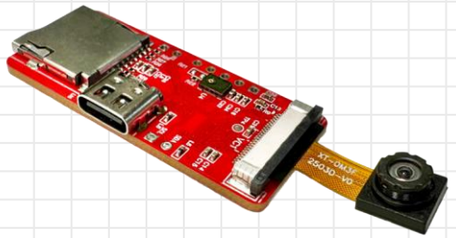
Zephyrは、リソースが制限されたMCU向けに設計された、オープンソースの小型で軽量なRTOSです。Nuvoton Technology（以下Nuvoton）のNuMicro®シリーズMCUにおいても、Zephyrのサポートが進められています。

本資料では、Zephyr上でYOLOX-Nanoを用いたサンプルコード「物体・ジェスチャー検出」の評価手順をご紹介します。評価ボードは「NuMaker-GestureAI-M55M1」および「NuMaker-X-M55M1D」を対象とします。

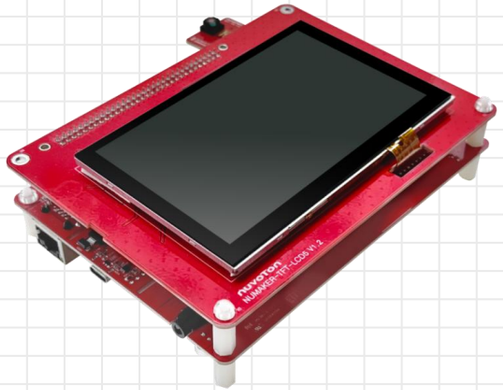
本資料の手順は、 GitHub上のM55M1用リポジトリに記載されている「ZephyrSamples → SampleCode → ObjectDetection_Zephyr_yoloxn」にある、「README.md」の手順をベースに構成しています。

## 動作環境

- Visual Studio Code: v1.136.1
- Zephyr IDE Extension Pack: v0.0.4
- Zephyr SDK: v1.0.1
- Zephyr OS: v4.4.0
- C Compiler: GNU 14.3.0



NuMaker-GestureAI-M55M1



NuMaker-X-M55M1D

© 2026 MARUBUN CORPORATION

Confidential

# 導入&セットアップ

© 2026 MARUBUN CORPORATION

# 導入&セットアップ

## 参照資料

導入編となる下記資料をご参照ください。

- VS CodeによるZephyr開発 [Part1] 導入&セットアップ編
- VS CodeによるZephyr開発 [Part2] ビルド&デバッグ環境編

**※以下、「VS CodeによるZephyr開発 [Part1] 導入&セットアップ編」の手順を実施したことを前提とします。**

## 環境確認

VS Codeを起動して以下を確認します。

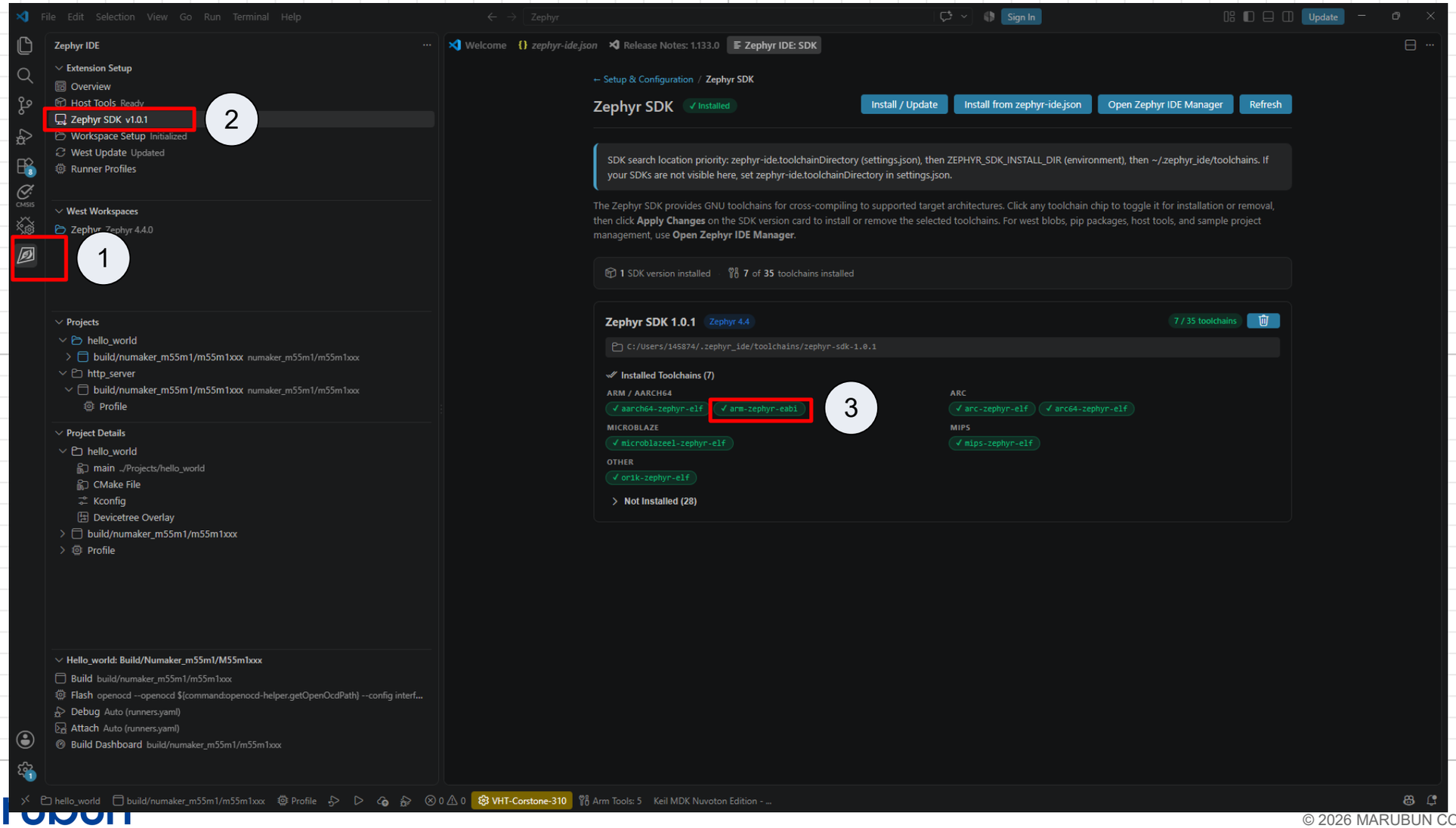
- ①「Zephyr IDE」をクリックします。
- ②「Zephyr SDK v1.0.1」をクリックします。
- ③「arm-zephyr-eabi」に緑のチェックが入っていることを確認します。

Since 1844 Marubun

© 2026 MARUBUN CORPORATION

# 導入&セットアップ

## Zephyr SDK



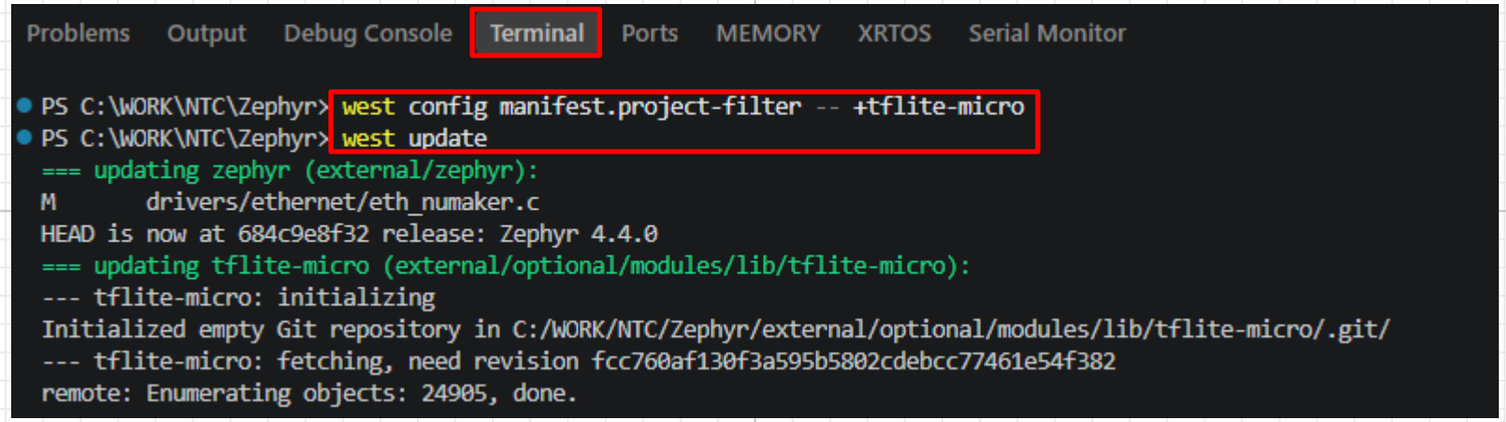

© 2026 MARUBUN CORPORATION

# 導入&セットアップ

## tflite-micro外部モジュールの追加

VS Codeの上部から「Terminal → New Terminal」をクリックして、以下のコマンドを入力します。

- cd C:¥WORK¥NTC¥Zephyr
- west config manifest.project-filter -- +tflite-micro
- west update




© 2026 MARUBUN CORPORATION

Confidential

# プロジェクト作成

© 2026 MARUBUN CORPORATION

Confidential

# プロジェクト作成

## サンプルコードのダウンロード

Terminalで以下のGitコマンドを実行します。（本書作成時点の最新タグであるv1.3.0を指定します。）

- cd C:¥WORK¥NTC
- git clone https://github.com/OpenNuvoton/ML_M55M1_SampleCode.git
- cd ML_M55M1_SampleCode
- git checkout v1.3.0

## Project Folderとは？

サンプルコードを配置し、プログラムの編集やビルドを行うフォルダを指します。

## サンプルコードのコピー

以下のようにフォルダをコピーします。（ProjectsがProject Folderです。）

- コピー元　C:¥WORK¥NTC¥ML_M55M1_SampleCode¥ZephyrSamples¥SampleCode¥ObjectDetection_Zephyr_yoloxn
- コピー先　C:¥WORK¥NTC¥Projects¥ObjectDetection_Zephyr_yoloxn

CMakeLists.txtが参照するフォルダもコピーします。

- コピー元　C:¥WORK¥NTC¥ML_M55M1_SampleCode¥ZephyrSamples¥ThirdParty
- コピー先　C:¥WORK¥NTC¥ThirdParty

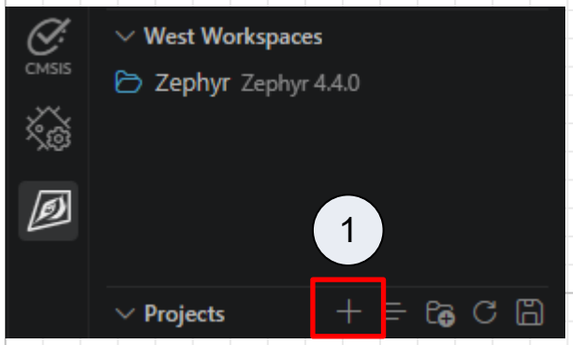
Marubun Since 1844

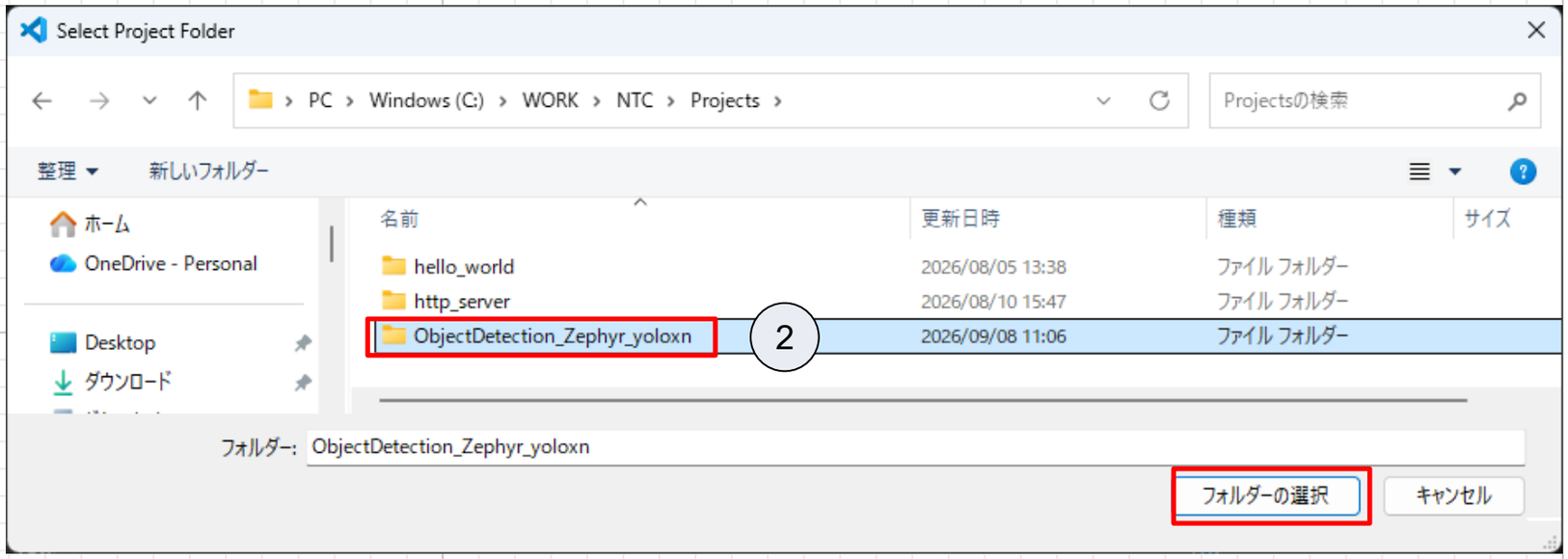
© 2026 MARUBUN CORPORATION

# プロジェクト作成

## サンプルコードのプロジェクト作成

- ① 「Projects」から「Add Project」をクリックします。
- ② フォルダ選択のポップアップでは、上記のコピー先のフォルダを選択します。



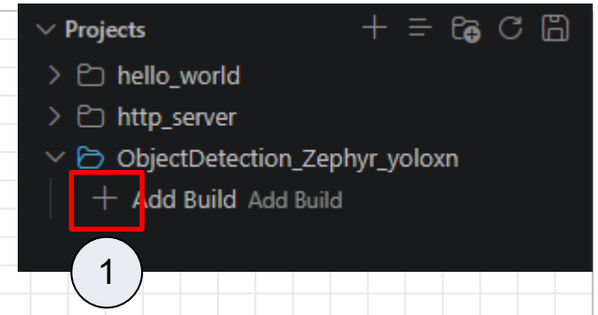

© 2026 MARUBUN CORPORATION

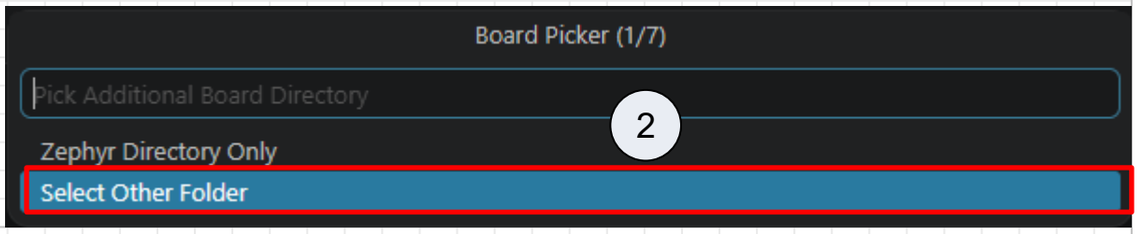
Confidential

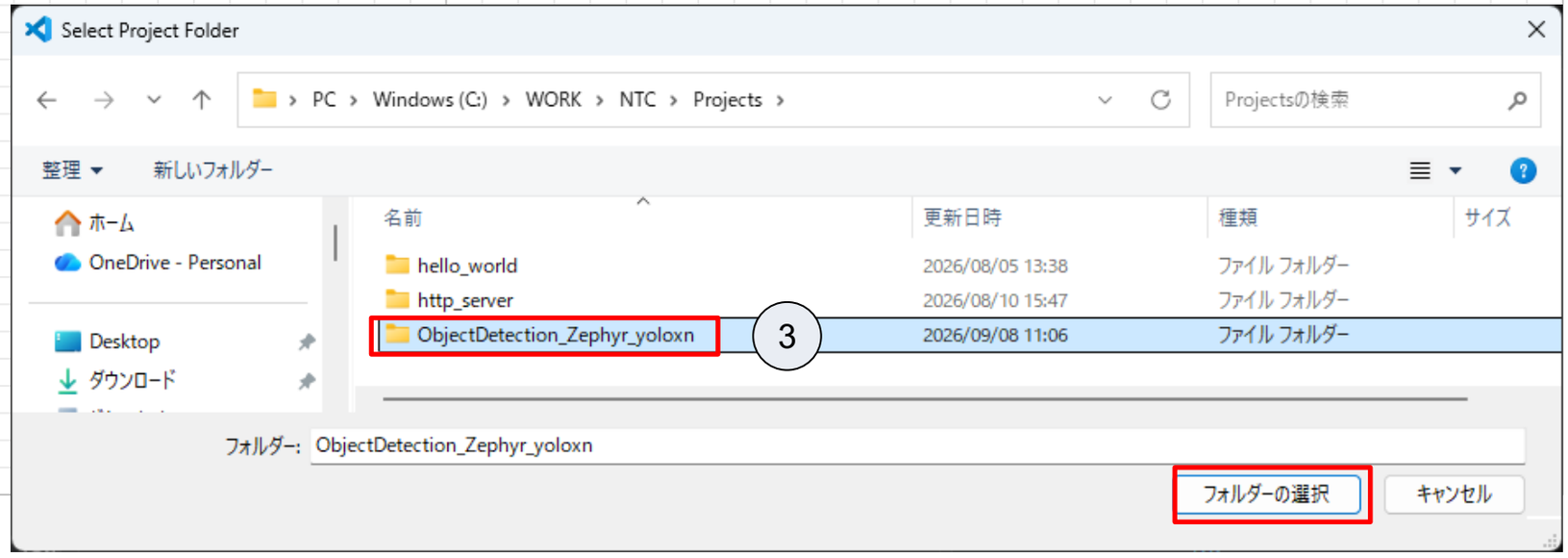
# プロジェクト作成

## ビルド構成の作成

- ① 追加されたObjectDetection_Zephyr_yoloxnの「Add Build」をクリックします。
- ② 画面上部の選択画面では、「Select Other Folder」を選択します。
- ③ フォルダ選択のポップアップでは、先ほどと同じフォルダを選択します。



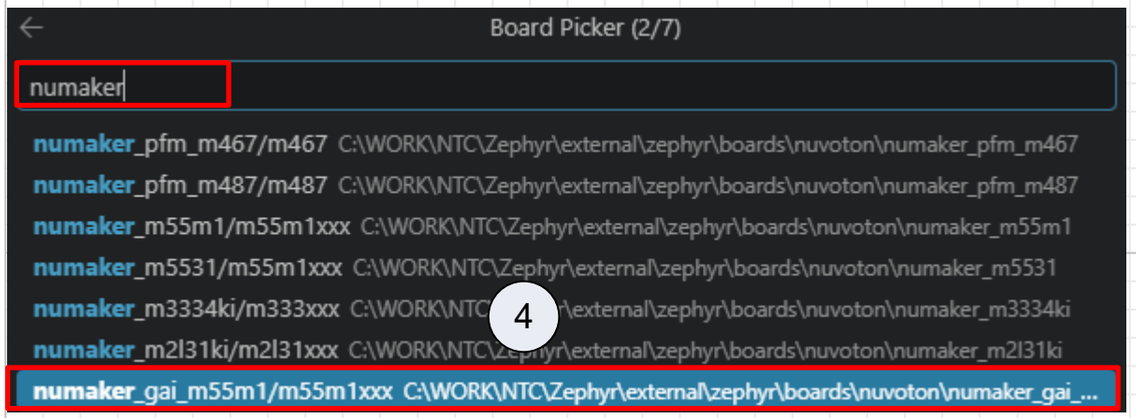




© 2026 MARUBUN CORPORATION

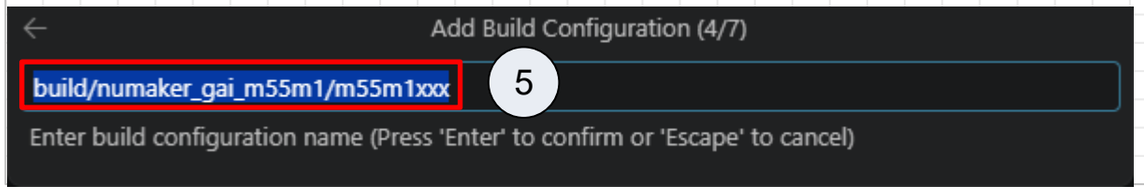
# プロジェクト作成

- ④ 「Board Picker」に numaker と入力して、使用するボードに合わせて「numaker_gai_m55m1/m55m1xxx」または「numaker_m55m1/m55m1xxx」を選択して、Enterキーを押します。
- ⑤ そのままEnterキーを押します。

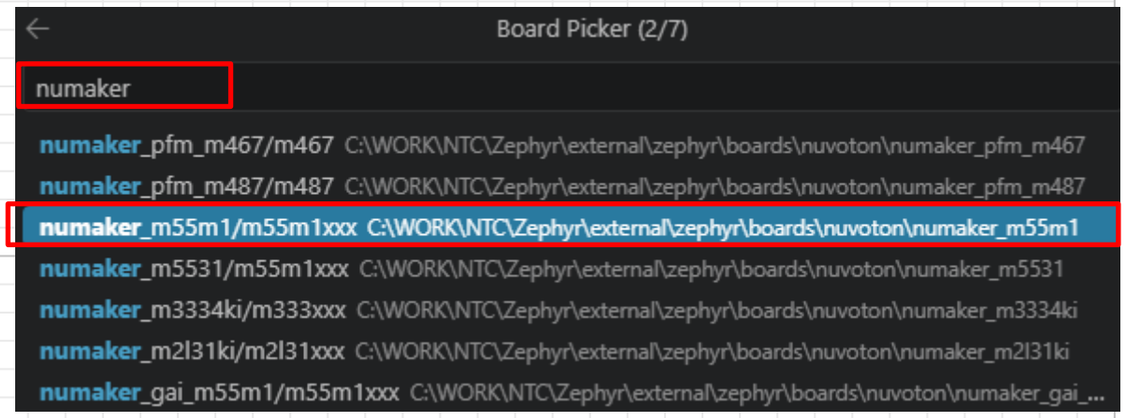
NuMaker-GestureAI-M55M1



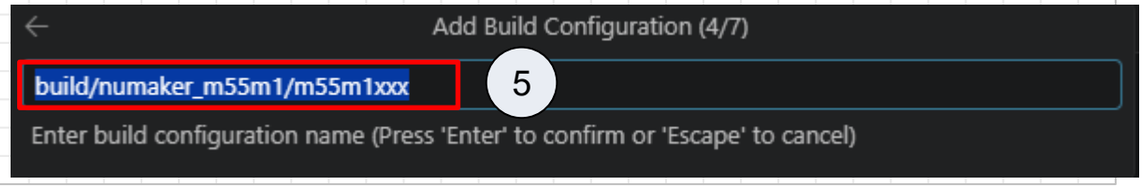

（Add Build Configuration (3/7) は表示なし）




NuMaker-X-M55M1D



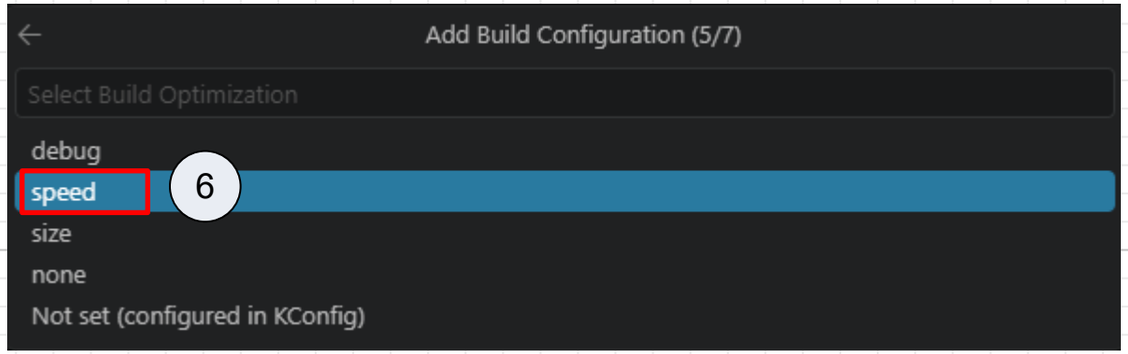

（Add Build Configuration (3/7) は表示なし）

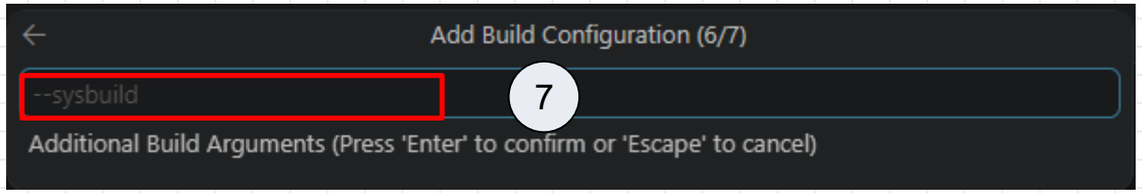


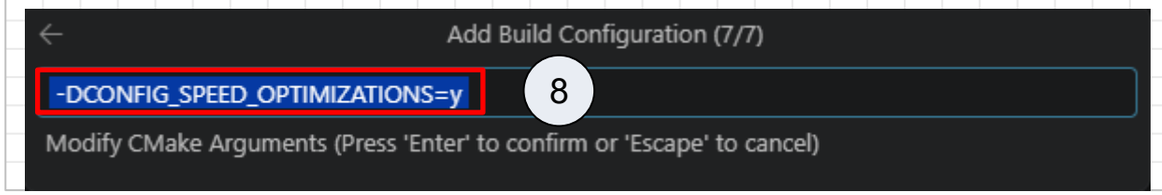

© 2026 MARUBUN CORPORATION

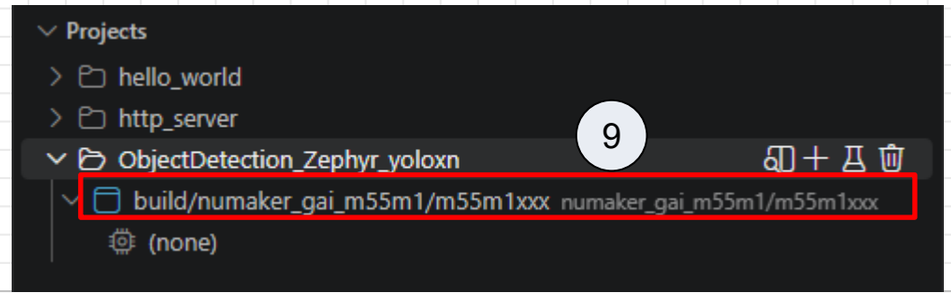
# プロジェクト作成

- ⑥ 「speed」を選んでEnterキーを押します。
- ⑦ そのままEnterキーを押します。
- ⑧ そのままEnterキーを押します。
- ⑨ プロジェクトが作成されたことを確認します。（NuMaker-GestureAI-M55M1での表示例）



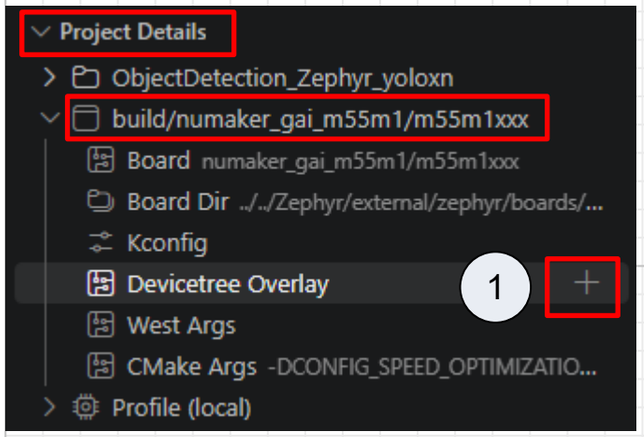



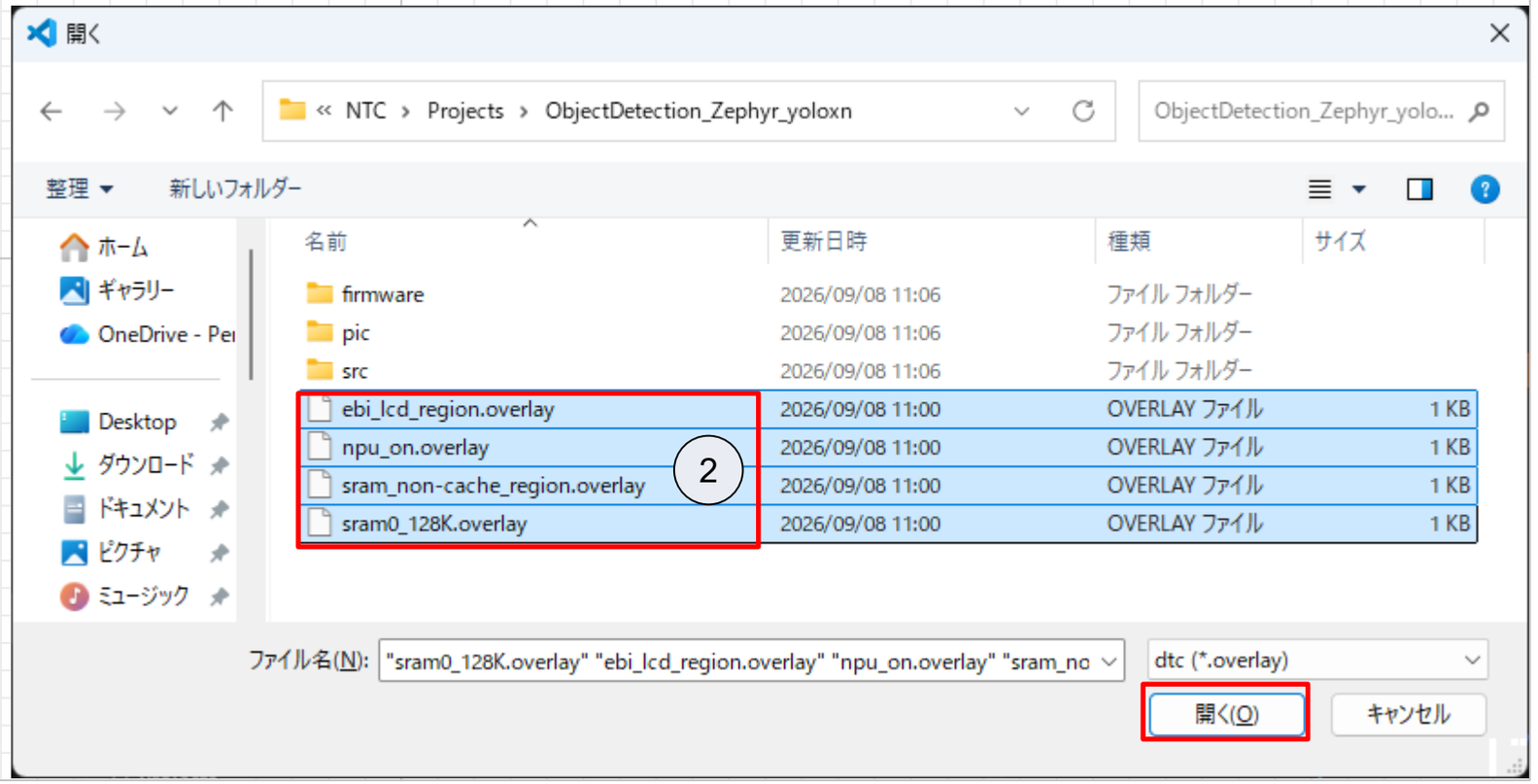

© 2026 MARUBUN CORPORATION

# プロジェクト作成

## Devicetree Overlayの追加

- ① 「Project Details」の「build/numaker_xx」側の「Devicetree Overlay」をクリックします。
- ② ファイル選択のポップアップでは、「ObjectDetection_Zephyr_yoloxn」の中にある4つのOVERLAYファイルを選択します。

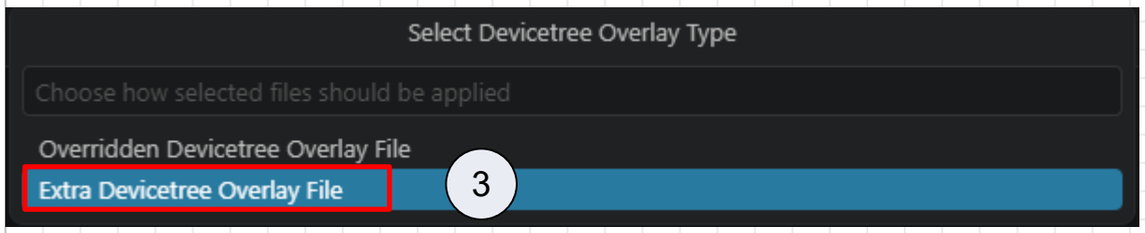


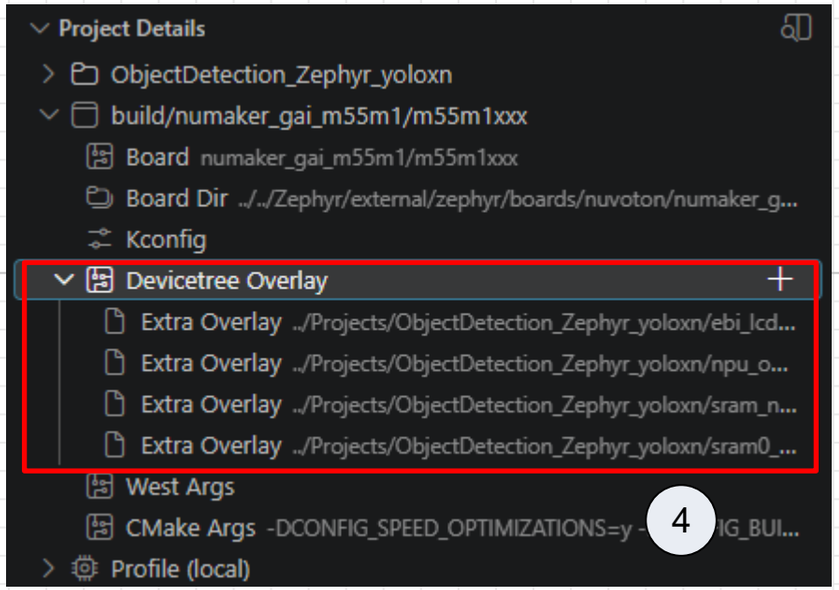




© 2026 MARUBUN CORPORATION

# プロジェクト作成

- ③ 「Extra Devicetree Overlay File」を選んでEnterキーを押します。
- ④ 「Devicetree Overlay」に追加されたことを確認します。



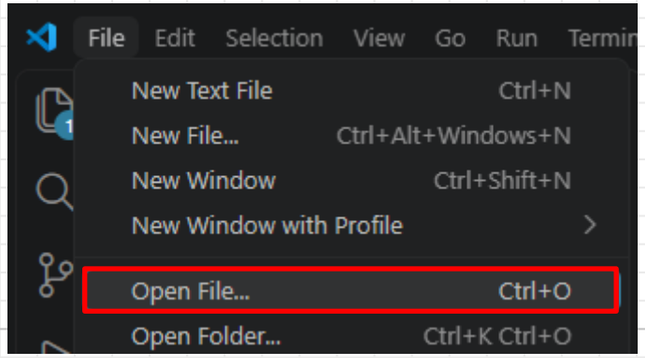

© 2026 MARUBUN CORPORATION

# プロジェクト作成

## prj.confの変更

File → Open File（またはCtrl+O）で「prj.conf」ファイルを開きます。

- C:¥WORK¥NTC¥Projects¥ObjectDetection_Zephyr_yoloxn¥prj.conf




NuMaker-GestureAI-M55M1の場合は、CONFIG_CMOS_GC0308_SENSOR=yの前の # を削除します。

```
47   # Enable GC0308 image sensor for NuGestureAI-M55M1 board
48   #CONFIG_CMOS_GC0308_SENSOR=y
```

```
47   # Enable GC0308 image sensor for NuGestureAI-M55M1 board
48   CONFIG_CMOS_GC0308_SENSOR=y
```

© 2026 MARUBUN CORPORATION

# プロジェクト作成

アプリケーション実行時の推論結果は、Serial Monitorからのメッセージ出力の他、プレビュー機能としてUVCかLCDから画像出力ができます。

画像出力をする場合は、以下の変更を行います。

- NuMaker-GestureAI-M55M1の場合は、CONFIG_APP_OD_UVC_SHOW_IMAGEをn→yに変更します。
- NuMaker-X-M55M1Dの場合は、CONFIG_APP_OD_UVC_SHOW_IMAGE, CONFIG_APP_OD_LCD_SHOW_IMAGEのどちらかをn→yに変更します。

```
41  # Show result image over UVC
42  CONFIG_APP_OD_UVC_SHOW_IMAGE=n
43  # Show result image over LCD
44  CONFIG_APP_OD_LCD_SHOW_IMAGE=n
```



```
41  # Show result image over UVC
42  CONFIG_APP_OD_UVC_SHOW_IMAGE=y
43  # Show result image over LCD
44  CONFIG_APP_OD_LCD_SHOW_IMAGE=n
```

```
41  # Show result image over UVC
42  CONFIG_APP_OD_UVC_SHOW_IMAGE=n
43  # Show result image over LCD
44  CONFIG_APP_OD_LCD_SHOW_IMAGE=y
```

編集が終わったら、Ctrl+Sでファイルを保存します。

「README.md」も合わせてご確認ください。

© 2026 MARUBUN CORPORATION

Confidential

# ビルド&Flash書き込み

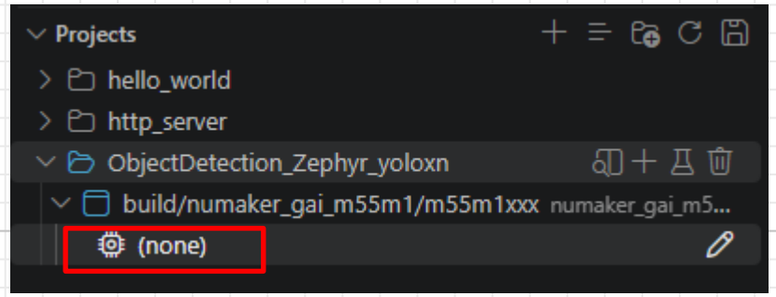
© 2026 MARUBUN CORPORATION

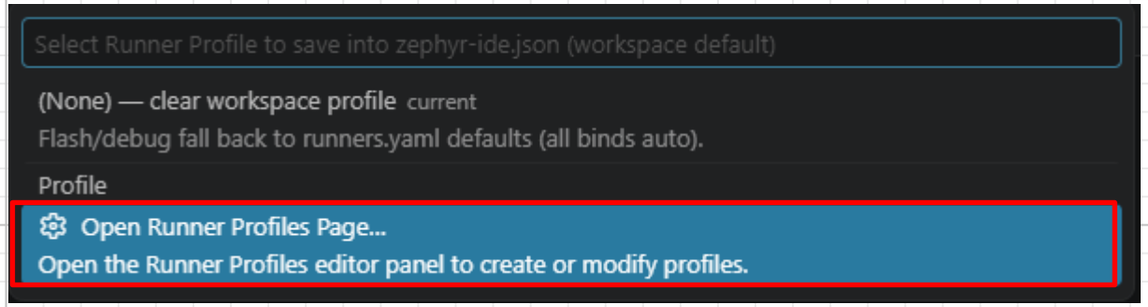
# Flash書き込み環境構築

**※以下、「VS CodeによるZephyr開発 [Part2] ビルド&デバッグ環境編」の「Flash書き込み」手順を実施したことを前提とします。**

## Runner Profile（Flash書き込み設定）の作成

「Projects」の(none) をクリックして、画面上部の選択画面では、「Open Runner Profiles Page...」を選択します。

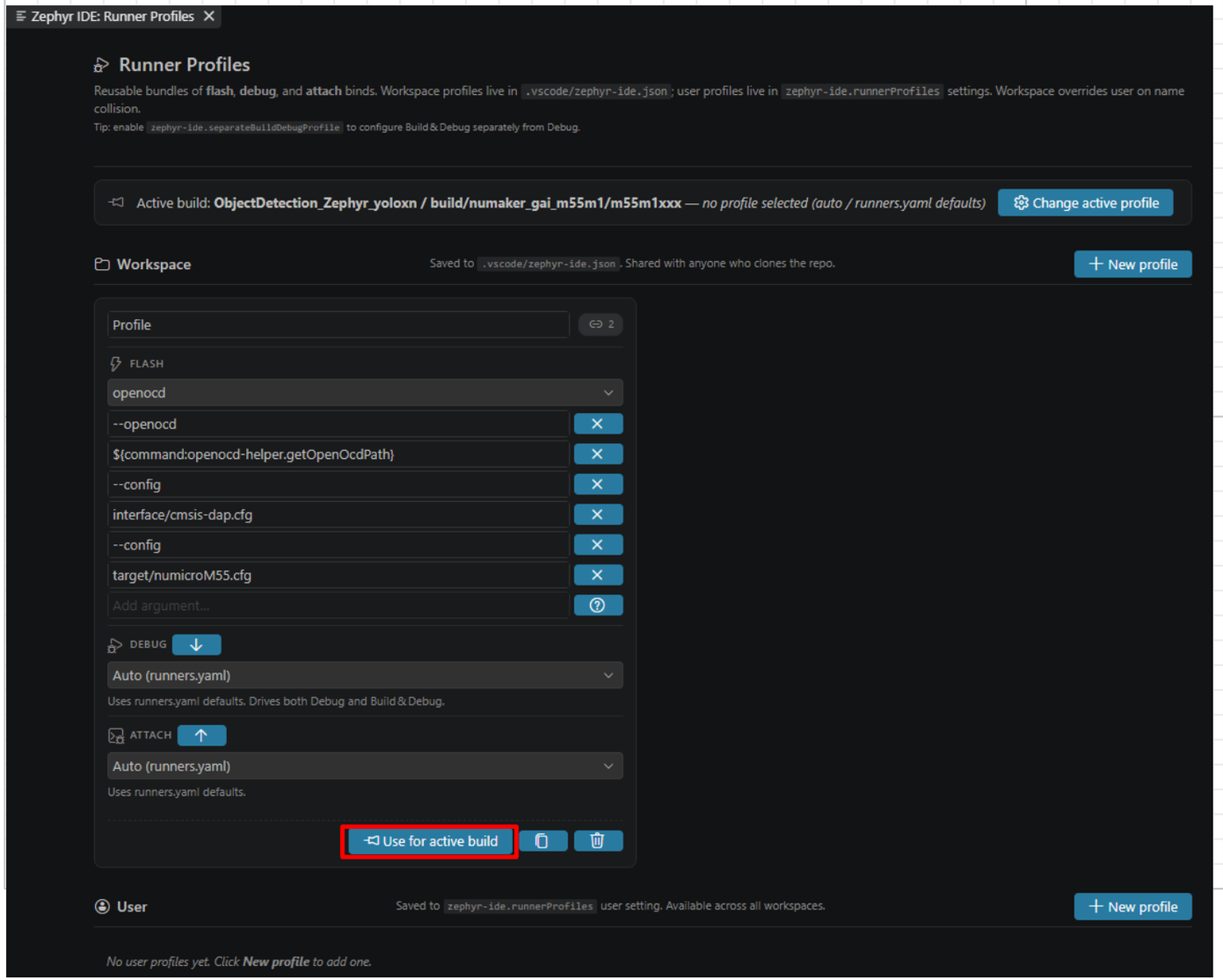


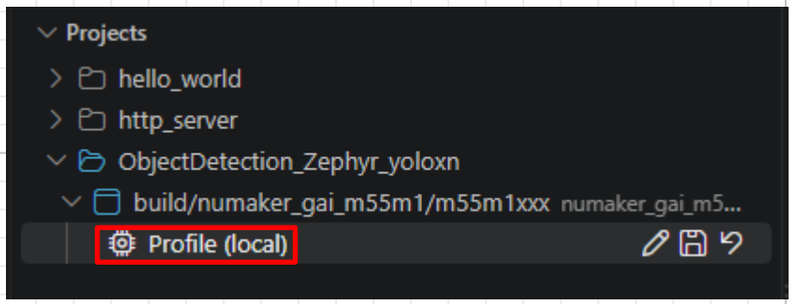




© 2026 MARUBUN CORPORATION

# Flash書き込み環境構築

既存の設定が読み込まれますので、「Use for active build」をクリックして本プロジェクトにも適用します。



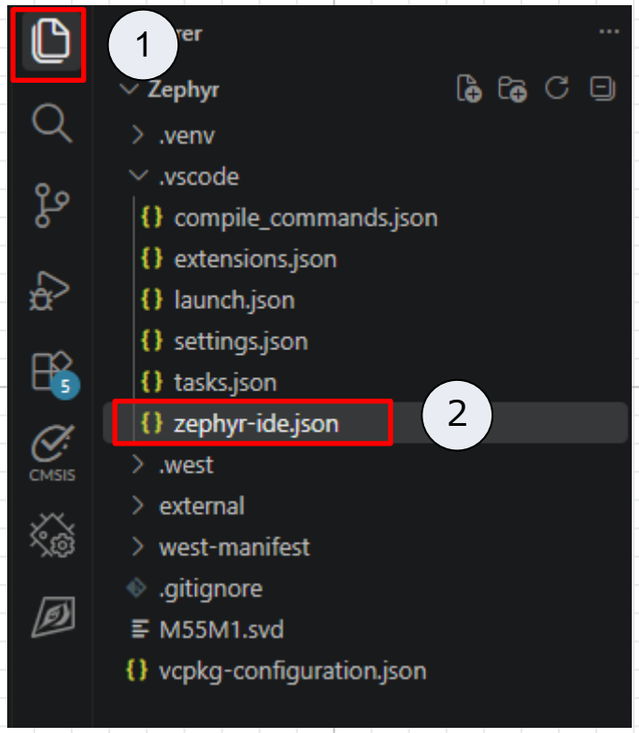




© 2026 MARUBUN CORPORATION

# Flash書き込み環境構築

## Runner Profileの編集

①「Explorer」をクリックし、②「zephyr-ide.json」をクリックしてファイルを開きます。




© 2026 MARUBUN CORPORATION

Confidential

# Flash書き込み環境構築

追加された“ObjectDetection_Zephyr_yoloxn“の”westBuildCMakeArgs”に、“-DCONFIG_BUILD_OUTPUT_HEX=y”を追加します。修正が終わったら、Ctrl+Sでファイルを保存します。

```
61      "ObjectDetection_Zephyr_yoloxn": {
62        "relPath": "../Projects/ObjectDetection_Zephyr_yoloxn",
63        "name": "ObjectDetection_Zephyr_yoloxn",
64        "buildConfigs": {
65          "build/numaker_gai_m55m1/m55m1xxx": {
66            "relBoardDir": "../Projects/ObjectDetection_Zephyr_yoloxn",
67            "relBoardSubDir": "../../Zephyr/external/zephyr/boards/nuvoton/numaker_gai_m55m1",
68            "board": "numaker_gai_m55m1/m55m1xxx",
69            "name": "build/numaker_gai_m55m1/m55m1xxx",
70            "compilerOptimization": "speed",
71            "westBuildArgs": [],
72            "westBuildCMakeArgs": [
73              "-DCONFIG_SPEED_OPTIMIZATIONS=y"
74            ],
75            "confFiles": {
76              "config": [],
77              "overlay": []
78            }
79          }
80        },
```

```
72            "westBuildCMakeArgs": [
73              "-DCONFIG_SPEED_OPTIMIZATIONS=y",
74              "-DCONFIG_BUILD_OUTPUT_HEX=y"
75            ],
```

記載内容

- "-DCONFIG_SPEED_OPTIMIZATIONS=y",
- "-DCONFIG_BUILD_OUTPUT_HEX=y"

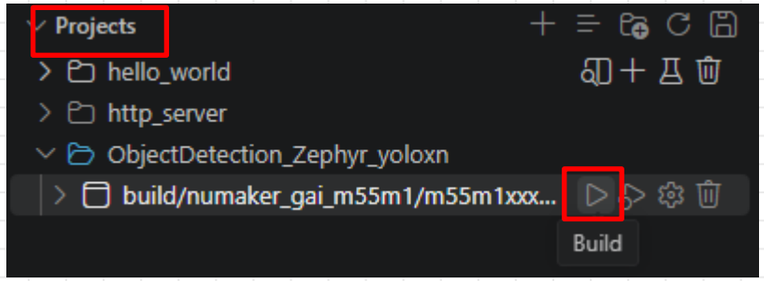
Since 1844 Marubun

© 2026 MARUBUN CORPORATION

# ビルド

## プロジェクトのビルド

「Projects」の「ObjectDetection_Zephyr_yoloxn: build/numaker_gai_m55m1」にある「Build」をクリックして、Terminal出力でビルド結果を確認します。（初回ビルドは完了まで十数分かかることがあります。）




ビルドが完了すると、以下にzephyr.binやzephyr.hexが生成されます。openOCDはこのHEXファイルを書き込みます。

- ObjectDetection_Zephyr_yoloxn¥build¥numaker_gai_m55m1¥m55m1xxx¥zephyr

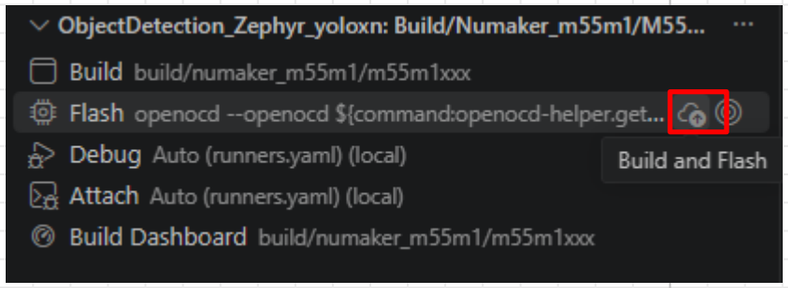



© 2026 MARUBUN CORPORATION

# Flash書き込み

## Flash書き込みの実行

「ObjectDetection_Zephyr_yoloxn: Build/Numaker_xxx」の「Flash」の「Build & Flash」をクリックして、Flash書き込みの完了を確認します。







© 2026 MARUBUN CORPORATION

Confidential

# 動作確認

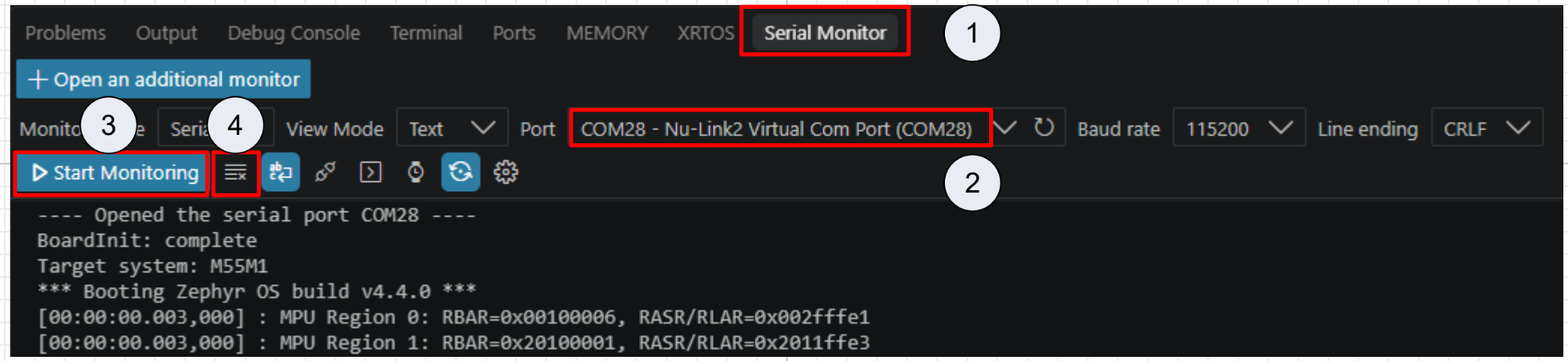
© 2026 MARUBUN CORPORATION

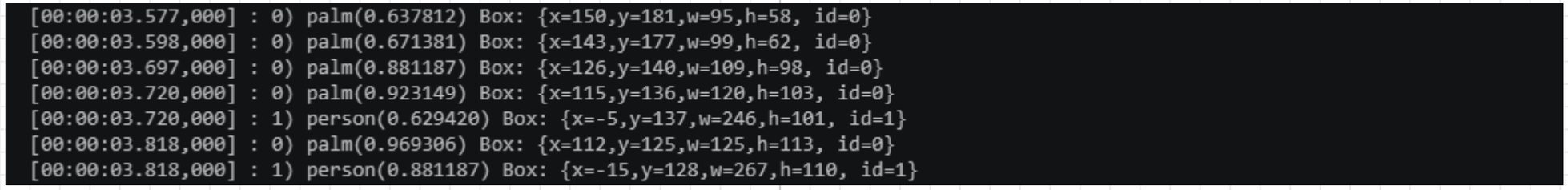
Confidential

# 動作確認

## ログ確認

- ① 「Serial Monitor」タブを開きます。
- ② 対象のCOMポートを選択します。
- ③ 「Start Monitoring」をクリックして接続状態（ボタンが「Stop Monitoring」の表示に変化）にします。
- ④ MCUにRESETがかかり、ログが出力されます。




© 2026 MARUBUN CORPORATION

# 動作確認

## Terminal出力例

Terminalへのログ出力例（抜粋）と項目の説明です。

- [00:00:09.943,000] : 0) peace(0.713342) Box: {x=133,y=91,w=63,h=113, id=0}
- [00:00:09.943,000] : 1) person(0.969306) Box: {x=-55,y=28,w=321,h=208, id=1}
- [00:00:10.070,000] : 0) peace(0.704950) Box: {x=140,y=79,w=60,h=90, id=0}
- [00:00:10.070,000] : 1) one(0.660891) Box: {x=140,y=77,w=60,h=94, id=0}
- [00:00:10.070,000] : 2) person(0.969306) Box: {x=-48,y=29,w=310,h=207, id=1}
- [00:00:10.209,000] : Total inference rate: 14

- 0 / 1 / 2: 同一フレーム内の検出インデックス（0始まりの連番）
- person / peace / one など：検出されたオブジェクト名・ジェスチャーの種類
- カッコ内の小数点値：信頼度スコア
- Box：Bounding Box座標（位置x, y、領域サイズw, h）
- id：ByteTrackによるトラッキングID（人は1始まりの連番、ジェスチャーは0固定）
- Total inference rate：5秒間の平均推論フレームレート



© 2026 MARUBUN CORPORATION

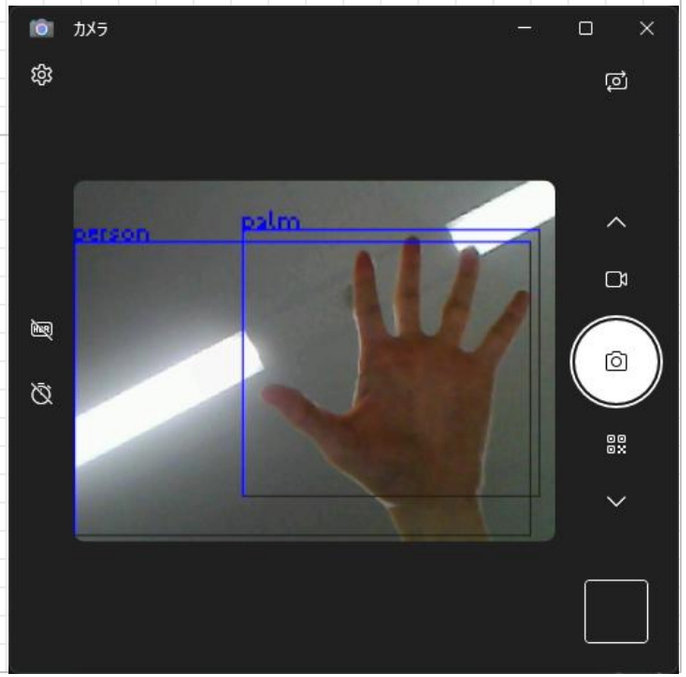
# 動作確認

## LCD出力

LCD出力を選択した場合は、LCDで推論画像が確認できます。（NuMaker-X-M55M1Dのみ）

## UVC出力

UVC出力を選択した場合は、PCとターゲットボードをUSB接続して、Windows標準のカメラアプリなどで推論画像が確認できます。（NuMaker-X-M55M1Dの場合は、FSUSBではなくHSUSBのコネクタを使用）




© 2026 MARUBUN CORPORATION

Confidential

# 動作モード別の処理内容/性能比較

## 解像度・パイプライン

| 処理項目 | プレビューあり (LCD/UVC出力) | プレビューなし (ログ出力のみ) |
|---|---|---|
| キャプチャ | 320×240 (RGB565) | 320×320 (RGB888) |
| 前処理 | 320×320へソフトウェア伸長 | なし |
| NPU推論 | 320×320 (INT8量子化) | 320×320 (INT8量子化) |
| 後処理 | 座標補正(0.75倍)＋BB描画 | なし |
| フレームレート※ | 12～14 fps | 24～26 fps |

※カメラで人や手をとらえて、推論処理を行った場合のベンチマーク例です。
「README.md」も合わせてご確認ください。

© 2026 MARUBUN CORPORATION

Confidential

# デバッグ環境

© 2026 MARUBUN CORPORATION

Confidential

# デバッグ環境の実行と動作確認

**※以下、「VS CodeによるZephyr開発 [Part2] ビルド&デバッグ環境編」の「デバッグ環境」手順を実施したことを前提とします。**

## デバッグセッションの設定と実行

- ① 「Run & Debug」をクリックします。
- ② 「Start Debugging」をクリックします。




デバッガが起動し、main関数の先頭で停止します。

Since 1844 Marubun

© 2026 MARUBUN CORPORATION


Since 1844
Marubun