VSCode + GNU Arm Toolchain and Build System User Guide

Outline – 1

- Install Development Tool & Plug
  - Visual Studio Code(VSCode)
  - Embedded IDE (EIDE)
  - .Net6 X64 Runtime Package
  - GNU Arm Embedded Toolchain
  - Openocd Programmer
  - Cortex-debug Plug
- Open Existing VSCode EIDE Project
- How to Add Tengen Compiler output C code(AI Model) in NPU Project
- How to Obtain pre/post process and inference time in NPU Project
- Project Switch Target
  - TinyML Project
  - NPU Project

Outline – 2

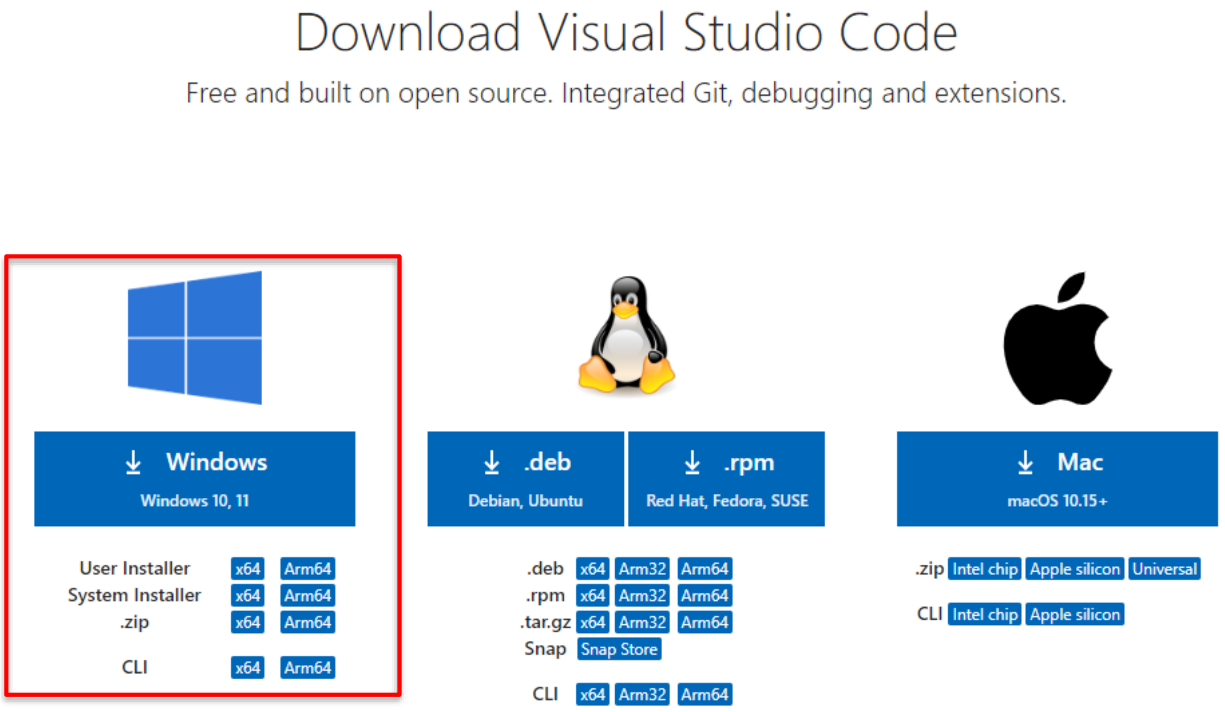
- Output Binary Path
  - TinyML Project
  - NPU Project
- Flash Download procedure and system run

Prepare: Install Development Tool & Plug

1. Install Visual Studio Code(VSCode)
   - https://code.visualstudio.com/download
2. Install Embedded IDE(EIDE) plug in VSCode
   - https://marketplace.visualstudio.com/items?itemName=CL.eide
3. Install .NET6 X64 Runtime package
   - After install EIDE done, The plug-in will auto download and install eide-binaries and .NET6 X64 Runtime package
4. Install GNU Arm Embedded Toolchain in VSCode
5. Install OpenOCD Programmer in VSCode
6. Install Cortex-Debug plug in VSCode

Prepare work:
Install Development Tool & Plug

Install Visual Studio Code(VSCode)



Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.

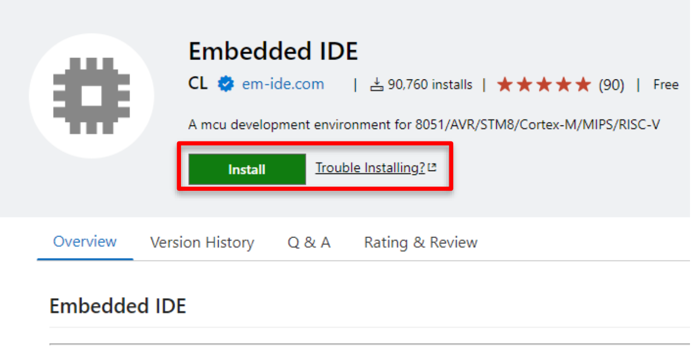
User Installer x64 Arm64
System Installer x64 Arm64
.zip x64 Arm64
CLI x64 Arm64

.deb x64 Arm32 Arm64
.rpm x64 Arm32 Arm64
.tar.gz x64 Arm32 Arm64
Snap Snap Store
CLI x64 Arm32 Arm64

.zip Intel chip Apple silicon Universal
CLI Intel chip Apple silicon

Install Embedded IDE(EIDE) in VSCode

- https://marketplace.visualstudio.com/items?itemName=CL.eide

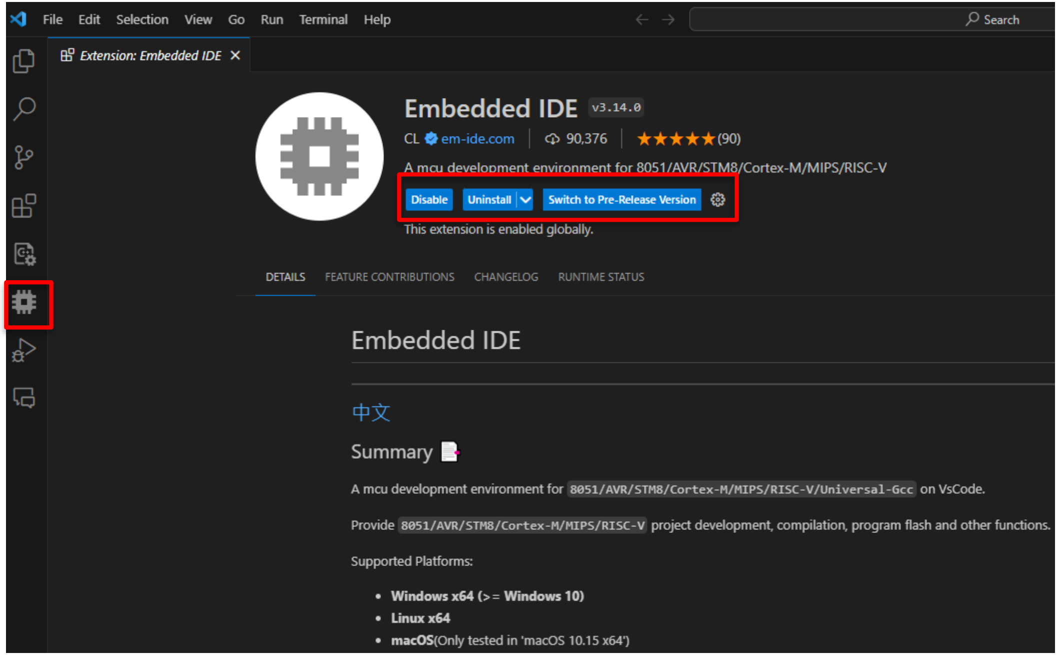


Embedded IDE

CL em-ide.com | 90,760 installs | ★★★★★ (90) | Free

A mcu development environment for 8051/AVR/STM8/Cortex-M/MIPS/RISC-V

Embedded IDE



Embedded IDE v3.14.0

CL em-ide.com | 90,376 | ★★★★★ (90)

A mcu development environment for 8051/AVR/STM8/Cortex-M/MIPS/RISC-V

This extension is enabled globally.

Embedded IDE

中文

Summary

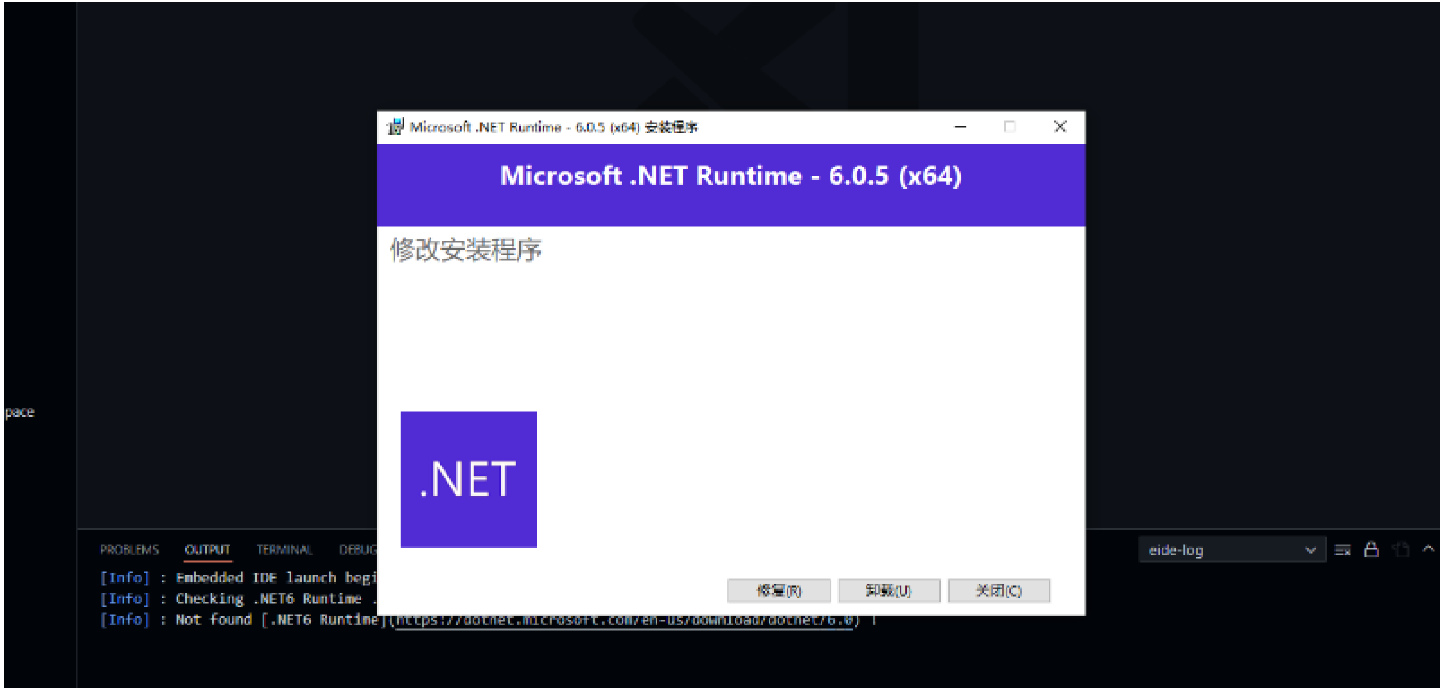
A mcu development environment for `8051/AVR/STM8/Cortex-M/MIPS/RISC-V/Universal-Gcc` on VsCode.

Provide `8051/AVR/STM8/Cortex-M/MIPS/RISC-V` project development, compilation, program flash and other functions.

Supported Platforms:

- **Windows x64 (>= Windows 10)**
- **Linux x64**
- **macOS**(Only tested in 'macOS 10.15 x64')

Install .NET6 X64 Runtime package



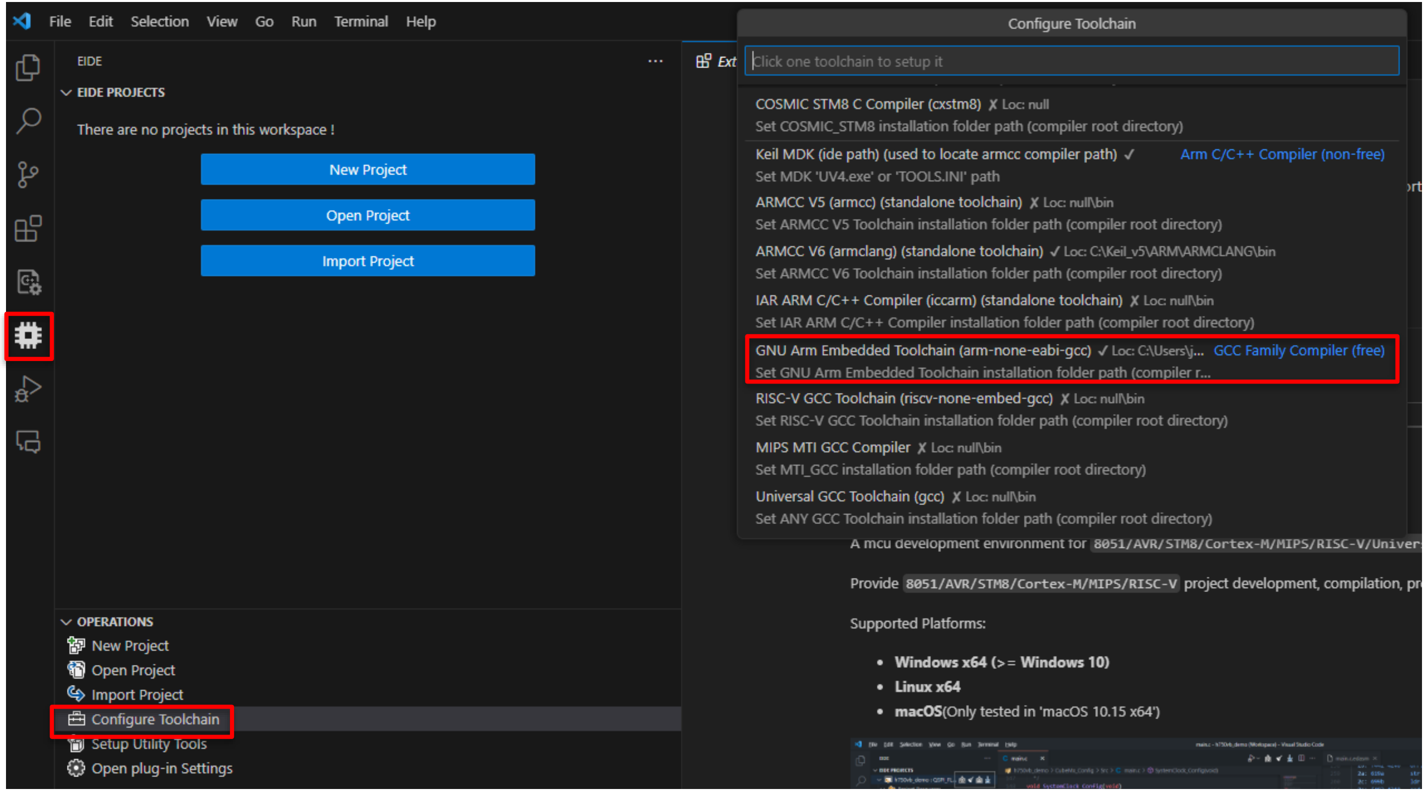
Microsoft .NET Runtime - 6.0.5 (x64) 安装程序

Microsoft .NET Runtime - 6.0.5 (x64)

修改安装程序

```
[Info] : Embedded IDE launch begi
[Info] : Checking .NET6 Runtime .
[Info] : Not found [.NET6 Runtime](https://dotnet.microsoft.com/en-us/download/dotnet/6.0) !
```

Install GNU Arm Embedded Toolchain in VSCode



There are no projects in this workspace !

Configure Toolchain

Click one toolchain to setup it

COSMIC STM8 C Compiler (cxstm8) ✗ Loc: null
Set COSMIC_STM8 installation folder path (compiler root directory)

Keil MDK (ide path) (used to locate armcc compiler path) ✓ Arm C/C++ Compiler (non-free)
Set MDK 'UV4.exe' or 'TOOLS.INI' path

ARMCC V5 (armcc) (standalone toolchain) ✗ Loc: null\bin
Set ARMCC V5 Toolchain installation folder path (compiler root directory)

ARMCC V6 (armclang) (standalone toolchain) ✓ Loc: C:\Keil_v5\ARM\ARMCLANG\bin
Set ARMCC V6 Toolchain installation folder path (compiler root directory)

IAR ARM C/C++ Compiler (iccarm) (standalone toolchain) ✗ Loc: null\bin
Set IAR ARM C/C++ Compiler installation folder path (compiler root directory)

GNU Arm Embedded Toolchain (arm-none-eabi-gcc) ✓ Loc: C:\Users\j... GCC Family Compiler (free)
Set GNU Arm Embedded Toolchain installation folder path (compiler r...

RISC-V GCC Toolchain (riscv-none-embed-gcc) ✗ Loc: null\bin
Set RISC-V GCC Toolchain installation folder path (compiler root directory)

MIPS MTI GCC Compiler ✗ Loc: null\bin
Set MTI_GCC installation folder path (compiler root directory)

Universal GCC Toolchain (gcc) ✗ Loc: null\bin
Set ANY GCC Toolchain installation folder path (compiler root directory)

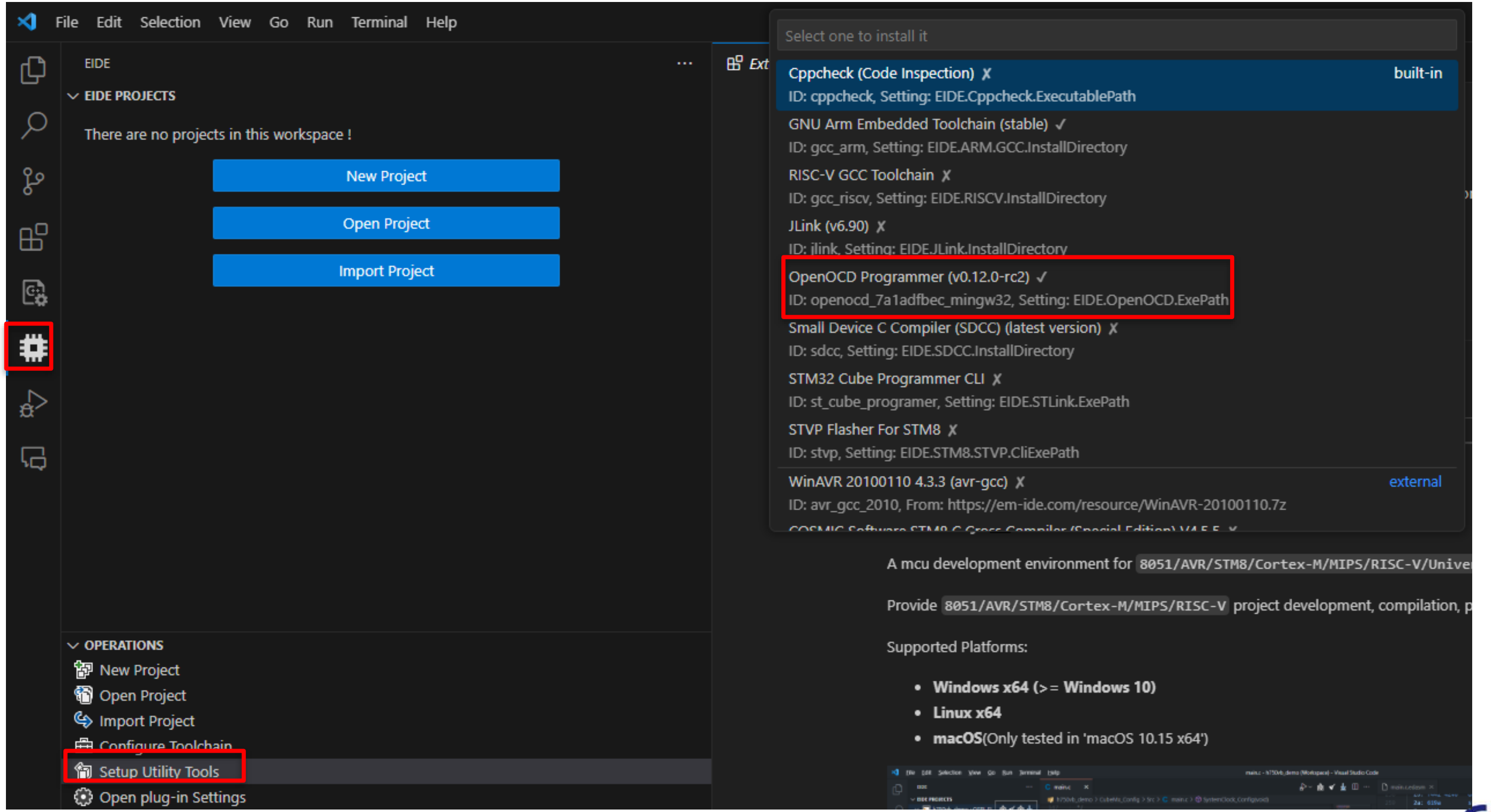
A mcu development environment for `8051/AVR/STM8/Cortex-M/MIPS/RISC-V/Univer`

Provide `8051/AVR/STM8/Cortex-M/MIPS/RISC-V` project development, compilation, pr

Supported Platforms:

- **Windows x64 (>= Windows 10)**
- **Linux x64**
- **macOS**(Only tested in 'macOS 10.15 x64')

Install OpenOCD Programmer in VSCode



There are no projects in this workspace !

Select one to install it

Cppcheck (Code Inspection) ✗ built-in
ID: cppcheck, Setting: EIDE.Cppcheck.ExecutablePath

GNU Arm Embedded Toolchain (stable) ✓
ID: gcc_arm, Setting: EIDE.ARM.GCC.InstallDirectory

RISC-V GCC Toolchain ✗
ID: gcc_riscv, Setting: EIDE.RISCV.InstallDirectory

JLink (v6.90) ✗
ID: jlink, Setting: EIDE.JLink.InstallDirectory

OpenOCD Programmer (v0.12.0-rc2) ✓
ID: openocd_7a1adfbec_mingw32, Setting: EIDE.OpenOCD.ExePath

Small Device C Compiler (SDCC) (latest version) ✗
ID: sdcc, Setting: EIDE.SDCC.InstallDirectory

STM32 Cube Programmer CLI ✗
ID: st_cube_programer, Setting: EIDE.STLink.ExePath

STVP Flasher For STM8 ✗
ID: stvp, Setting: EIDE.STM8.STVP.CliExePath

WinAVR 20100110 4.3.3 (avr-gcc) ✗ external
ID: avr_gcc_2010, From: https://em-ide.com/resource/WinAVR-20100110.7z

COSMIC Software STM8 C Cross Compiler (Special Edition) V4.5.5 ✗

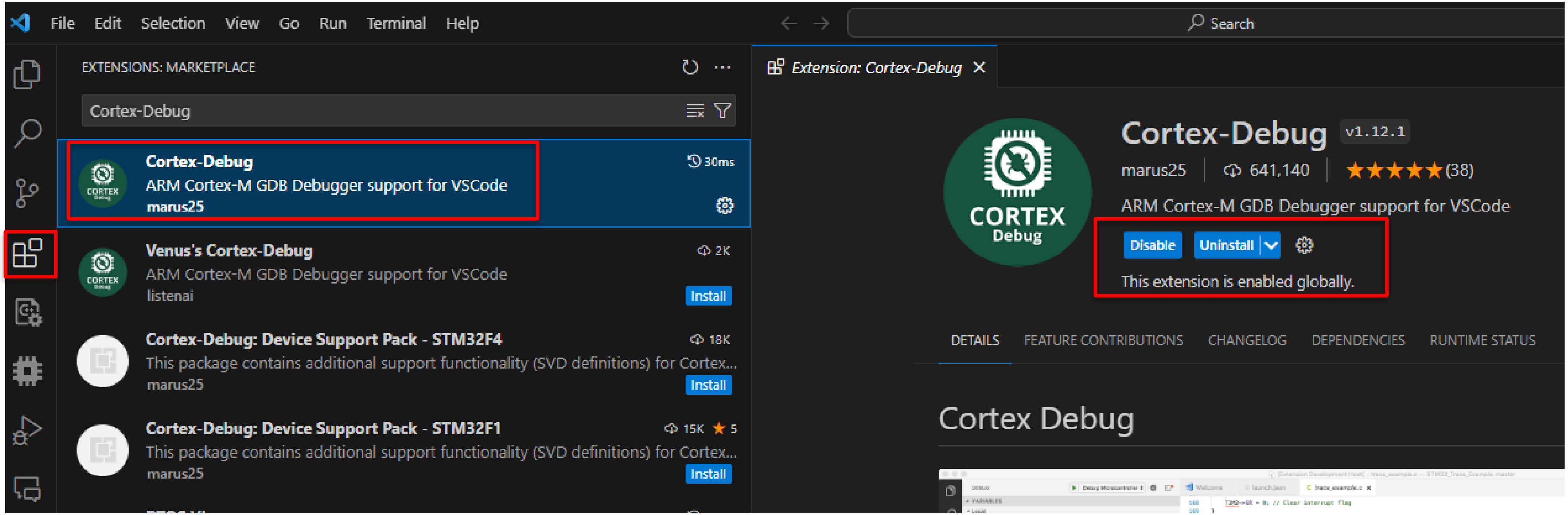
A mcu development environment for `8051/AVR/STM8/Cortex-M/MIPS/RISC-V/Unive`

Provide `8051/AVR/STM8/Cortex-M/MIPS/RISC-V` project development, compilation, p

Supported Platforms:

- **Windows x64 (>= Windows 10)**
- **Linux x64**
- **macOS**(Only tested in 'macOS 10.15 x64')

Install Cortex-Debug plug in VSCode



Cortex-Debug

Cortex-Debug 30ms
ARM Cortex-M GDB Debugger support for VSCode
marus25

Venus's Cortex-Debug 2K
ARM Cortex-M GDB Debugger support for VSCode
listenai

Cortex-Debug: Device Support Pack - STM32F4 18K
This package contains additional support functionality (SVD definitions) for Cortex...
marus25

Cortex-Debug: Device Support Pack - STM32F1 15K ★ 5
This package contains additional support functionality (SVD definitions) for Cortex...
marus25

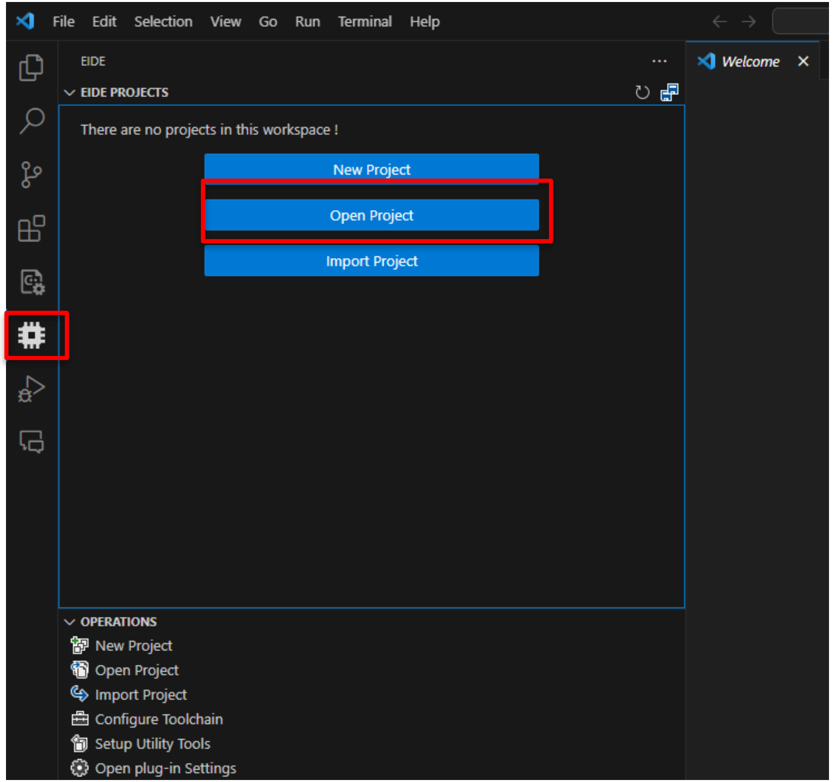
Cortex-Debug v1.12.1
marus25 | 641,140 | ★★★★★ (38)
ARM Cortex-M GDB Debugger support for VSCode

This extension is enabled globally.

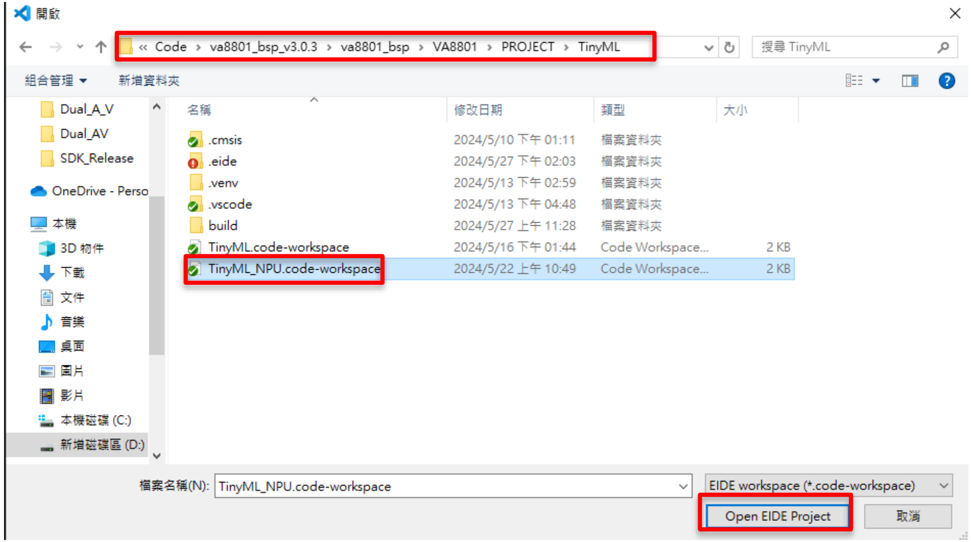
Cortex Debug

Open VSCode EIDE Project

Open Existing EIDE Project – 1



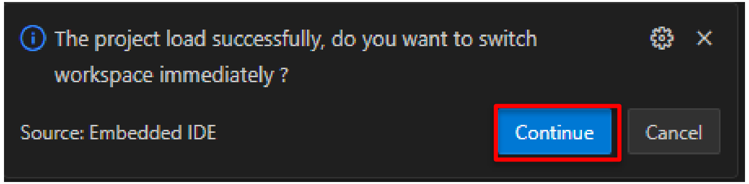
There are no projects in this workspace !



名稱	修改日期	類型	大小
.cmsis	2024/5/10 下午 01:11	檔案資料夾	
.eide	2024/5/27 下午 02:03	檔案資料夾	
.venv	2024/5/13 下午 02:59	檔案資料夾	
.vscode	2024/5/13 下午 04:48	檔案資料夾	
build	2024/5/27 上午 11:28	檔案資料夾	
TinyML.code-workspace	2024/5/16 下午 01:44	Code Workspace...	2 KB
TinyML_NPU.code-workspace	2024/5/22 上午 10:49	Code Workspace...	2 KB

檔案名稱(N): TinyML_NPU.code-workspace

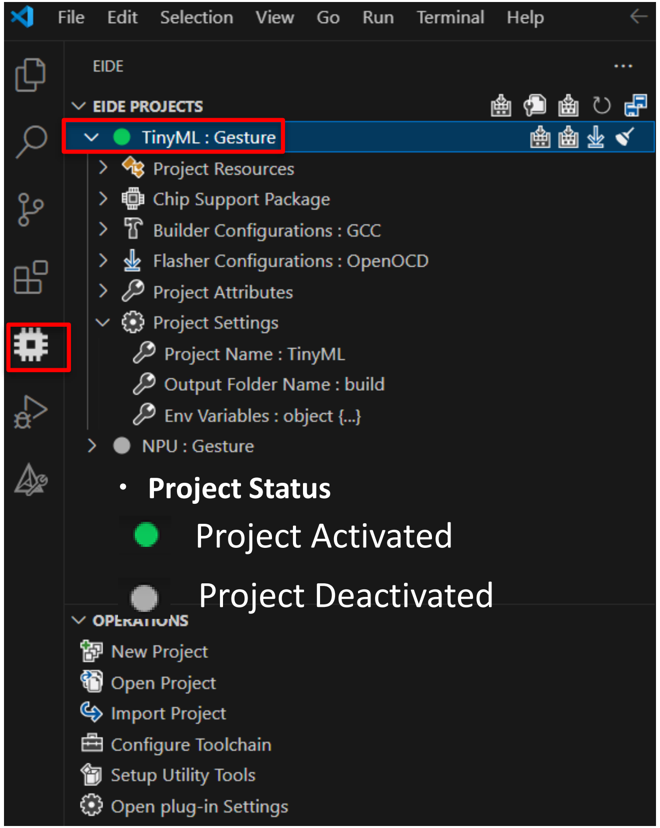
EIDE workspace (*.code-workspace)



The project load successfully, do you want to switch workspace immediately ?

Source: Embedded IDE

Open Existing EIDE Project (TinyML) – 2



EIDE PROJECTS

- TinyML : Gesture
  - Project Resources
  - Chip Support Package
  - Builder Configurations : GCC
  - Flasher Configurations : OpenOCD
  - Project Attributes
  - Project Settings
    - Project Name : TinyML
    - Output Folder Name : build
    - Env Variables : object {...}
- NPU : Gesture

- **Project Status**

● Project Activated

● Project Deactivated

OPERATIONS

- New Project
- Open Project
- Import Project
- Configure Toolchain
- Setup Utility Tools
- Open plug-in Settings

- **Project Resources**
  - Display project source code folder, builder output files
- **Chip Support Package**
  - Ignore
- **Builder Configuration**
  - Setting CPU Type, Linker Script(.ld), Complier/Linker configuration etc..
- **Flasher Configuration**
  - Setting Debug mode configuration
- **Project Attributes**
  - Setting Project Include header file paths, Preprocessor Defines, Library Search Directories etc..
- **Project Setting**
  - Setting Project Name, Output Folder Name, Environment Variables

EIDE Project

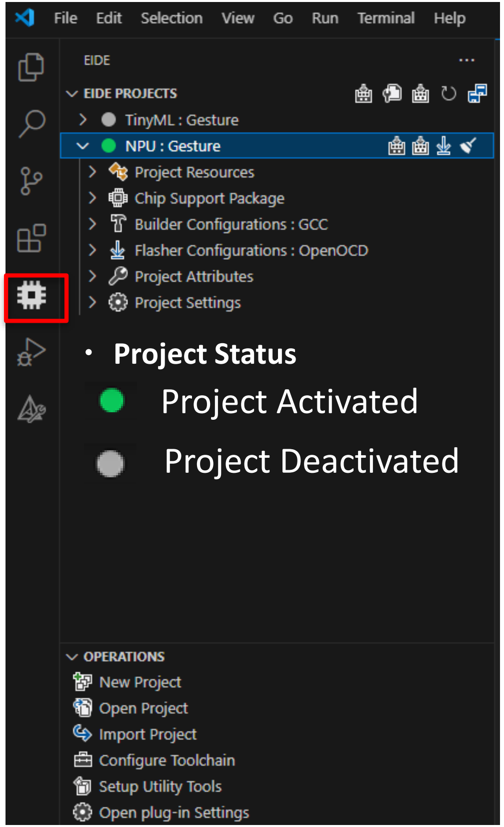
Build

Clean

Program Flash

Rebuild

Open Existing EIDE Project (NPU) – 3



EIDE

EIDE PROJECTS

- TinyML : Gesture
- NPU : Gesture
  - Project Resources
  - Chip Support Package
  - Builder Configurations : GCC
  - Flasher Configurations : OpenOCD
  - Project Attributes
  - Project Settings

- **Project Status**

● Project Activated

● Project Deactivated

OPERATIONS

- New Project
- Open Project
- Import Project
- Configure Toolchain
- Setup Utility Tools
- Open plug-in Settings

- **Project Resources**
  - Display project source code folder, builder output files
  - Tengen Compiler output file(AI Model C code)
- **Chip Support Package**
  - Ignore
- **Builder Configuration**
  - Setting CPU Type, Linker Script(.ld), Complier/Linker configuration etc..
- **Flasher Configuration**
  - Setting Debug mode configuration
- **Project Attributes**
  - Setting Project Include header file paths, Preprocessor Defines, Library Search Directories etc..
- **Project Setting**
  - Setting Project Name, Output Folder Name, Environment Variables

EIDE Project

Build

Clean

Program Flash

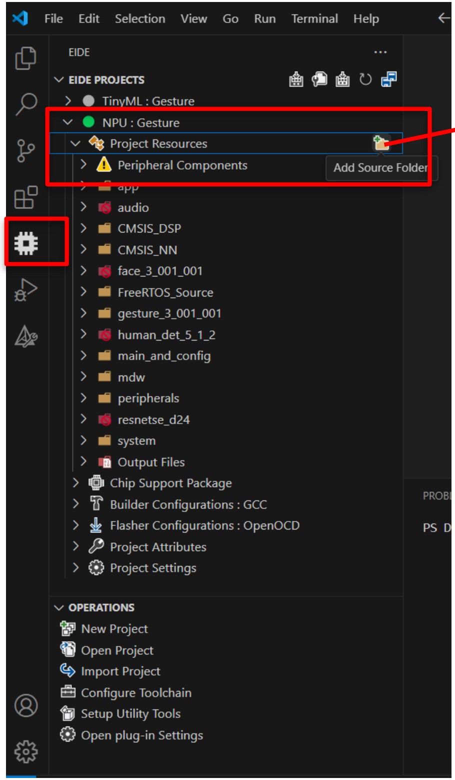
Rebuild

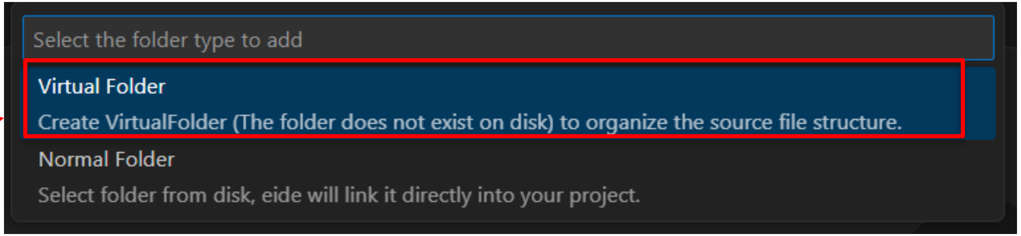
How to Add Tengen Compiler output C code(AI Model) in NPU Project

How to Add Tengen Compiler output C code(AI Model) in NPU Project – 1

- Prepare the AI model (There are the following two options)
  1. Download from the VA8801 Model Zoo
     - https://github.com/FITI-HCITA/VA8801_Model_Zoo
  2. Self-develop AI model
- Use Tengen Compiler to convert the AI model into C code
  1. Detail reference: SDK root Path\VA8801_BSPSDK_V3.000.000_release\Tengen Compiler\Tengen Compiler User Guide v1.0.3.pdf
- Tengen Compiler output AI mode C code inc and src file put to VA8801_BSPSDK_V3.000.000 _release \ Code\va8801_bsp-v3.000.000.zip\va8801_bsp-v3.000.000\VA8801\NPU\Middleware\codegen
- How to add AI model C code inc and src in VSCode NPU Project
  - Reference next page

How to Add Tengen Compiler output C code(AI Model) in NPU Project – 2

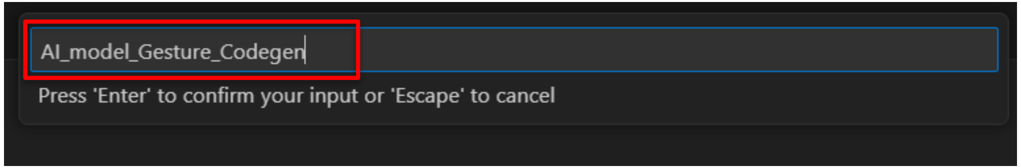




Select the folder type to add

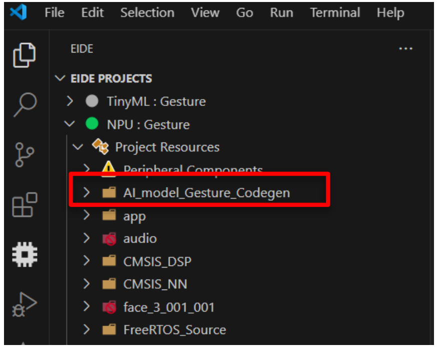
Virtual Folder
Create VirtualFolder (The folder does not exist on disk) to organize the source file structure.

Normal Folder
Select folder from disk, eide will link it directly into your project.

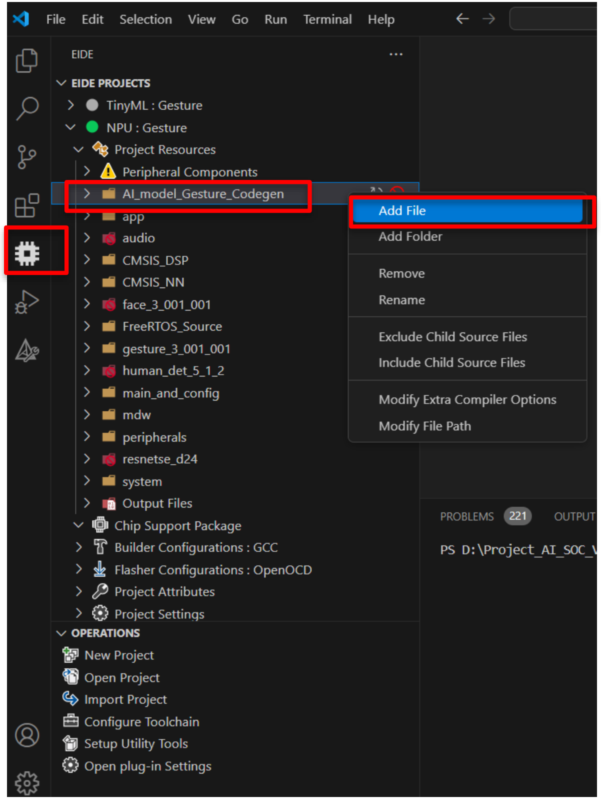


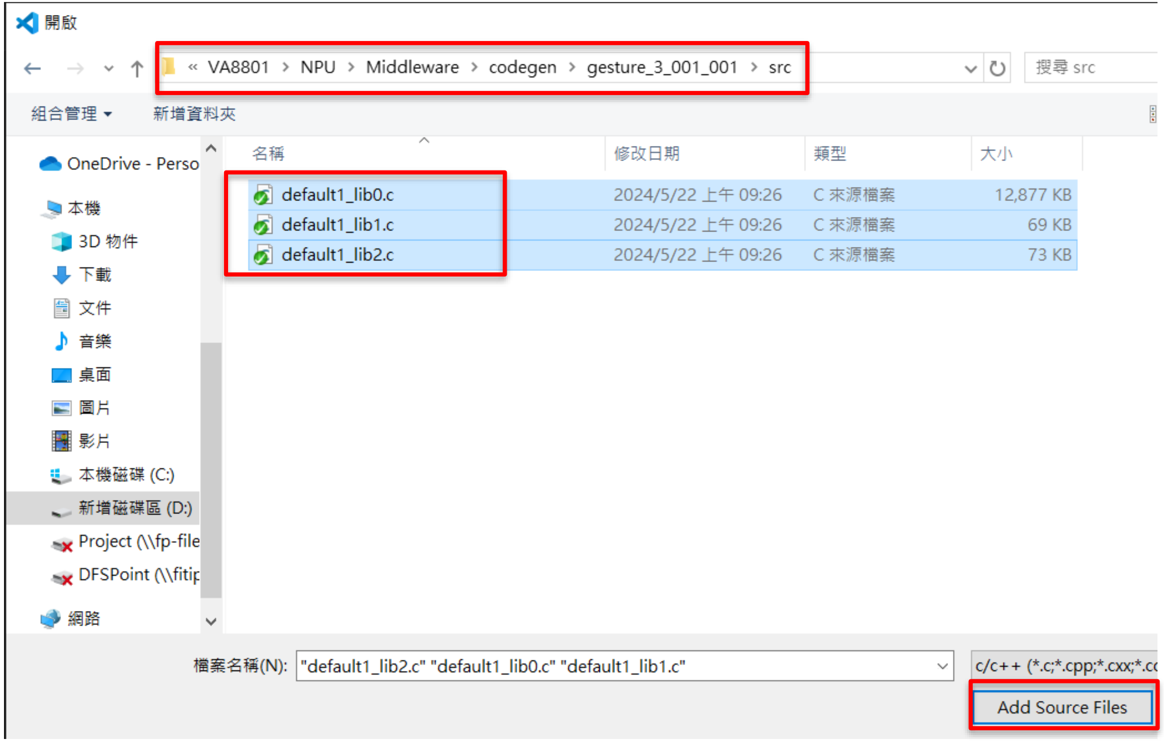
AI_model_Gesture_Codegen

Press 'Enter' to confirm your input or 'Escape' to cancel



How to Add Tengen Compiler output C code(AI Model) in NPU Project – 3

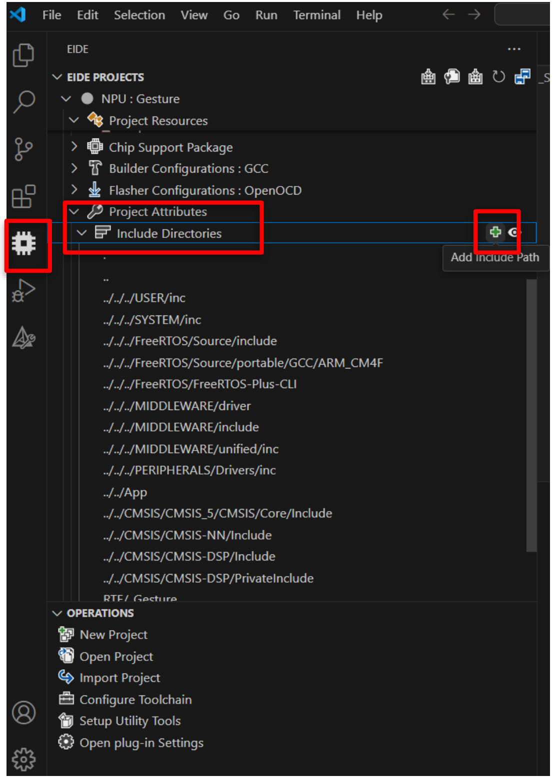




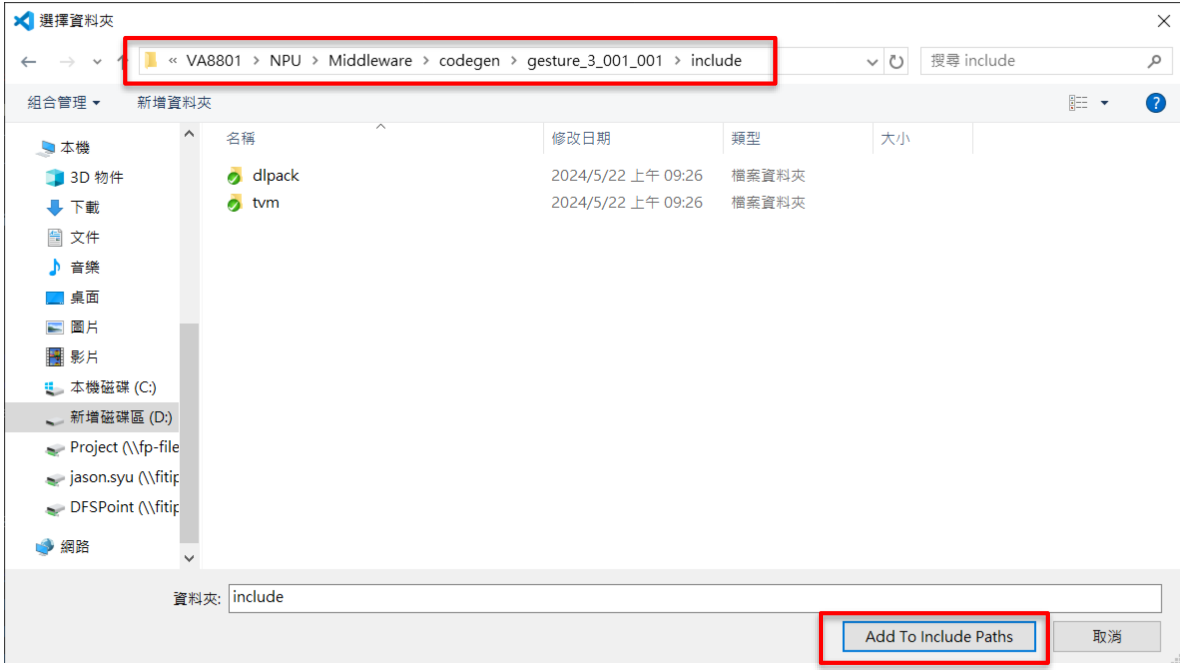
名稱	修改日期	類型	大小
default1_lib0.c	2024/5/22 上午 09:26	C 來源檔案	12,877 KB
default1_lib1.c	2024/5/22 上午 09:26	C 來源檔案	69 KB
default1_lib2.c	2024/5/22 上午 09:26	C 來源檔案	73 KB

檔案名稱(N): "default1_lib2.c" "default1_lib0.c" "default1_lib1.c"

How to Add Tengen Compiler output C code(AI Model) in NPU Project – 4



```
.
..
../../../USER/inc
../../../SYSTEM/inc
../../../FreeRTOS/Source/include
../../../FreeRTOS/Source/portable/GCC/ARM_CM4F
../../../FreeRTOS/FreeRTOS-Plus-CLI
../../../MIDDLEWARE/driver
../../../MIDDLEWARE/include
../../../MIDDLEWARE/unified/inc
../../../PERIPHERALS/Drivers/inc
../../App
../../CMSIS/CMSIS_5/CMSIS/Core/Include
../../CMSIS/CMSIS-NN/Include
../../CMSIS/CMSIS-DSP/Include
../../CMSIS/CMSIS-DSP/PrivateInclude
RTE/_Gesture
```

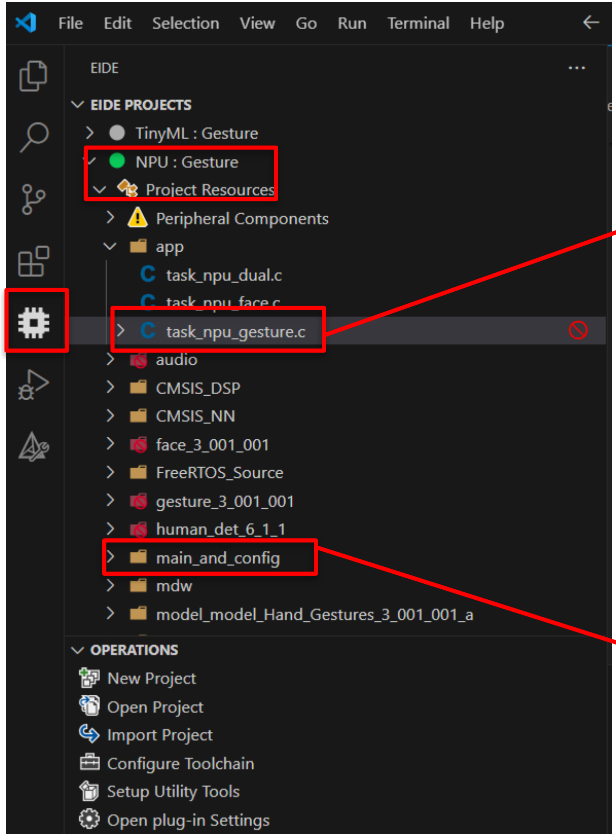


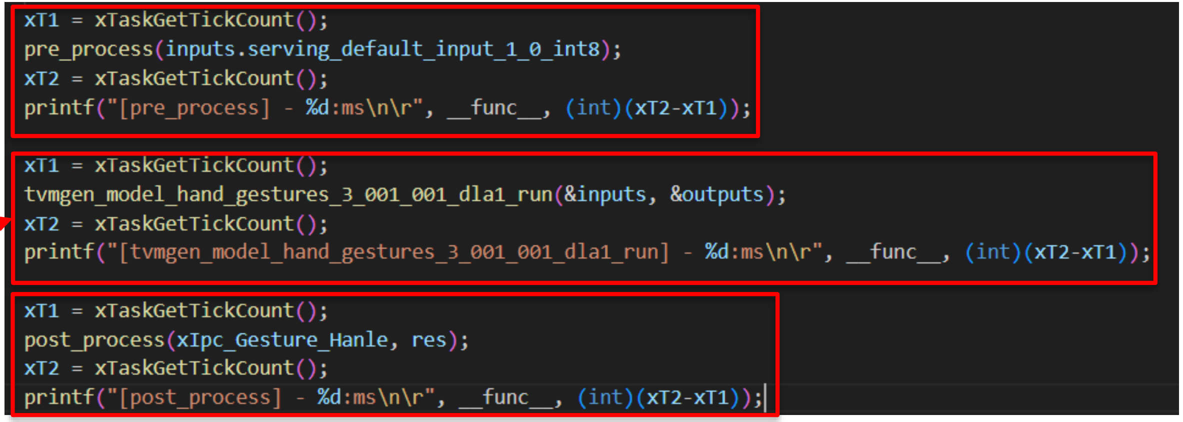
名稱	修改日期	類型	大小
dlpack	2024/5/22 上午 09:26	檔案資料夾	
tvm	2024/5/22 上午 09:26	檔案資料夾	

資料夾: include

How to Obtain pre/post process and inference time in NPU Project

How to Obtain pre/post process and **inference time** in NPU Project



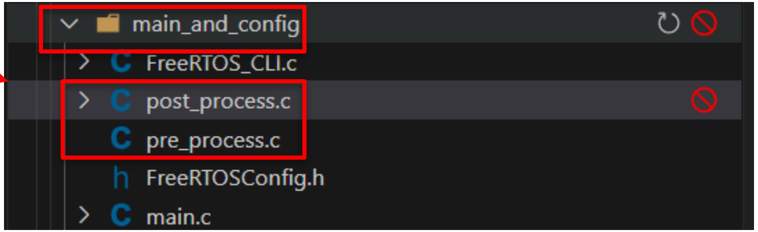


```
xT1 = xTaskGetTickCount();
pre_process(inputs.serving_default_input_1_0_int8);
xT2 = xTaskGetTickCount();
printf("[pre_process] - %d:ms\n\r", __func__, (int)(xT2-xT1));

xT1 = xTaskGetTickCount();
tvmgen_model_hand_gestures_3_001_001_dla1_run(&inputs, &outputs);
xT2 = xTaskGetTickCount();
printf("[tvmgen_model_hand_gestures_3_001_001_dla1_run] - %d:ms\n\r", __func__, (int)(xT2-xT1));

xT1 = xTaskGetTickCount();
post_process(xIpc_Gesture_Hanle, res);
xT2 = xTaskGetTickCount();
printf("[post_process] - %d:ms\n\r", __func__, (int)(xT2-xT1));
```

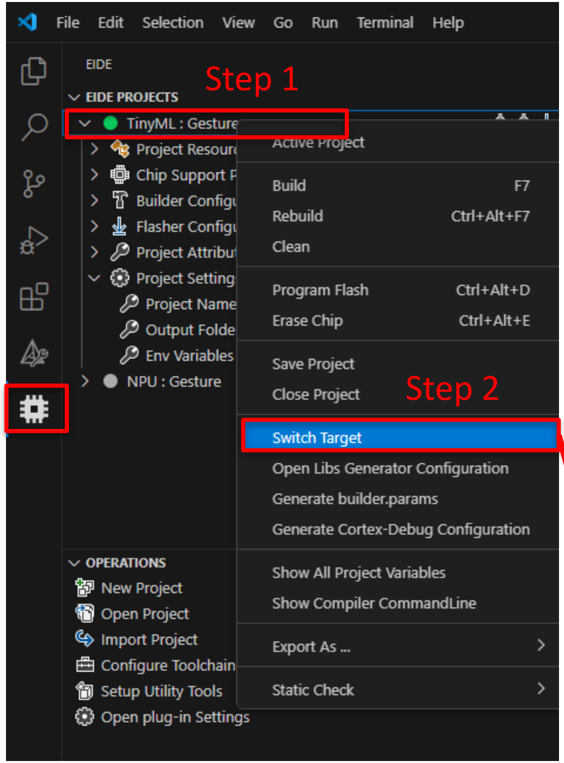
`static void prvGestureTask( void *pvParameters )`



Here you can self-define pre/post process develop

TinyML Project Switch Target

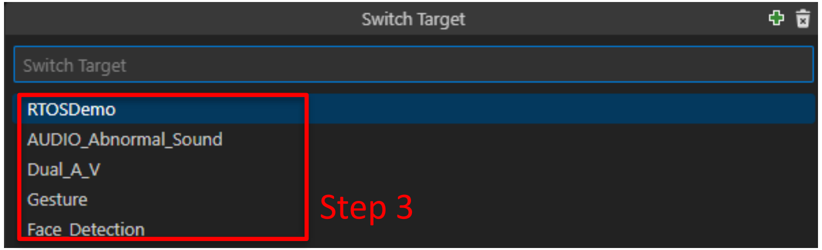
TinyML Project Switch Target – 1



Step 1

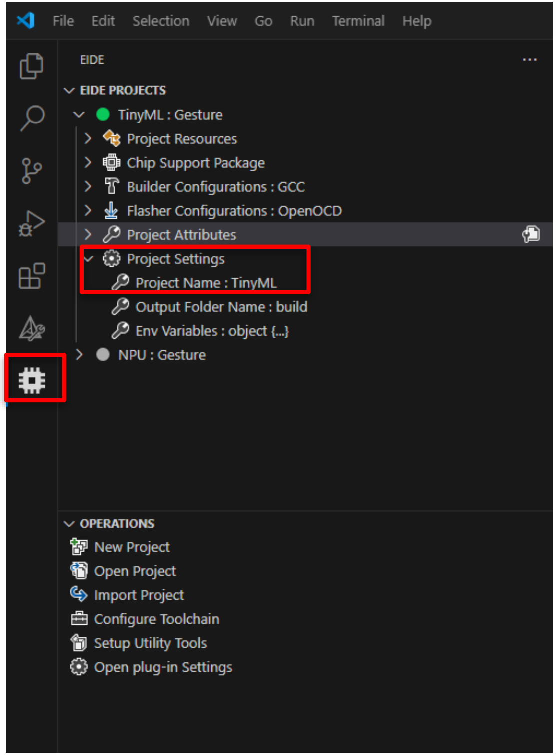
Step 2

- **Target Description**
  - **Dual_A_V :** Abnormal sound + Human detection
  - **Gesture :** Gesture Detection
  - **Face_Detection:** face detection
- **Step 1**
  - Right click
- **Step 2**
  - Click Switch Target
- **Step3**
  - Choose Target (ex: choose Gesture)



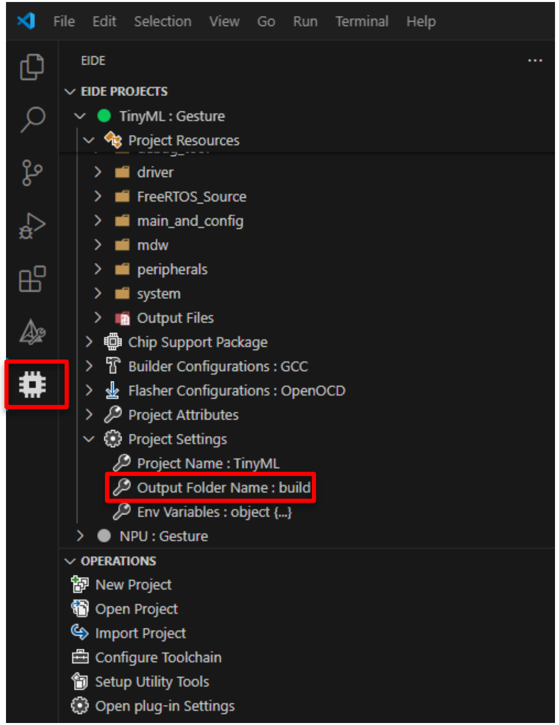
Step 3

TinyML Project Switch Target – 2



- **Step4 – Project Settings**
  - Modify Project Name (as follow)
    - **Target Dual_A_V :** SYS_A_V
    - **Target Gesture:** SYS_Gesture
    - **Target Face detection:** SYS_Face_detection

TinyML Project – Output Binary path



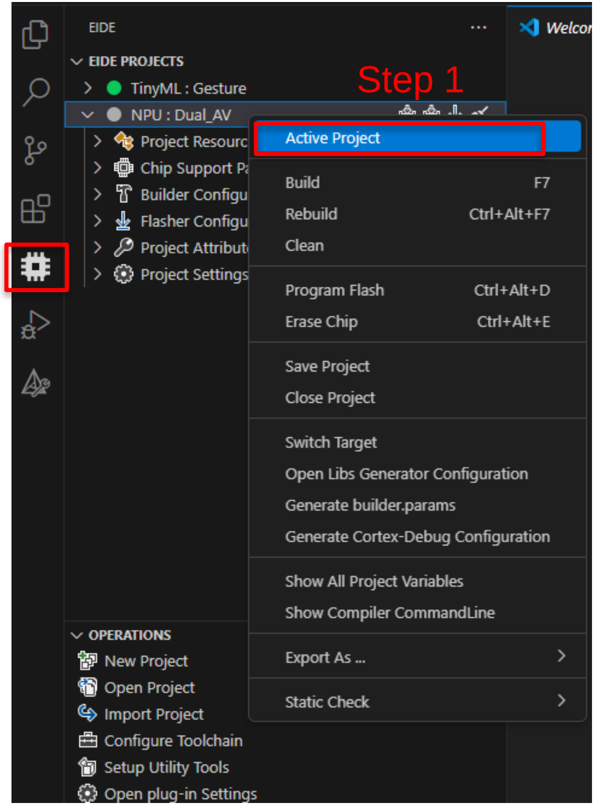
EIDE PROJECTS
- TinyML : Gesture
  - Project Resources
    - driver
    - FreeRTOS_Source
    - main_and_config
    - mdw
    - peripherals
    - system
    - Output Files
  - Chip Support Package
  - Builder Configurations : GCC
  - Flasher Configurations : OpenOCD
  - Project Attributes
  - Project Settings
    - Project Name : TinyML
    - Output Folder Name : build
    - Env Variables : object {...}
- NPU : Gesture

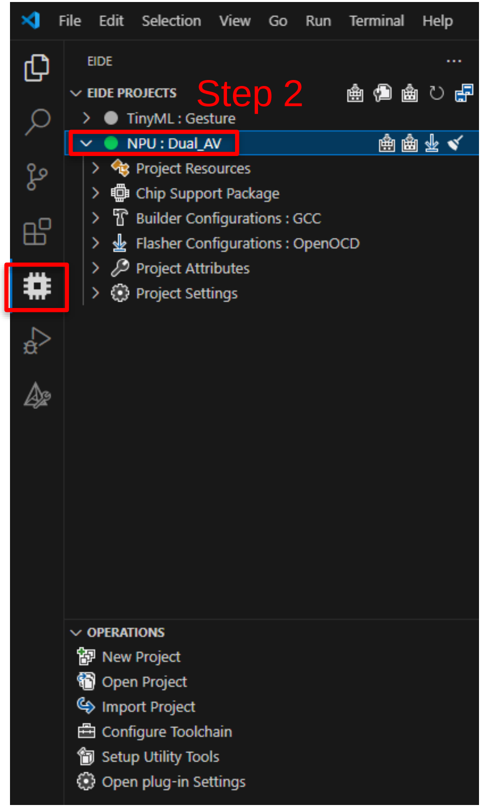
OPERATIONS
- New Project
- Open Project
- Import Project
- Configure Toolchain
- Setup Utility Tools
- Open plug-in Settings

- **Output binary path**
  - SDK root path\va8801_bsp\Fiti_M4F\PROJECT\TinyML\build
    - **Target Dual_A_V :** Dual_A_V\SYS_A_V.bin
    - **Target Gesture:** Gesture\SYS_Gesture.bin
    - **Target Face detection:** Face_Detection\SYS_Face_detection.bin

NPU Project Switch Target

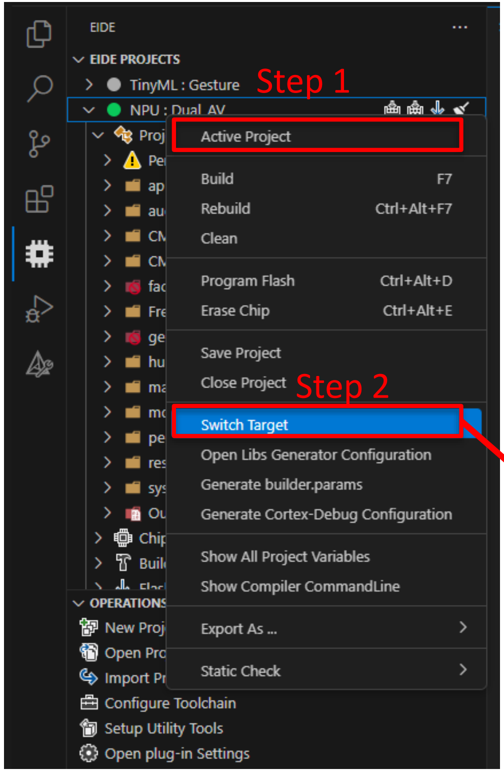
NPU Project Switch Target – 1





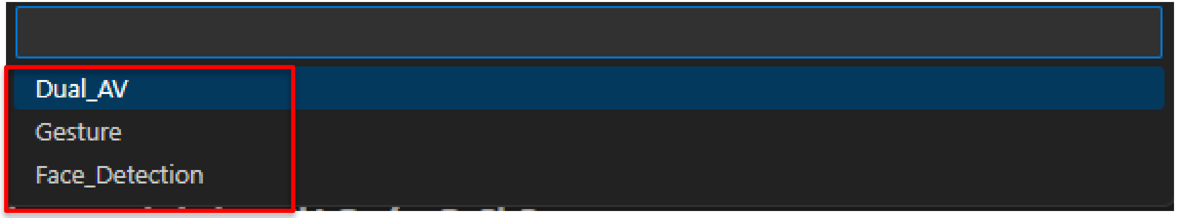
- **Step 1**
  - Right click, Choose Active Project
- **Step 2**
  - Check if the NPU project is active (indicated by a green light)

NPU Project Switch Target – 2



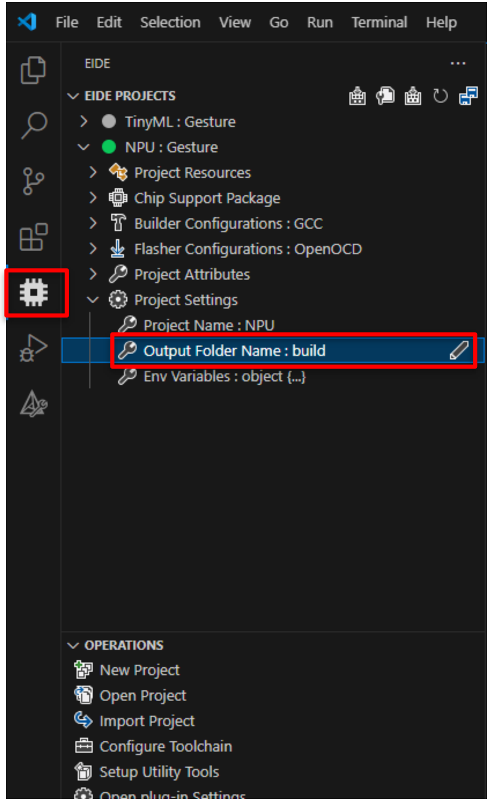
Step 1

Step 2



- **Target Description**
  - **Dual_A_V :** Abnormal sound + Human detection
  - **Gesture :** Gesture Detection
  - **Face_Detection:** face detection
- **Step 1**
  - Right click
- **Step 2**
  - Click Switch Target
- **Step3**
  - Choose Target (ex: choose Gesture)

NPU Project – Output Binary path



- TinyML : Gesture
- NPU : Gesture
  - Project Resources
  - Chip Support Package
  - Builder Configurations : GCC
  - Flasher Configurations : OpenOCD
  - Project Attributes
  - Project Settings
    - Project Name : NPU
    - Output Folder Name : build
    - Env Variables : object {...}

OPERATIONS

- New Project
- Open Project
- Import Project
- Configure Toolchain
- Setup Utility Tools
- Open plug-in Settings

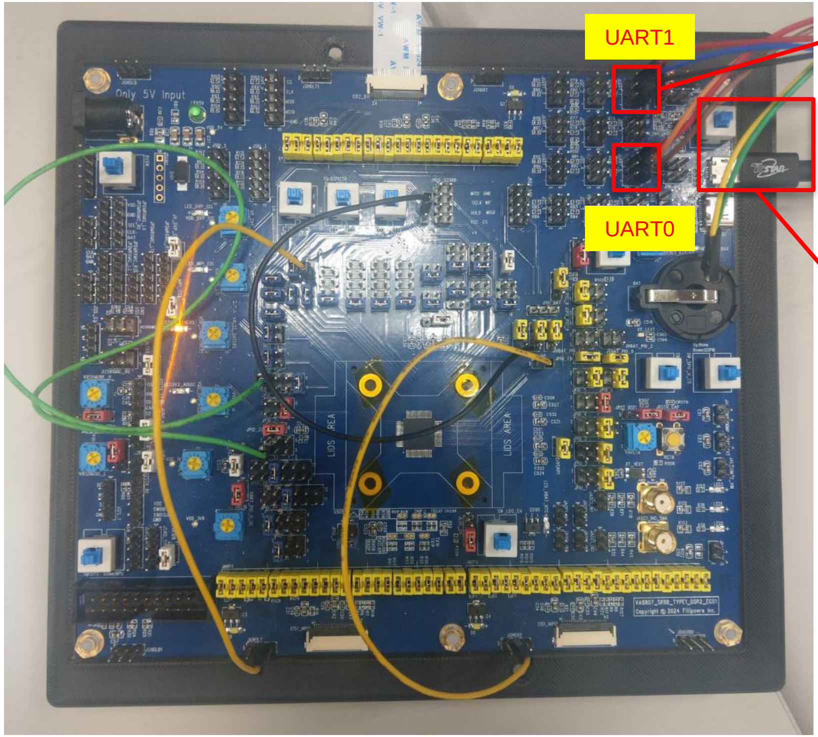
- **Output binary path**
  - SDK root path\va8801_bsp\VA8801\NPU\Project\NPU\build
    - **Target Dual_A_V :**
      - Dual_A_V\NPU_code.bin
      - Dual_A_V\NPU_data.bin
    - **Target Gesture**
      - Gesture\NPU_code.bin
      - Gesture\NPU_data.bin
    - **Target Face detection:**
      - Face_Detection\NPU_code.bin
      - Face_Detection\NPU_data.bin

Flash Download procedure and System run

Flash Download procedure and system run – 1

- Flash Download procedure
  - Reference SDK root path\VA8801_BSPSDK_V3.000.000_release\DFU Tool\FITI_VA8801_DFU_ToolKit_v1.0.0_20240522_1430.pdf
- System run scenarios include Dual_A_V, Face Detection, and Gesture
  - ➢**Dual_A_V:** TinyML & NPU Project Target needs to choose Dual_A_V
  - ➢**Face Detection :** TinyML & NPU Project Target needs to choose Face Detection
  - ➢**Gesture :** TinyML & NPU Project Target needs to choose Gesture
- Demo tool execute file
  - Reference SDK root path\VA8801_BSPSDK_V3.000.000_release\Demo Tool
  - Demo tool operation reference page 30

Flash Download procedure and system run – VA8801 HW Settings



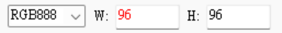
Pin from top to bottom:
UART_RX
UART_TX
UART_GND

FT232 connect to PC/NB
FT232_RX connect to UART_TX
FT232_TX connect to UART_RX
FT232_GND connect to UART_GND

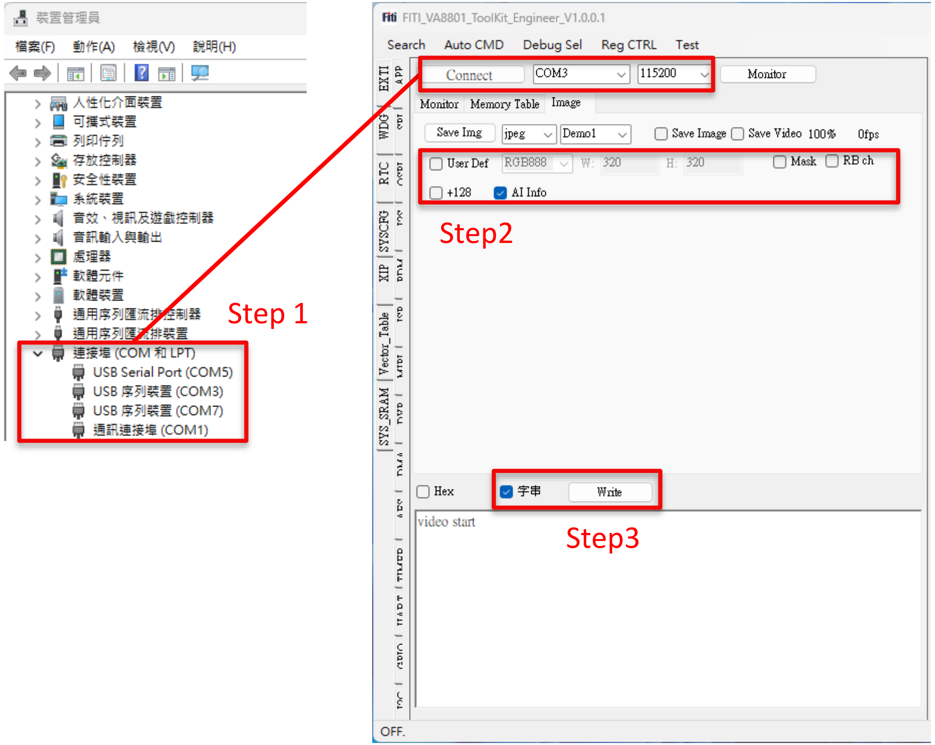
Micro USB connect to PC/NB
Push button to power on

Demo Tool Operation

- **Step1 – Tool connect to VA8801 via USB**
  - In device manager, find VA8801 USB COM port
  - Select baud rate 115200
  - Click connect
- **Step2 – Configure Tool options**
  - Tick User Def
    - Face Detection, Gesture - RGB888 W:320 H:320
    - A+V - Y8 W:96 H:96
    - Push enter after input W and H (red text will turn black)
  - Tick RB ch
  - Tick +128
  - Tick AI info
    - Enable AI info (bounding box)
- **Step3 – Display inference result**
  - Click Write



RGB888 W: 96 H: 96



装置管理員

檔案(F) 動作(A) 檢視(V) 說明(H)

- 人性化介面裝置
- 可攜式裝置
- 列印佇列
- 存放控制器
- 安全性裝置
- 系統裝置
- 音效、視訊及遊戲控制器
- 音訊輸入與輸出
- 處理器
- 軟體元件
- 軟體裝置
- 通用序列匯流排控制器
- 通用序列匯流排裝置
- 連接埠 (COM 和 LPT)
  - USB Serial Port (COM5)
  - USB 序列裝置 (COM3)
  - USB 序列裝置 (COM7)
  - 通訊連接埠 (COM1)

Step 1

FITI_VA8801_ToolKit_Engineer_V1.0.0.1

Connect | COM3 | 115200 | Monitor

Save Img | jpeg | Demo1 | □ Save Image | □ Save Video 100% | 0fps

□ User Def | RGB888 | W: 320 | H: 320 | □ Mask | □ RB ch

□ +128 | ☑ AI Info

Step2

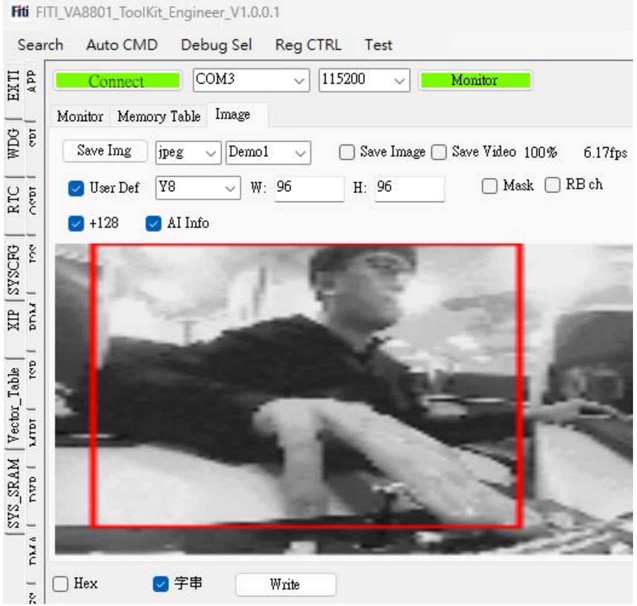
□ Hex | ☑ 字串 | Write

video start

Step3

OFF.

Flash Download procedure and system run – Scenario Dual_A_V


FITI_VA8801_ToolKit_Engineer_V1.0.0.1

COM3 | 115200

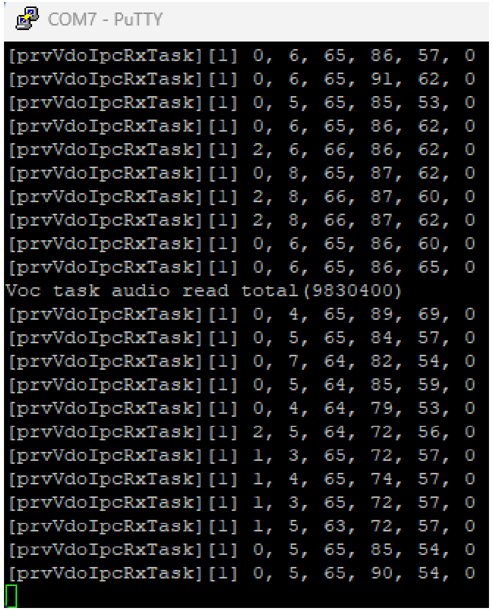
jpeg | Demo1 | □ Save Image □ Save Video 100% 6.17fps

☑ User Def Y8 W: 96 H: 96 □ Mask □ RB ch

☑ +128 ☑ AI Info

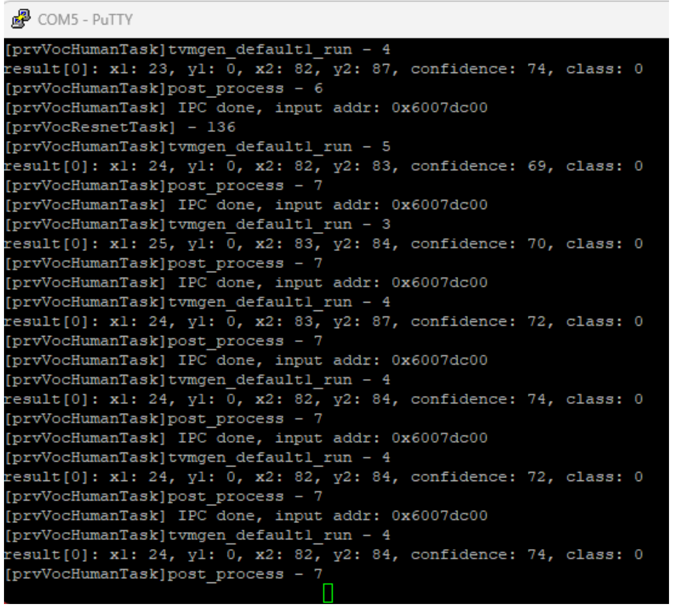
□ Hex ☑ 字串

Demo Tool – A+V


COM7 - PuTTY

```
[prvVdoIpcRxTask][1] 0, 6, 65, 86, 57, 0
[prvVdoIpcRxTask][1] 0, 6, 65, 91, 62, 0
[prvVdoIpcRxTask][1] 0, 5, 65, 85, 53, 0
[prvVdoIpcRxTask][1] 0, 6, 65, 86, 62, 0
[prvVdoIpcRxTask][1] 2, 6, 66, 86, 62, 0
[prvVdoIpcRxTask][1] 0, 8, 65, 87, 62, 0
[prvVdoIpcRxTask][1] 2, 8, 66, 87, 60, 0
[prvVdoIpcRxTask][1] 2, 8, 66, 87, 62, 0
[prvVdoIpcRxTask][1] 0, 6, 65, 86, 60, 0
[prvVdoIpcRxTask][1] 0, 6, 65, 86, 65, 0
Voc task audio read total(9830400)
[prvVdoIpcRxTask][1] 0, 4, 65, 89, 69, 0
[prvVdoIpcRxTask][1] 0, 5, 65, 84, 57, 0
[prvVdoIpcRxTask][1] 0, 7, 64, 82, 54, 0
[prvVdoIpcRxTask][1] 0, 5, 64, 85, 59, 0
[prvVdoIpcRxTask][1] 0, 4, 64, 79, 53, 0
[prvVdoIpcRxTask][1] 2, 5, 64, 72, 56, 0
[prvVdoIpcRxTask][1] 1, 3, 65, 72, 57, 0
[prvVdoIpcRxTask][1] 1, 4, 65, 74, 57, 0
[prvVdoIpcRxTask][1] 1, 3, 65, 72, 57, 0
[prvVdoIpcRxTask][1] 1, 5, 63, 72, 57, 0
[prvVdoIpcRxTask][1] 0, 5, 65, 85, 54, 0
[prvVdoIpcRxTask][1] 0, 5, 65, 90, 54, 0
```

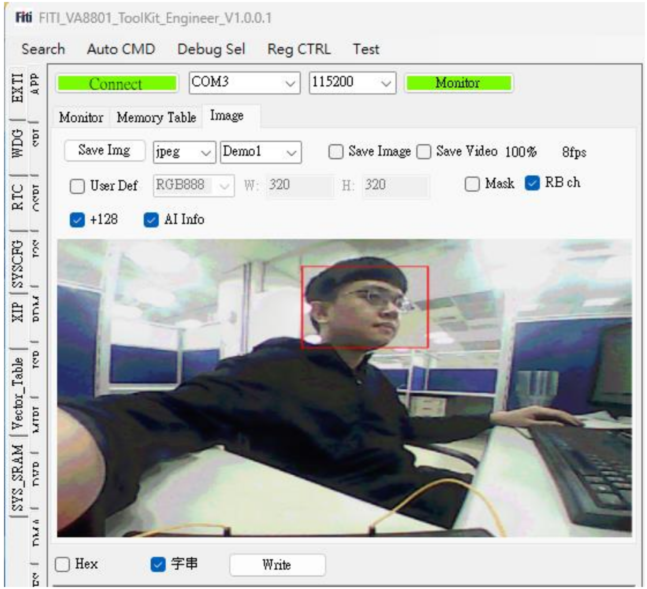
UART0 System log


COM5 - PuTTY

```
[prvVocHumanTask]tvmgen_default1_run - 4
result[0]: x1: 23, y1: 0, x2: 82, y2: 87, confidence: 74, class: 0
[prvVocHumanTask]post_process - 6
[prvVocHumanTask] IPC done, input addr: 0x6007dc00
[prvVocResnetTask] - 136
[prvVocHumanTask]tvmgen_default1_run - 5
result[0]: x1: 24, y1: 0, x2: 82, y2: 83, confidence: 69, class: 0
[prvVocHumanTask]post_process - 7
[prvVocHumanTask] IPC done, input addr: 0x6007dc00
[prvVocHumanTask]tvmgen_default1_run - 3
result[0]: x1: 25, y1: 0, x2: 83, y2: 84, confidence: 70, class: 0
[prvVocHumanTask]post_process - 7
[prvVocHumanTask] IPC done, input addr: 0x6007dc00
[prvVocHumanTask]tvmgen_default1_run - 4
result[0]: x1: 24, y1: 0, x2: 83, y2: 87, confidence: 72, class: 0
[prvVocHumanTask]post_process - 7
[prvVocHumanTask] IPC done, input addr: 0x6007dc00
[prvVocHumanTask]tvmgen_default1_run - 4
result[0]: x1: 24, y1: 0, x2: 82, y2: 84, confidence: 74, class: 0
[prvVocHumanTask]post_process - 7
[prvVocHumanTask] IPC done, input addr: 0x6007dc00
[prvVocHumanTask]tvmgen_default1_run - 4
result[0]: x1: 24, y1: 0, x2: 82, y2: 84, confidence: 72, class: 0
[prvVocHumanTask]post_process - 7
[prvVocHumanTask] IPC done, input addr: 0x6007dc00
[prvVocHumanTask]tvmgen_default1_run - 4
result[0]: x1: 24, y1: 0, x2: 82, y2: 84, confidence: 74, class: 0
[prvVocHumanTask]post_process - 7
```

UART1 NPU log

Flash Download procedure and system run – Scenario Face Detection



COM3 115200

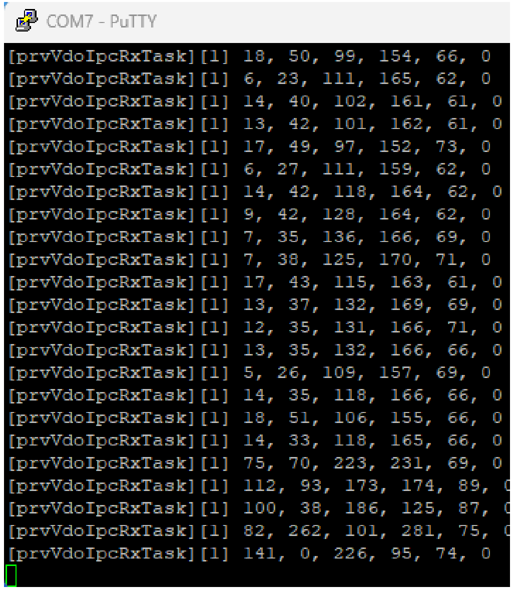
jpeg Demo1 □ Save Image □ Save Video 100% 8fps

□ User Def RGB888 W: 320 H: 320 □ Mask ☑ RB ch

☑ +128 ☑ AI Info

□ Hex ☑ 字串

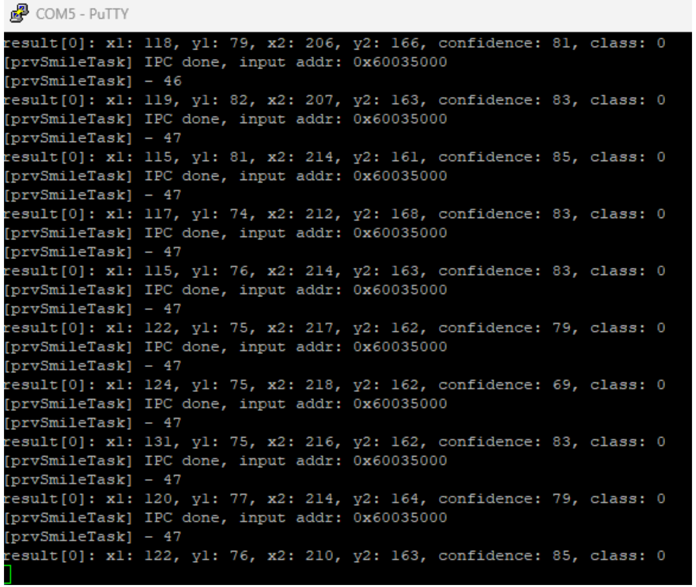
Demo Tool – Face Detection



COM7 - PuTTY

```
[prvVdoIpcRxTask][1] 18, 50, 99, 154, 66, 0
[prvVdoIpcRxTask][1] 6, 23, 111, 165, 62, 0
[prvVdoIpcRxTask][1] 14, 40, 102, 161, 61, 0
[prvVdoIpcRxTask][1] 13, 42, 101, 162, 61, 0
[prvVdoIpcRxTask][1] 17, 49, 97, 152, 73, 0
[prvVdoIpcRxTask][1] 6, 27, 111, 159, 62, 0
[prvVdoIpcRxTask][1] 14, 42, 118, 164, 62, 0
[prvVdoIpcRxTask][1] 9, 42, 128, 164, 62, 0
[prvVdoIpcRxTask][1] 7, 35, 136, 166, 69, 0
[prvVdoIpcRxTask][1] 7, 38, 125, 170, 71, 0
[prvVdoIpcRxTask][1] 17, 43, 115, 163, 61, 0
[prvVdoIpcRxTask][1] 13, 37, 132, 169, 69, 0
[prvVdoIpcRxTask][1] 12, 35, 131, 166, 71, 0
[prvVdoIpcRxTask][1] 13, 35, 132, 166, 66, 0
[prvVdoIpcRxTask][1] 5, 26, 109, 157, 69, 0
[prvVdoIpcRxTask][1] 14, 35, 118, 166, 66, 0
[prvVdoIpcRxTask][1] 18, 51, 106, 155, 66, 0
[prvVdoIpcRxTask][1] 14, 33, 118, 165, 66, 0
[prvVdoIpcRxTask][1] 75, 70, 223, 231, 69, 0
[prvVdoIpcRxTask][1] 112, 93, 173, 174, 89, 0
[prvVdoIpcRxTask][1] 100, 38, 186, 125, 87, 0
[prvVdoIpcRxTask][1] 82, 262, 101, 281, 75, 0
[prvVdoIpcRxTask][1] 141, 0, 226, 95, 74, 0
```

UART0 System log

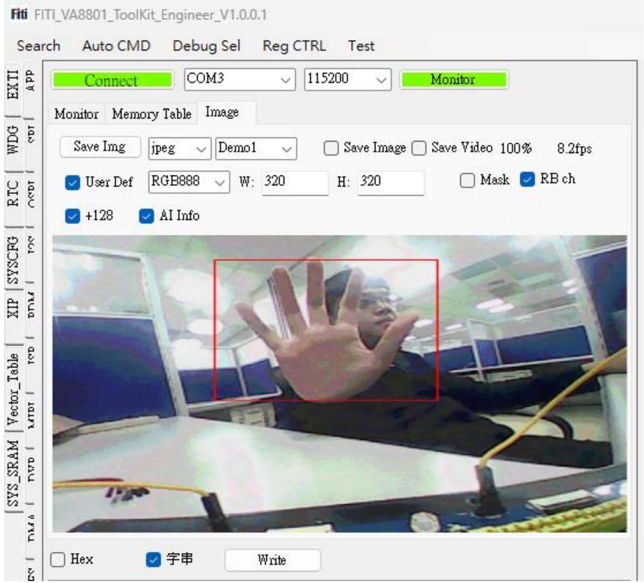


COM5 - PuTTY

```
result[0]: x1: 118, y1: 79, x2: 206, y2: 166, confidence: 81, class: 0
[prvSmileTask] IPC done, input addr: 0x60035000
[prvSmileTask] - 46
result[0]: x1: 119, y1: 82, x2: 207, y2: 163, confidence: 83, class: 0
[prvSmileTask] IPC done, input addr: 0x60035000
[prvSmileTask] - 47
result[0]: x1: 115, y1: 81, x2: 214, y2: 161, confidence: 85, class: 0
[prvSmileTask] IPC done, input addr: 0x60035000
[prvSmileTask] - 47
result[0]: x1: 117, y1: 74, x2: 212, y2: 168, confidence: 83, class: 0
[prvSmileTask] IPC done, input addr: 0x60035000
[prvSmileTask] - 47
result[0]: x1: 115, y1: 76, x2: 214, y2: 163, confidence: 83, class: 0
[prvSmileTask] IPC done, input addr: 0x60035000
[prvSmileTask] - 47
result[0]: x1: 122, y1: 75, x2: 217, y2: 162, confidence: 79, class: 0
[prvSmileTask] IPC done, input addr: 0x60035000
[prvSmileTask] - 47
result[0]: x1: 124, y1: 75, x2: 218, y2: 162, confidence: 69, class: 0
[prvSmileTask] IPC done, input addr: 0x60035000
[prvSmileTask] - 47
result[0]: x1: 131, y1: 75, x2: 216, y2: 162, confidence: 83, class: 0
[prvSmileTask] IPC done, input addr: 0x60035000
[prvSmileTask] - 47
result[0]: x1: 120, y1: 77, x2: 214, y2: 164, confidence: 79, class: 0
[prvSmileTask] IPC done, input addr: 0x60035000
[prvSmileTask] - 47
result[0]: x1: 122, y1: 76, x2: 210, y2: 163, confidence: 85, class: 0
```

UART1 NPU log

Flash Download procedure and system run – Scenario Gesture



FITI_VA8801_ToolKit_Engineer_V1.0.0.1

COM3 115200

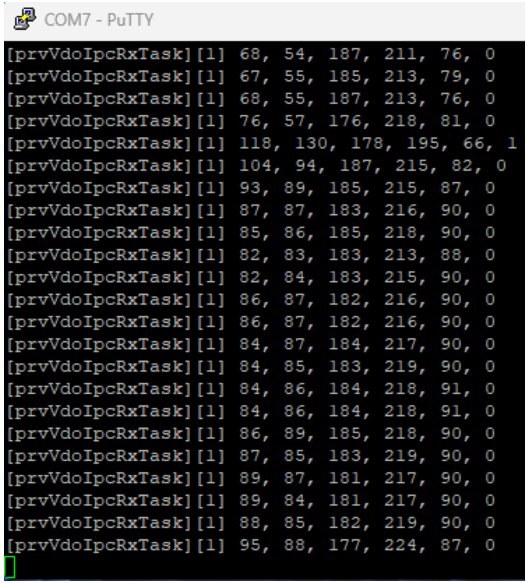
jpeg Demo1 □ Save Image □ Save Video 100% 8.2fps

☑ User Def RGB888 W: 320 H: 320 □ Mask ☑ RB ch

☑ +128 ☑ AI Info

□ Hex ☑ 字串

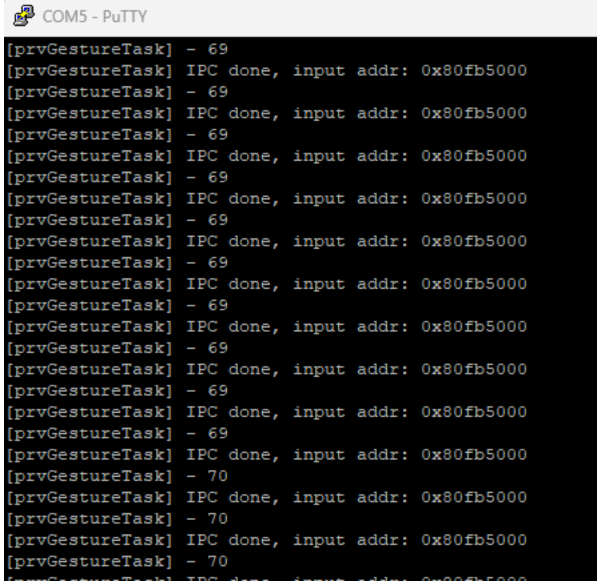
Demo Tool – Gesture



COM7 - PuTTY

```
[prvVdoIpcRxTask][1] 68, 54, 187, 211, 76, 0
[prvVdoIpcRxTask][1] 67, 55, 185, 213, 79, 0
[prvVdoIpcRxTask][1] 68, 55, 187, 213, 76, 0
[prvVdoIpcRxTask][1] 76, 57, 176, 218, 81, 0
[prvVdoIpcRxTask][1] 118, 130, 178, 195, 66, 1
[prvVdoIpcRxTask][1] 104, 94, 187, 215, 82, 0
[prvVdoIpcRxTask][1] 93, 89, 185, 215, 87, 0
[prvVdoIpcRxTask][1] 87, 87, 183, 216, 90, 0
[prvVdoIpcRxTask][1] 85, 86, 185, 218, 90, 0
[prvVdoIpcRxTask][1] 82, 83, 183, 213, 88, 0
[prvVdoIpcRxTask][1] 82, 84, 183, 215, 90, 0
[prvVdoIpcRxTask][1] 86, 87, 182, 216, 90, 0
[prvVdoIpcRxTask][1] 86, 87, 182, 216, 90, 0
[prvVdoIpcRxTask][1] 84, 87, 184, 217, 90, 0
[prvVdoIpcRxTask][1] 84, 85, 183, 219, 90, 0
[prvVdoIpcRxTask][1] 84, 86, 184, 218, 91, 0
[prvVdoIpcRxTask][1] 84, 86, 184, 218, 91, 0
[prvVdoIpcRxTask][1] 86, 89, 185, 218, 90, 0
[prvVdoIpcRxTask][1] 87, 85, 183, 219, 90, 0
[prvVdoIpcRxTask][1] 89, 87, 181, 217, 90, 0
[prvVdoIpcRxTask][1] 89, 84, 181, 217, 90, 0
[prvVdoIpcRxTask][1] 88, 85, 182, 219, 90, 0
[prvVdoIpcRxTask][1] 95, 88, 177, 224, 87, 0
```

UART0 System log



COM5 - PuTTY

```
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 69
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 70
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 70
[prvGestureTask] IPC done, input addr: 0x80fb5000
[prvGestureTask] - 70
```

UART1 NPU log

REVISION HISTORY

Revision	Date	Author	Description
0.1	2024/02/26	Jason SYU	New issued
2.0	2024/05/24	Jason SYU	1. Add Build NPU Project guide 2. Add How to Obtain pre/post process and inference time in NPU Project 3. Add Flash Download procedure and system run

THANKS