



CYPRESS SEMICONDUCTOR CORPORATION

Internal Correspondence

Date: 12/4/2019 **WW:** 1949
To: WHAO
Author: CHMA
Author File#: CHMA#44*B
Subject: MTB2.0 Dual-Core Support Trial – Create Project via Template
Category:
Distribution: WHAO; XITO; JRTI; ZQLI; MIQI; CCTA; KOZU; HAXI; LWAX

Summary

The memo documents the detailed steps of how to create a dual-core supported application project for specific MPN via the template on MTB2.0.

Feature

- Add a component.mk file in CM4 to share CM0p's Device Configurator tool
- Dual-Core debugging is supported
- Auto-generate the merged final.hex and final.elf files
- Add a variable in CM0p project's makefile to implement MPN auto-switching
- Fixed EEPROM/SMIF etc. limitation on CM0p core
- Extract PDL to public path

Prerequisites

ModusToolbox 2.0: Build ID#1703

<https://www.cypress.com/products/modustoolbox-software-environment>

Template: V1.1.0

https://github.com/emyhello/mtb2.0/blob/master/dual-core_template/Template_V110.zip

Template Introduction

- 1) Download the template from the above URL.

Note: this template is based on CY8CKIT-062-BLE (CY8C6347BZI-BLD53) kit.





mtb2.0/Template_V110.zip at master · emyhelo · GitHub

Join GitHub today
GitHub is home to over 40 million developers working together to host and review code, manage projects, and build software together.
[Sign up](#)

Branch: master ▾ mtb2.0 / dual-core_template / Template_V110.zip [Find file](#) [Copy path](#)

emyhelo Update Template_V110.zip 58b1737 11 minutes ago
1 contributor

32.3 MB [Download](#) [History](#)  

- 2) Un-compressing the template to your workspace (**the path cannot contain “Space”**), the structure of each one as following,
Template_V110

Share View

Template_V110

	Name	Date modified	Type
ess	Cm0p_App	11/28/2019 3:59 PM	File folder
,	Cm4_App	11/28/2019 3:59 PM	File folder
ads	Custom_Tool	12/4/2019 2:03 PM	File folder
nts	Cy_Pdl	11/19/2019 10:35 AM	File folder

Cm0p folder: CM0p core's project

Cm4 folder: CM4 core's project

Custom_Tool folder: It consists of batch files, which are used to auto-switch MPN and generate whole MCU image files.

Cy_Pdl: PDL, middleware and lib files etc. provided by cypress

Details

1. Import the template to MTB2.0 IDE



- 1) Launch MTB2.0 and set the workspace to previous folder

ModusToolbox IDE Launcher



Select a directory as workspace

ModusToolbox IDE uses the workspace directory to store its preferences and development artifacts.

Workspace: C:\Users\chma\Desktop\Template_V110

Browse...

☐ Use this as the default and do not ask again

Recent Workspaces

Launch Cancel

- 2) Right-Click on the blank space in “Project Explorer”, and click “Import...” Option, then select “General->Existing Projects into Workspace” as following,

1 Right-click here

2

3

Import Projects

Select a directory to search for existing Eclipse projects.

☒ Select root directory: C:\Users\chma\Desktop\Template_V110

☐ Select archive file:

Projects:

- ☒ Cm0p_App (C:\Users\chma\Desktop\Template_V110\Cm0p_App)
- ☒ Cm4_App (C:\Users\chma\Desktop\Template_V110\Cm4_App)

Options

- ☐ Search for nested projects
- ☐ Copy projects into workspace
- ☐ Close newly imported projects upon completion
- ☐ Hide projects that already exist in the workspace

Working sets

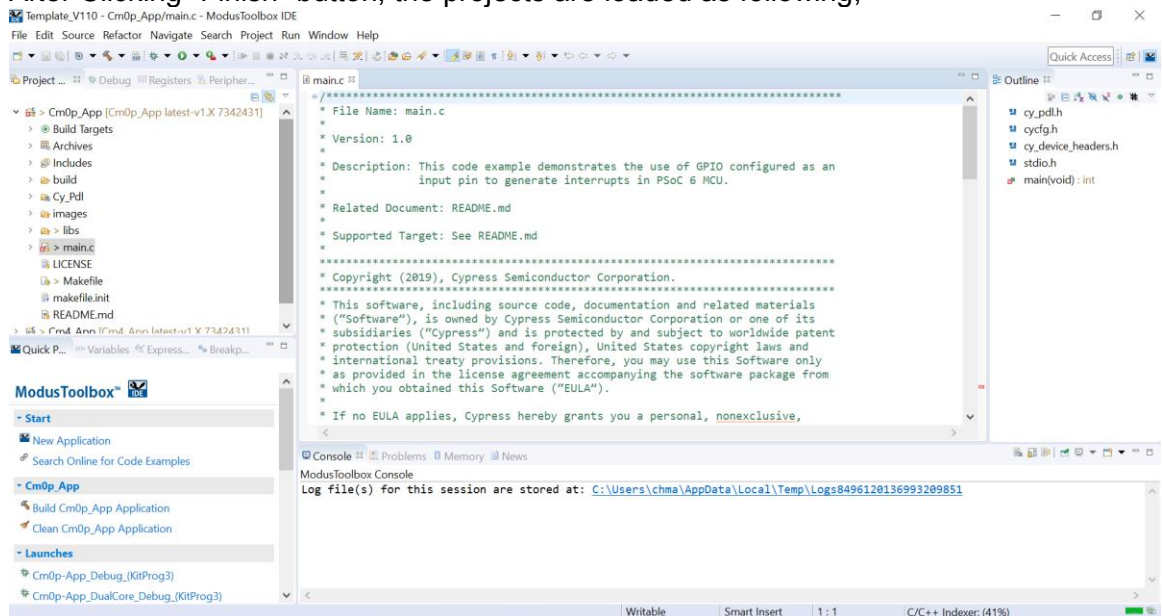
☐ Add project to working sets

Working sets:

Finish

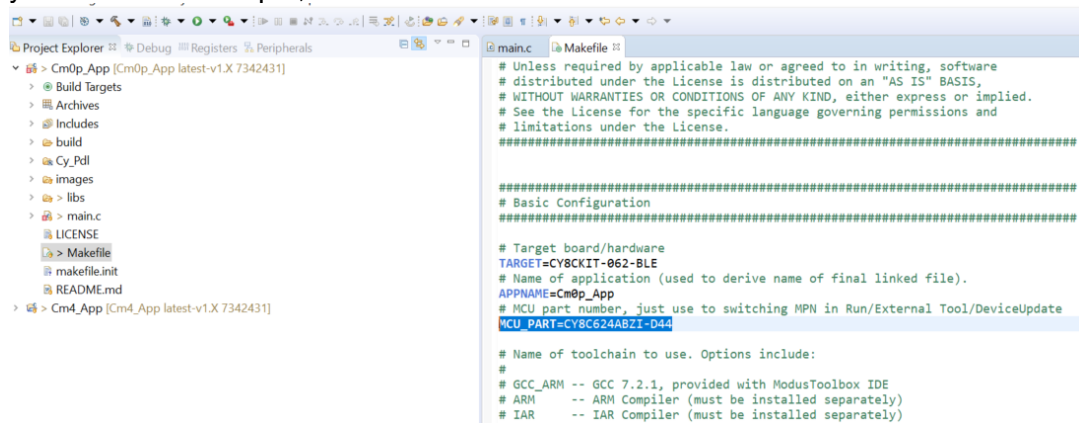


3) After Clicking “Finish” button, the projects are loaded as following,

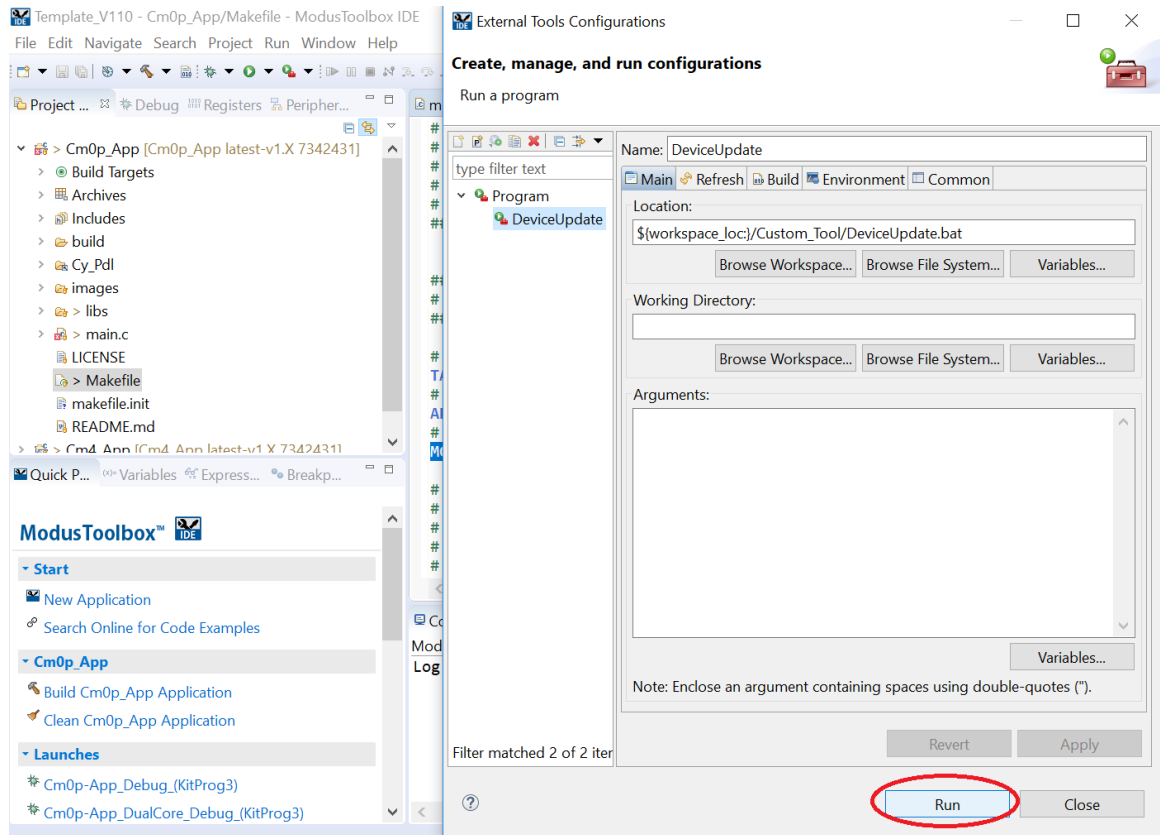


2. Switch the MCU part number

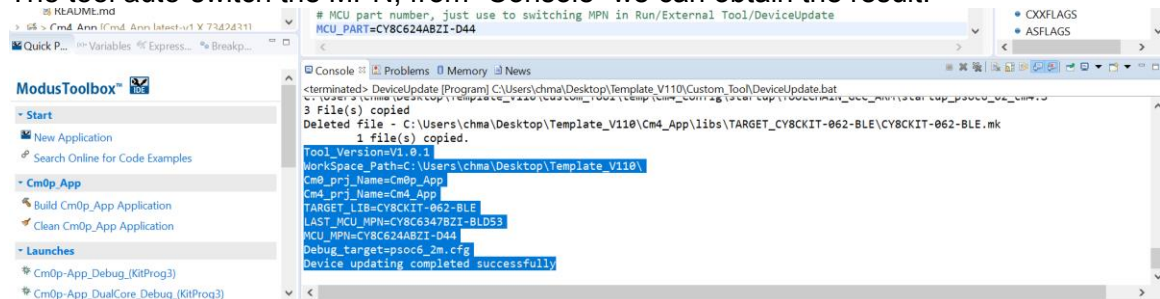
1) Open the makefile in CM0p project, modify the value of “MCU_PART” as your like. For example, switch to CY8C624ABZI-D44.



2) Select the Run->External Tools->External Tools Configurations on IDE menu, then run “DeviceUpdate” tool.

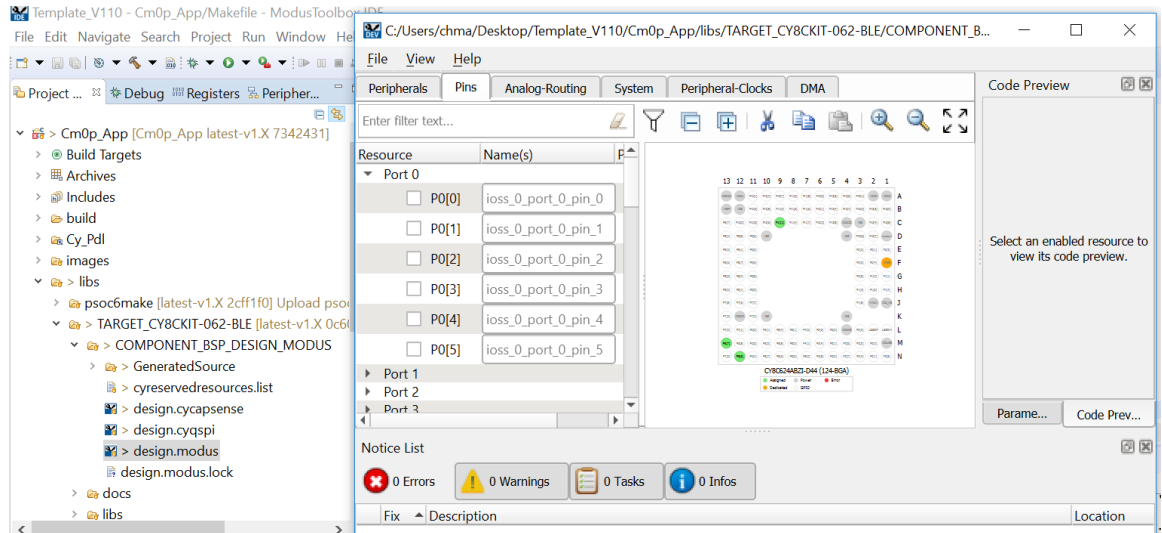


3) The tool auto-switch the MPN, from “Console” we can obtain the result.

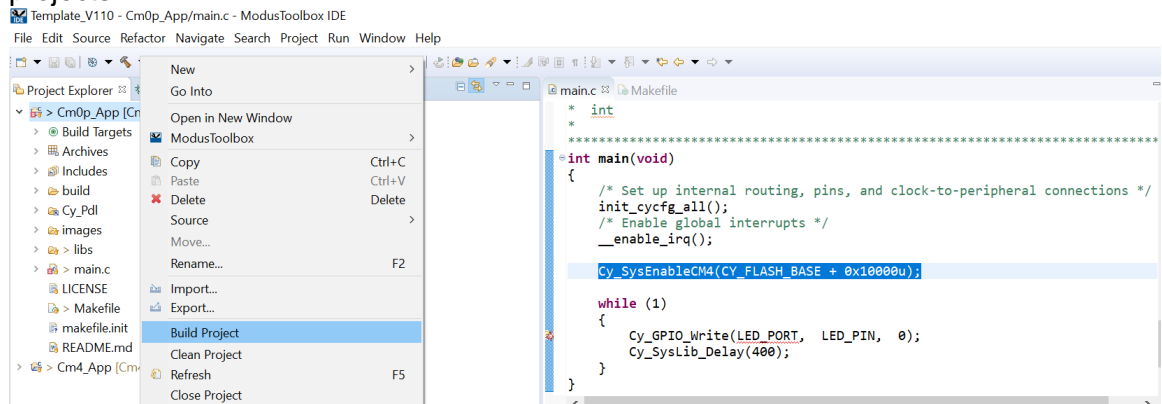


3. Build the projects

- 1) Open the design.modus to configure the Pin/Device functions then save configuration.



- 2) Right-click project folder to select “Build Project” to build CM0p and CM4 projects.



- 3) After building CM4 project, a script will auto-run to merge CM0p and CM4 project's elf to generate whole MCU image file. A reminder is displayed on “Console” window.



```

template_v110 - Cm0p_App/main.c - m04us1000box i2c
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer | Debug | Registers | Peripherals | Makefile
main.c
int
int main(void)
{
    /* Set up internal routing, pins, and clock-to-peripheral connections */
    init_cyfg_all();
    /* Enable global interrupts */
    __enable_irq();
}

Console | Problems | Memory | News
Cm4 Build Console [Cm4_App]
Could Not Find C:\Users\chma\Desktop\Template_V110\Cm4_App\build\CY8CKIT-062-BLE\Debug\final.elf
No ELF section .cychecksum found, creating one
Application checksum calculated and stored in ELF section .cychecksum
No ELF section .cymeta found, creating one
Checksum calculated and stored in ELF section .cymeta
Tool_VersionsV1.0.1
Cm0_prj_Name=Cm0p_App
Cm4_prj_Name=Cm4_App
Workspace_Path=C:\Users\chma\Desktop\Template_V110\
Cm0_TARGET_LIB=Cy8CKIT-062-BLE
Cm4_TARGET_LIB=Cy8CKIT-062-BLE
Cm4_APP_NAME=Cm4_App
ErrCode=000
Merge files were generated successfully!
The final.elf and final.hex files are generated in C:\Users\chma\Desktop\Template_V110\Cm4_App\build\CY8CKIT-062-BLE\Debug\
=====
Build complete =
=====
Calculating memory consumption: CY8C624ABZI-D44 GCC_ARM -Og

| Section Name | Address | Size |
|---|---|---|
| .text | 0x10010000 | 4384 |
| .copy.table | 0x10011120 | 24 |
| .zero.table | 0x10011138 | 8 |
| .data | 0x080082e0 | 1400 |
| .cy_sharedmem | 0x08008858 | 8 |
| .noinit | 0x08008860 | 148 |
| .bss | 0x080088f4 | 480 |
| .heap | 0x08008ad8 | 945448 |
|---|---|---|
Total Internal Flash (Available) | 2097152 |
Total Internal Flash (Utilized) | 5824 |
Total Internal SRAM (Available) | 1046528 |
Total Internal SRAM (Utilized) | 947484 |

15:19:35 Build Finished. 0 errors, 0 warnings. (took 59s.594ms)

```

CM0p build output: \Cm0p_App\build\CY8CKIT-062-BLE\Debug
 CM4 build output: \Cm4_App\build\CY8CKIT-062-BLE\Debug
 Final merged files: \Cm4_App\build\CY8CKIT-062-BLE\Debug

4. Debug Projects

- 1) Run->Debug Configuration in IDE menu, you can debugging/programming MCU here.

Note: the "Final_Program_(KitProg3)" will program whole MCU both CM0p and CM4 project.

Template_V110 - Cm0p_App/main.c | Debug Configurations

Create, manage, and run configurations

Name: Cm0p-App_Attach_(KitProg3)

Project: Cm0p_App

C/C++ Application: build\CY8CKIT-062-BLE\Debug\Cm0p_App.elf

Build (if required) before launching

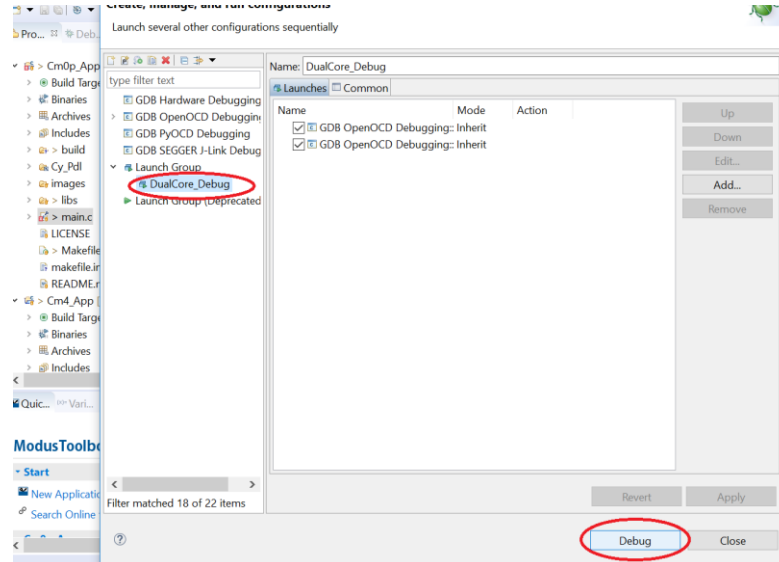
Build Configuration: Select Automatically

☐ Enable auto build

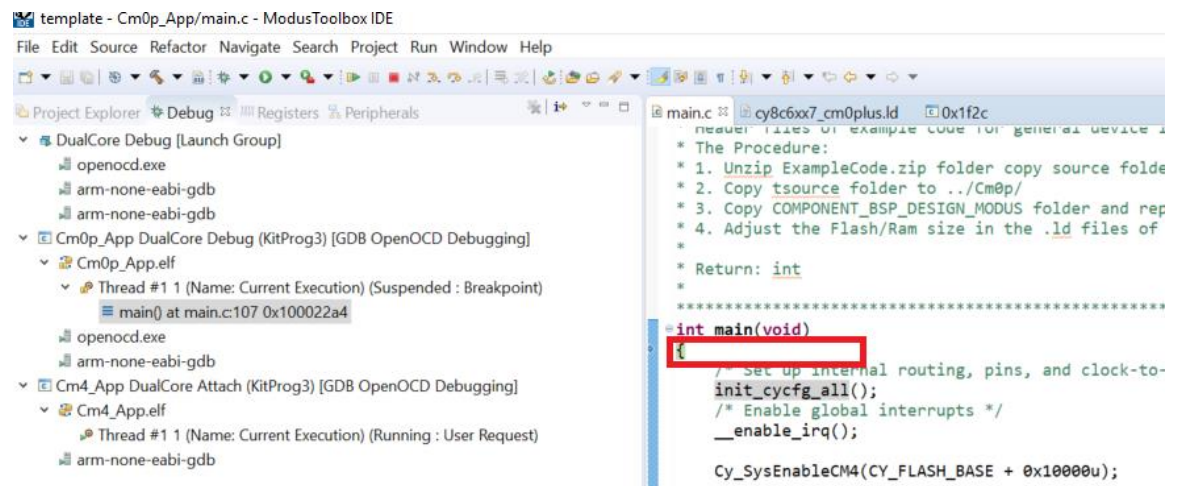
☒ Use workspace settings



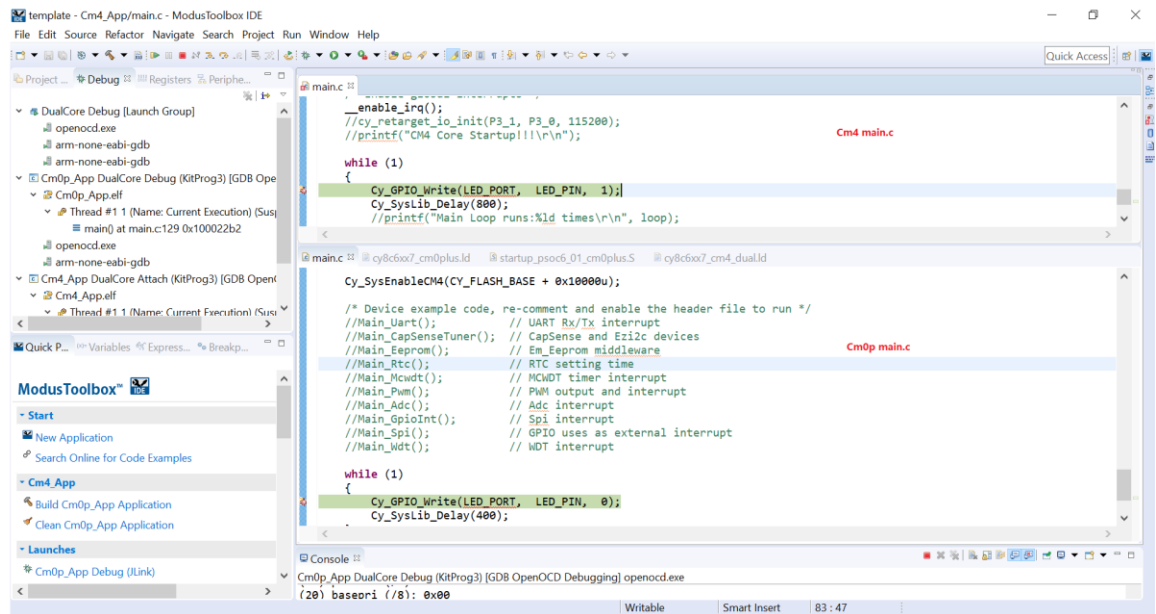
- 2) “DualCore_Debug” option is under “Launch Group”, which is used to dual core debugging



- a. Selected Cm4_App project, then launch the “DualCore_Debug (KitProg3)”.



- b. Debugging have jumped to Cm0p main(). After running to “Cy_SysEnableCM4(CY_FLASH_BASE + 0x10000u);”, we can debug the projects separately.



Attention

1. The path of MTB project cannot contain “space”.
2. You can adjust the Flash/Ram size of CM0p and CM4 core via the linker file in /libs/TARGET_CY8CKIT-062-BLE/linker/TOOLCHAIN_GCC_ARM/ folder.
3. Make sure the address of following function in your CM0p main.c matches the flash assignment.
`Cy_SysEnableCM4(CY_FLASH_BASE + 0x10000u);`
4. Before starting the debugging, please select the corresponding project.

Troubleshooting

1. Building error, remind “***region 'flash' overflowed by 6360 bytes***”.
A: The flash size for project is not enough, modify the linker file and try again.
2. Building error, remind “***303 cannot move location counter backwards (from 08000bf0 to 08000000)***”.
A: The ram size for project is not enough, modify the linker file and try again.
3. Linking error after building, remind “***../xxx/xxx.o: No such file or directory***”
A: Be sure the path does not contain any “Space”.
4. Building error, remind “***'cy_capsense_tuner' undeclared (first use in this function); did you mean 'Main_CapSenseTuner'? etc.*** when using CapSense on Cm0p core?



A: Two ways to fix this issue, in ../lib/TARGET_XXX/

COMPONENT_BSP_DESIGN_MODUS/ folder

- 1) Modify the macro **#define** CY_CAPSENSE_CORE 0u) in / GneratedSource /cycfg_peripherals.h file by manual when updating the device configurator every time
- 2) Open the design.modus by editor then modify: <Param id="CapSenseCore" value="0"/>

Reference

1. LWAX-6: Kaadas MTB 2.0 dual-core on-site support

END OF MEMO