

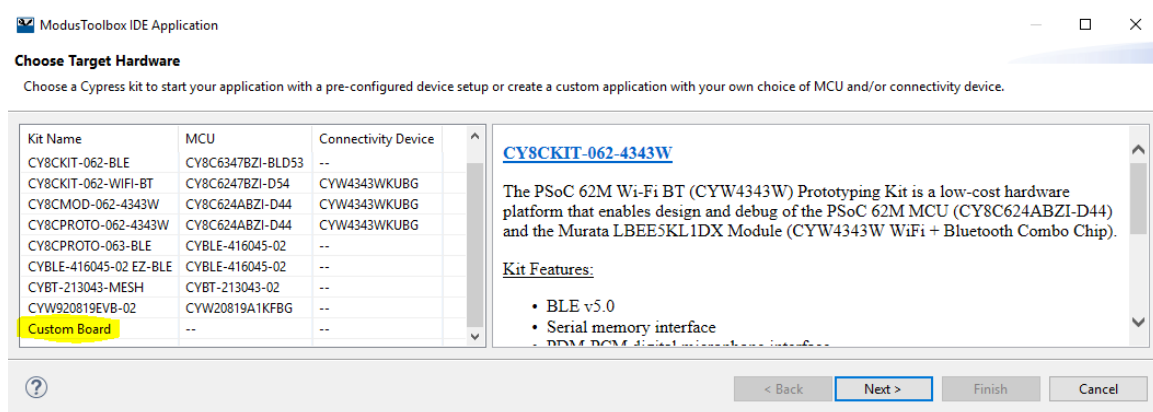
# This document demonstrates how to convert your Modus Toolbox 1.0 application too a Modus Toolbox 1.1 application

By Jelle Huiberts.

First of all create a new Modus Toolbox application

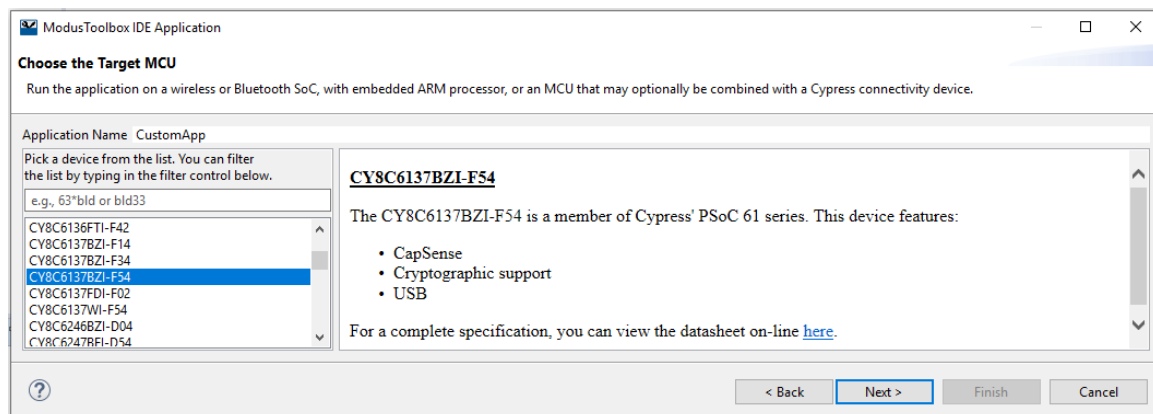


Select custom board

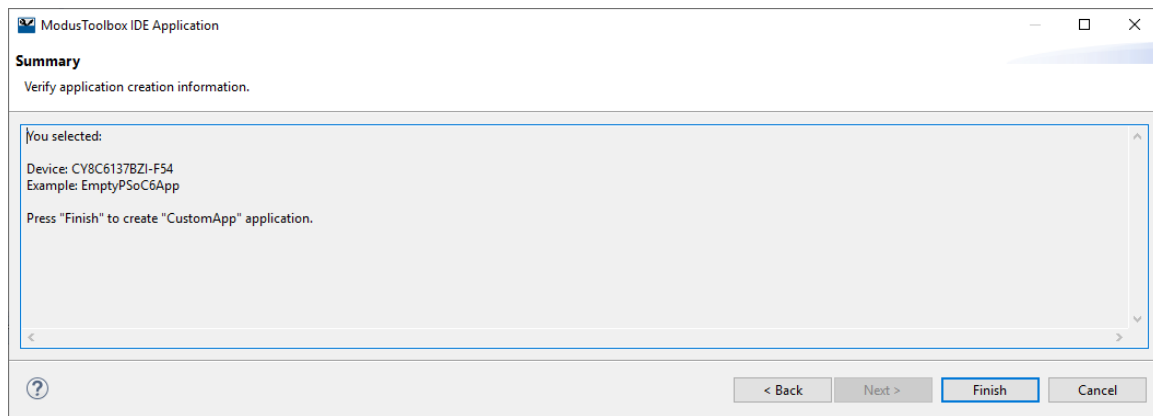


Click next

Select CY8C6137BZI-F54

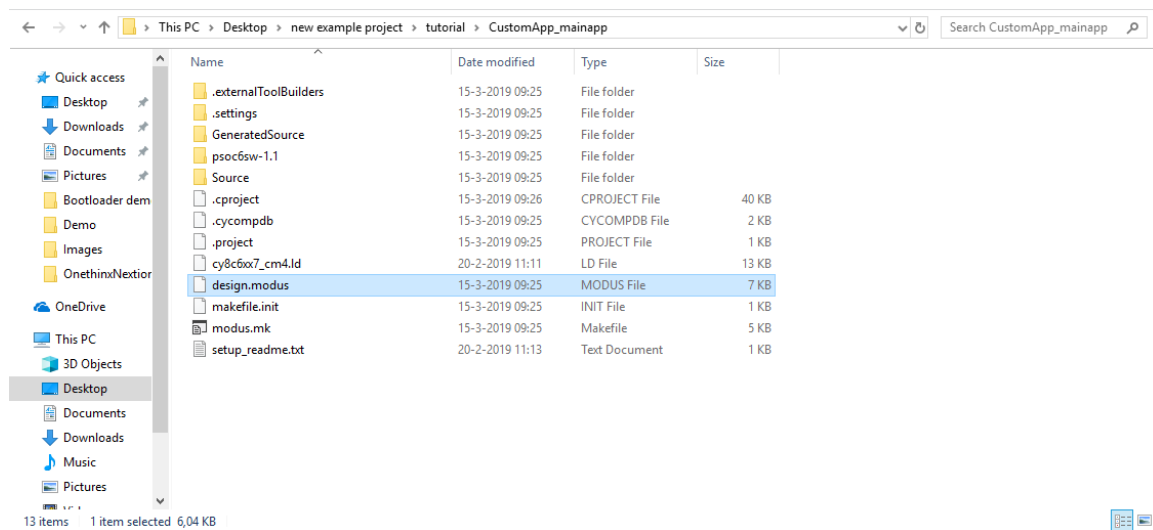


Click next

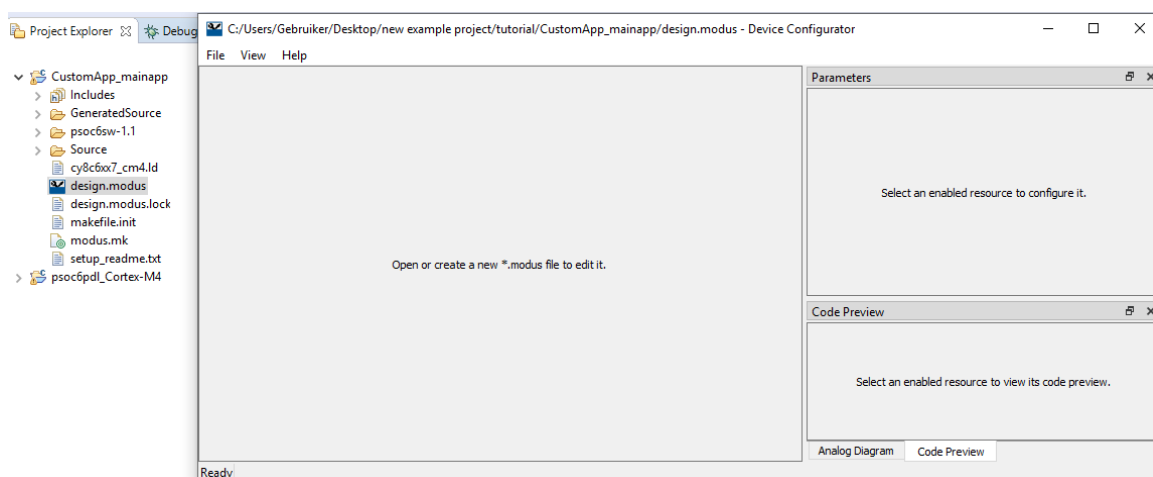


And finish

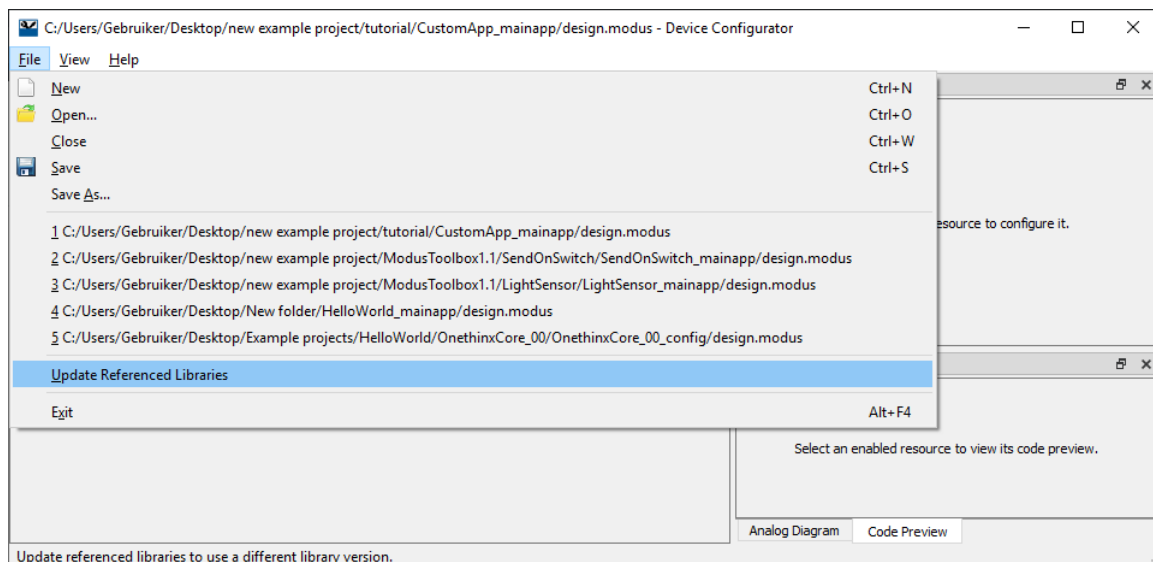
Replace the design.modus file in your \_mainapp folder with the design.modus file from your Modus Toolbox 1.0 project



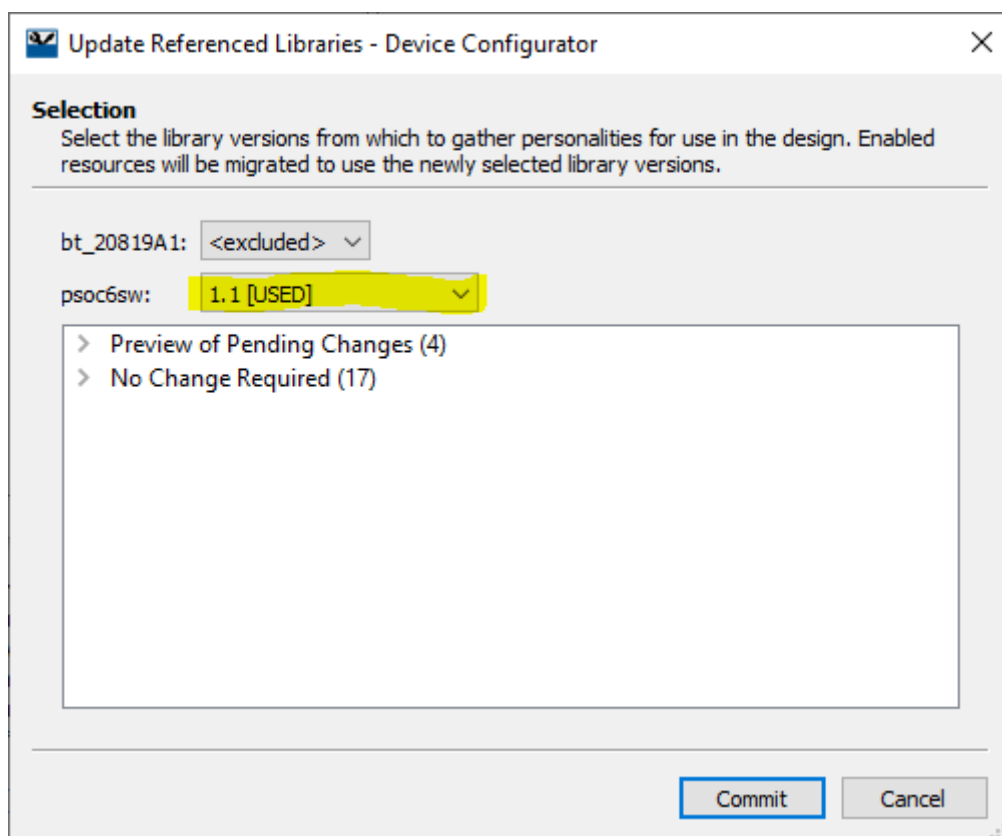
Open your design.modus file you should see the device configurator



Hover file and go too Update Referenced Libraries



Change psoc6sw from 1.0 to 1.1

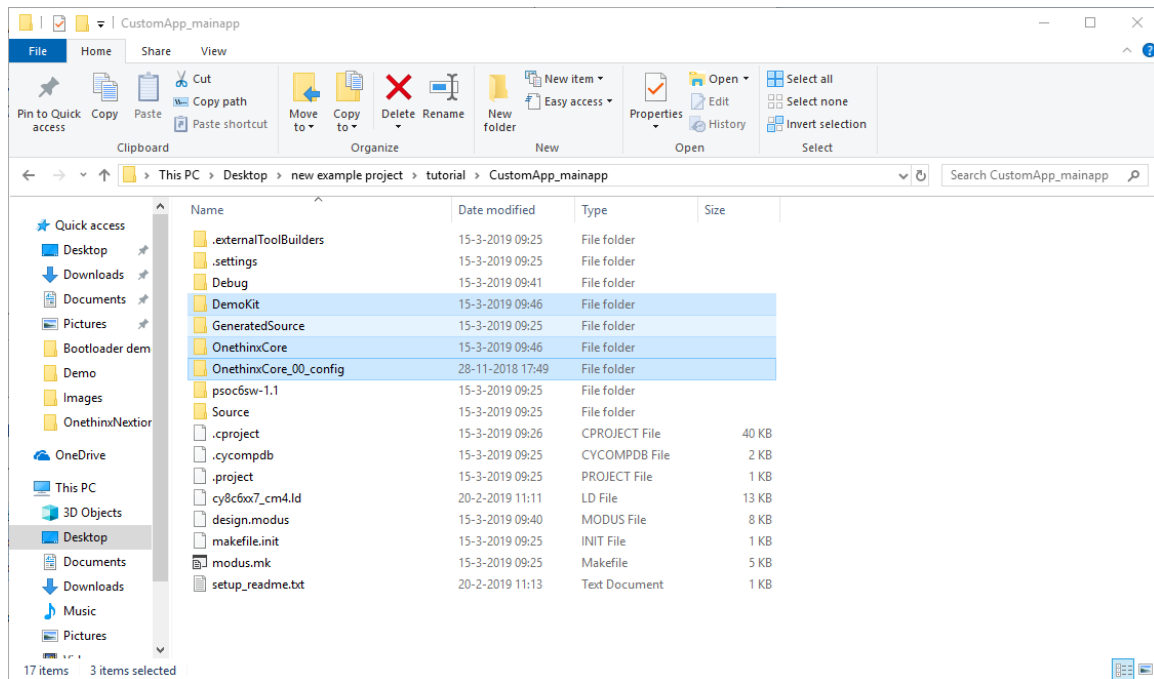


Commit and finish

Next copy these folders:

DemoKit	14-12-2018 12:05	File folder
OnethinxCore	17-12-2018 16:13	File folder
OnethinxCore_00_config	28-11-2018 17:49	File folder

over from your Modus Toolbox 1.0 project too your new one.



Next go too your projects build settings

## ModusToolbox™ IDE

### Start

- New Application
- Search Online for Code Examples

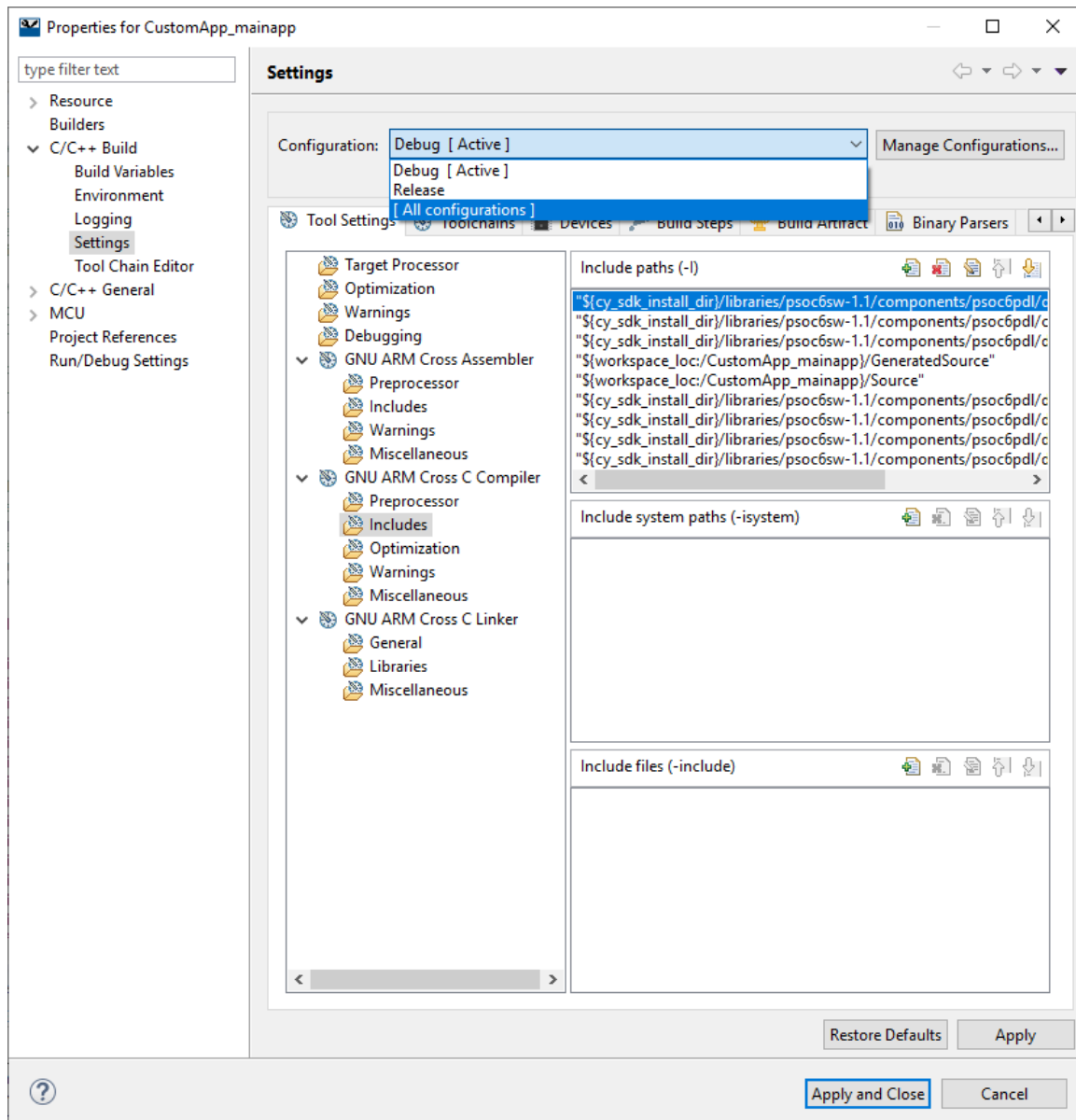
### CustomApp\_mainapp

- Build CustomApp Application
- Clean CustomApp Application
- Project Build Settings
- Configure Device
- Select Middleware

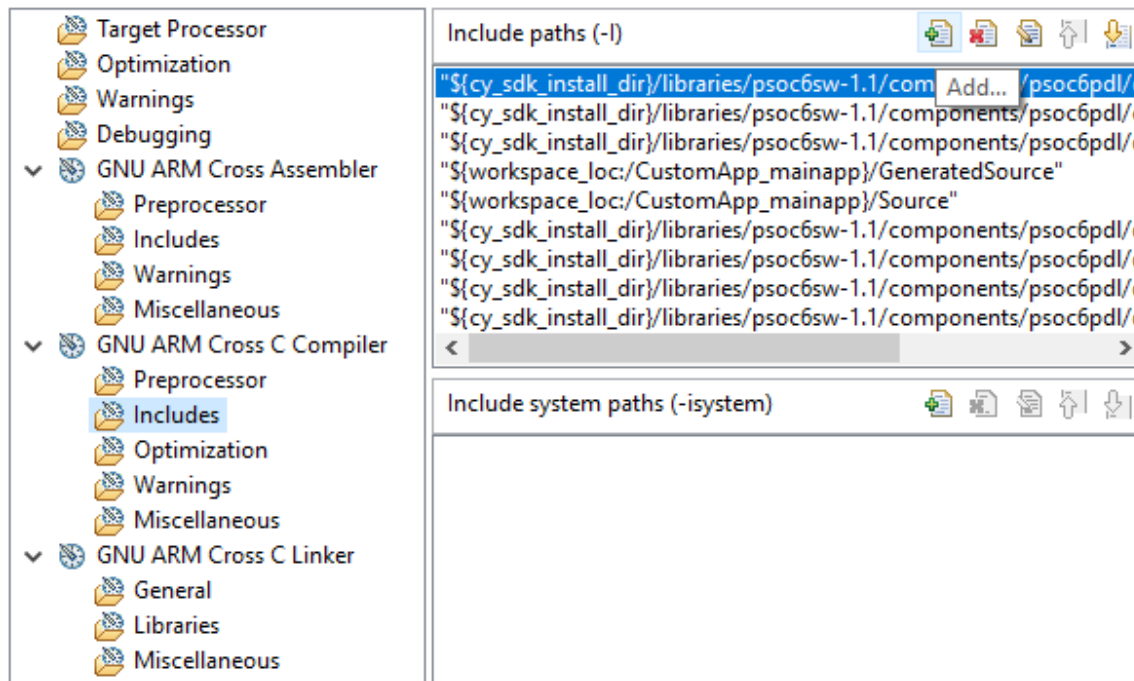
### Launches

- CustomApp Debug (J-Link)
- CustomApp Debug (KitProg3)
- CustomApp Program (J-Link)
- CustomApp Program (KitProg3)

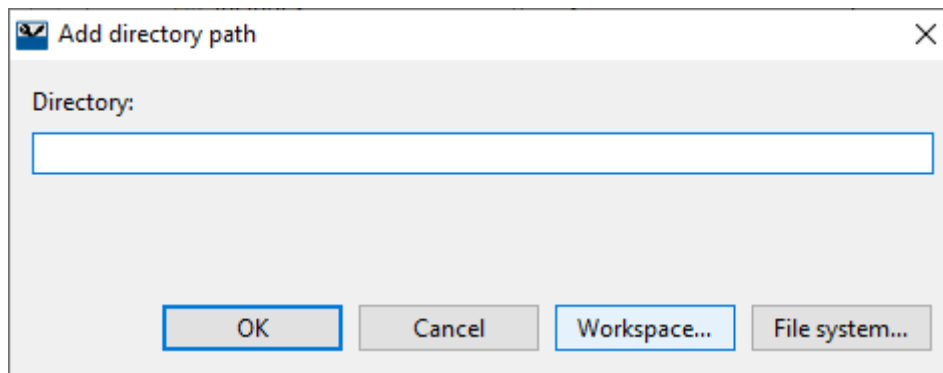
Select all configurations



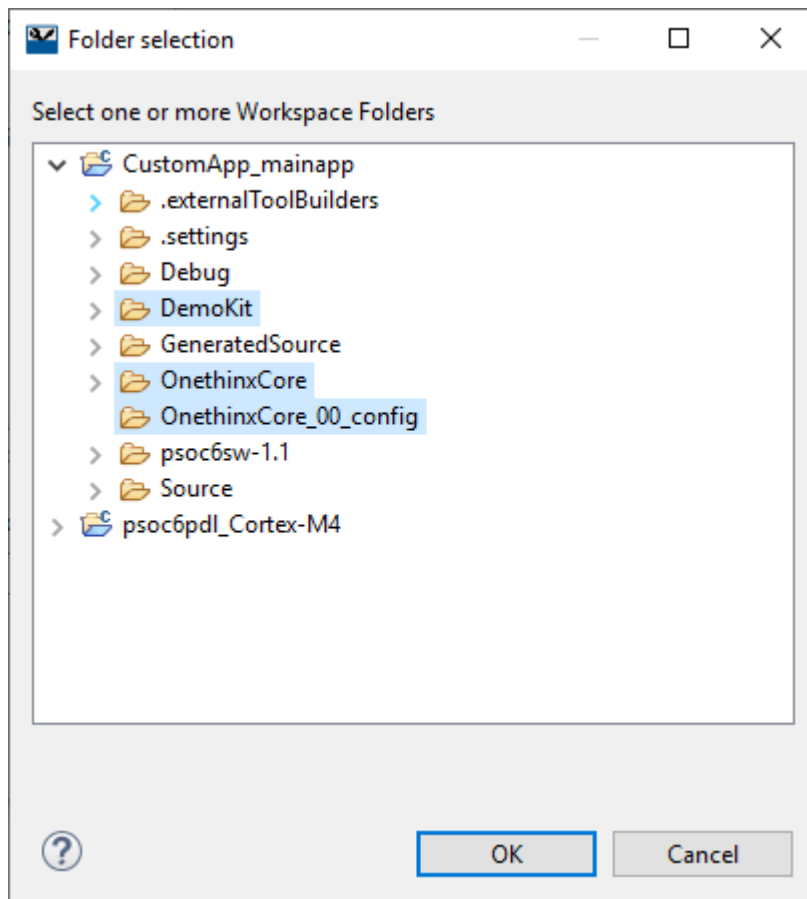
Click on add by GNU ARM Cross C Compiler > Includes



Go too workspace

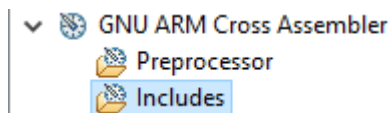


And select these folders

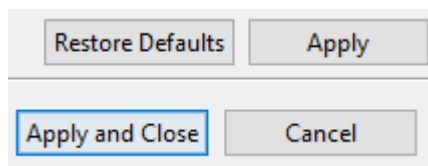


Click OK

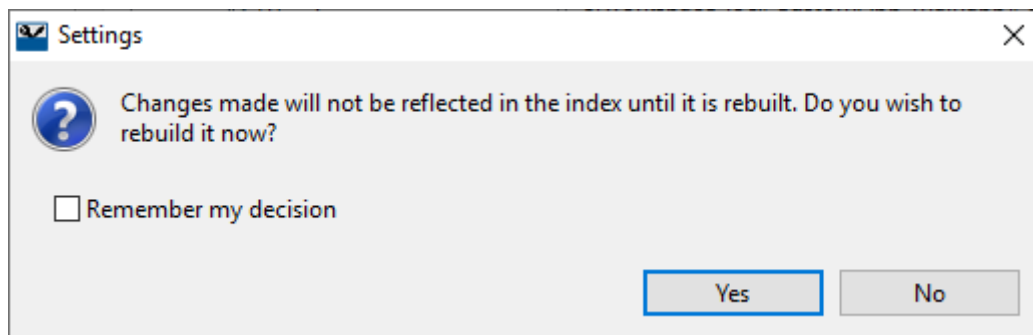
Do the same for the GNU ARM Cross Assembler > Includes



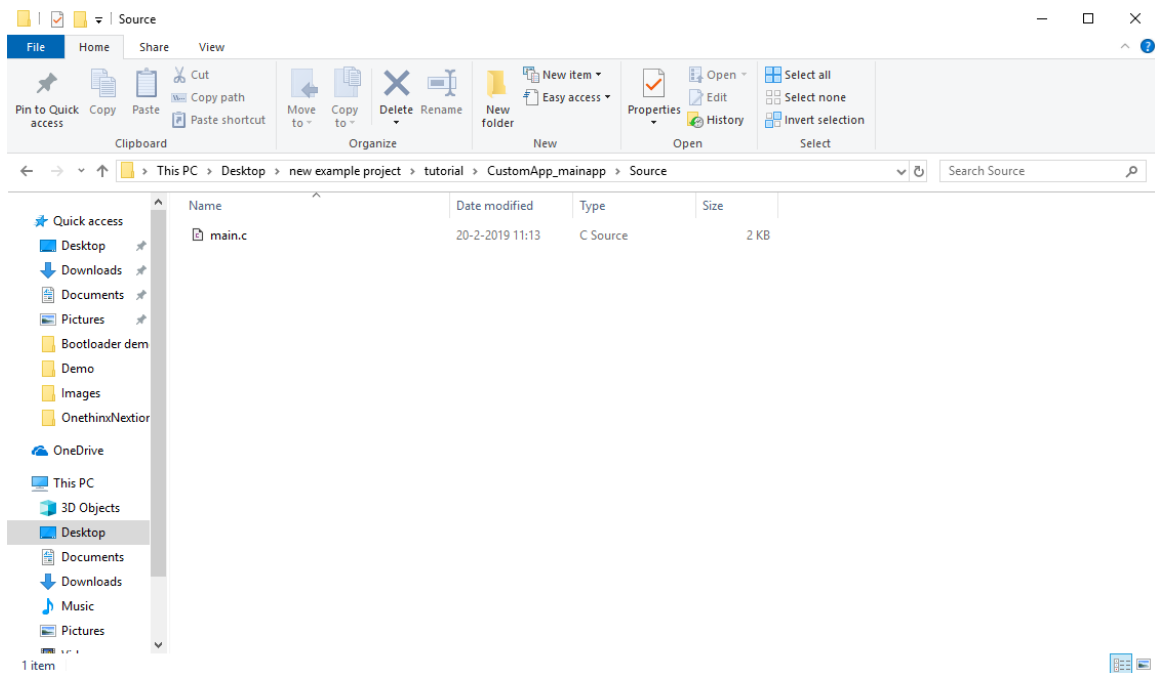
Apply and Close



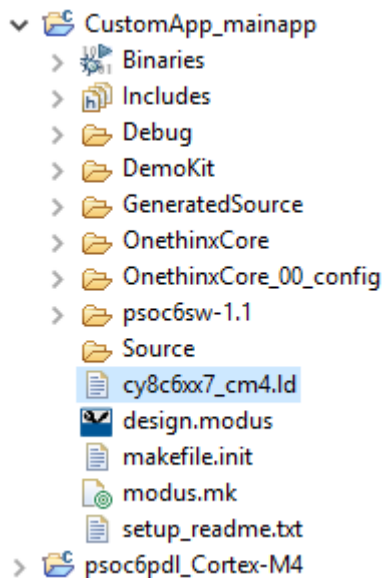
Click Yes



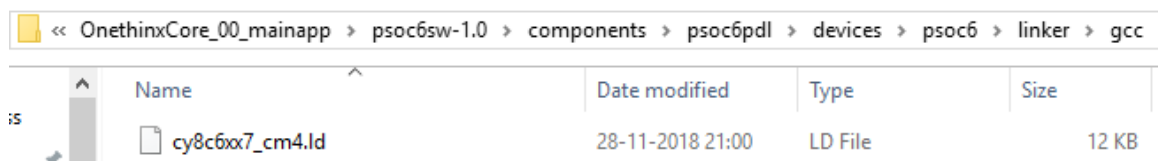
Next replace the main.c file in your source folder with the one from your Modus Toolbox 1.0 application



Next replace cy8c6xx7\_cm4.ld with the one from your Modus Toolbox 1.0 application



You can find it here: \psoc6sw-1.0\components\psoc6pdl\devices\psoc6\linker\gcc












First clean then build your application

## ModusToolbox™





### ▼ Start

-  New Application
-  Search Online for Code Examples
























### ▼ CustomApp\_mainapp

-  Build CustomApp Application
-  Clean CustomApp Application
-  Project Build Settings
-  Configure Device
-  Select Middleware

### ▼ Launches

-  CustomApp Debug (J-Link)
-  CustomApp Debug (KitProg3)
-  CustomApp Program (J-Link)
-  CustomApp Program (KitProg3)

You should get an error in cycfg\_platform.c

- ▼  CustomApp\_mainapp
  - >  Includes
  - >  Debug
  - >  DemoKit
  - ▼  GeneratedSource
    - >  cycfg\_notices.h
    - >  cycfg\_pins.c
    - >  cycfg\_pins.h
    - >  cycfg\_platform.c
    - >  cycfg\_platform.h
    - >  cycfg\_routing.c
    - >  cycfg\_routing.h
    - >  cycfg.c
    - >  cycfg.h
  - >  OnethinxCore
  - >  OnethinxCore\_00\_config
  - >  psoc6sw-1.1
  - ▼  Source
    - >  main.c
    -  cy8c6xx7\_cm4.ld
    -  design.modus
    -  makefile.init
    -  modus.mk

This is not intended (we are working on a fix!) for now here is a hotfix!

Comment these lines out

```
__STATIC_INLINE void Cy_SysClk_AlthfInit()
{
    // cy_en_ble_eco_status_t status = Cy_BLE_EcoConfigure(CY_BLE_BLESS_ECO_FREQ_32MHZ, CY_BLE_SYS_ECO_CLK_DIV_4, 22U, 25U, CY_BLE_ECO_VOLTAGE_REG_AUTO);
    // if ((CY_BLE_ECO_SUCCESS != status) && (CY_BLE_ECO_ALREADY_STARTED != status))
    // {
    //     cycfg_ClockStartupError(CY_CFG_SYSCLK_ALTHF_ERROR);
    // }
}
```

And change this function

```
void init_cycfg_platform(void)
{
    /* Set worst case memory wait states (! ultra low power, 150 MHz), will update at the end */
    Cy_SysLib_SetWaitStates(false, 150UL);
    #if (CY_CFG_PWR_VBAC_SUPPLY == CY_CFG_PWR_VBAC_SUPPLY_VDD)
    if (0u == Cy_SysLib_GetResetReason() /* POR, XRES, or BOD */)
    {
        Cy_SysLib_ResetBackupDomain();
        Cy_SysClk_IloDisable();
        Cy_SysClk_IloInit();
    }
}
```

too CySysClk\_Wcolnit();

```
void init_cycfg_platform(void)
{
    /* Set worst case memory wait states (! ultra low power, 150 MHz), will update at the end */
    Cy_SysLib_SetWaitStates(false, 150UL);
    #if (CY_CFG_PWR_VBAC_SUPPLY == CY_CFG_PWR_VBAC_SUPPLY_VDD)
    if (0u == Cy_SysLib_GetResetReason() /* POR, XRES, or BOD */)
    {
        Cy_SysLib_ResetBackupDomain();
        Cy_SysClk_IloDisable();
        Cy_SysClk_WcoInit();
    }
}
```

Click Debug (KitProg3)



#### Start

- New Application
- Search Online for Code Examples

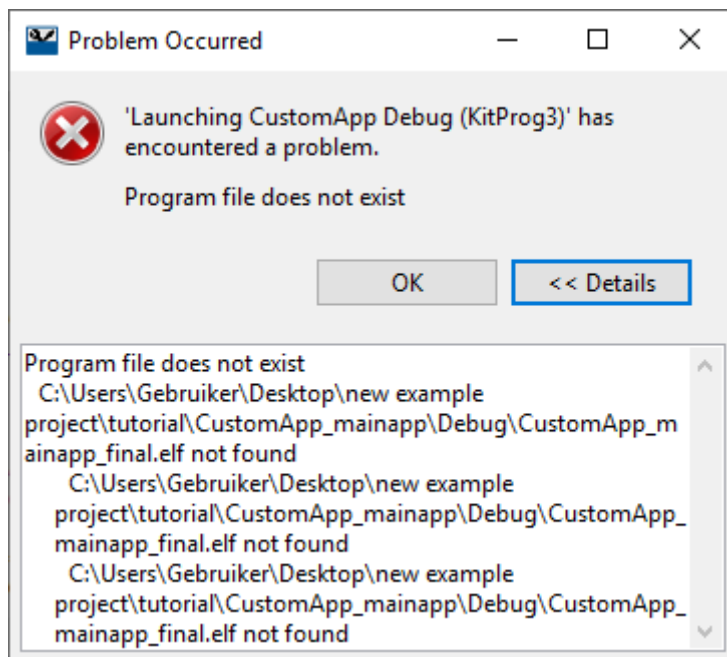
#### CustomApp\_mainapp

- Build CustomApp Application
- Clean CustomApp Application
- Project Build Settings
- Configure Device
- Select Middleware

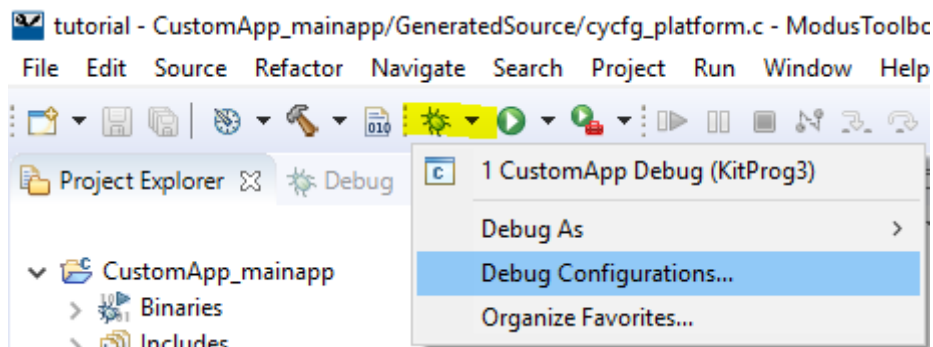
#### Launches

- CustomApp Debug (J-Link)
- CustomApp Debug (KitProg3)
- CustomApp Program (J-Link)
- CustomApp Program (KitProg3)

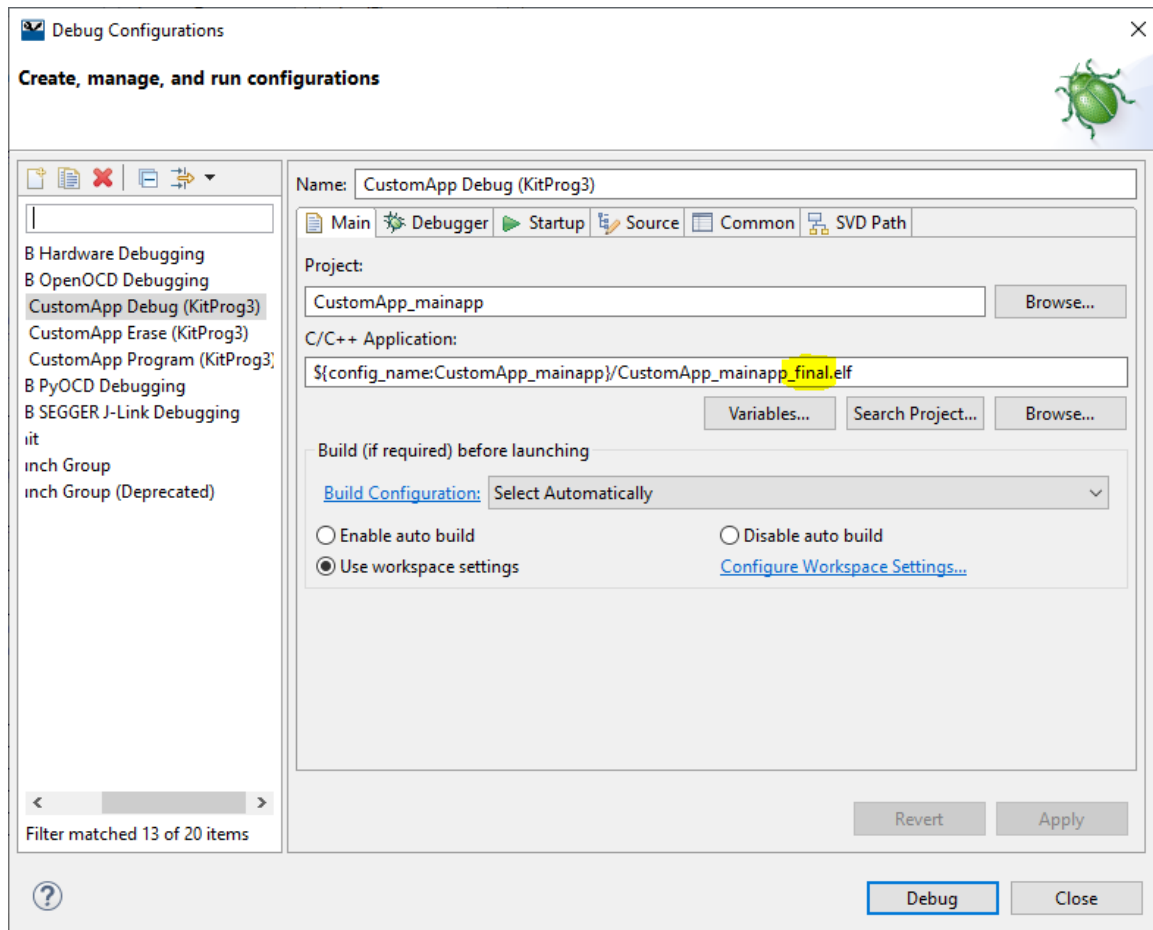
You should see this error



go too Debug configurations





Remove \_final from C/C++ Application:








Apply

Clean your application and debug





## Start

-  New Application
-  Search Online for Code Examples

## CustomApp\_mainapp

-  Build CustomApp Application
-  Clean CustomApp Application
-  Project Build Settings
-  Configure Device
-  Select Middleware

## Launches

-  CustomApp Debug (J-Link)
-  CustomApp Debug (KitProg3)
-  CustomApp Program (J-Link)
-  CustomApp Program (KitProg3)

Your application should work!

