Tags: #cec32x, #dev_tools, cec32x_devtool_21_porting_a_project

How to Port a Project

Below we provide steps of porting a demo project to your own project with its own folder structure using the DFSDM (digital filter for sigma-delta modulator) demo project as an example.

Testing out the demo project

Before porting the demo project, you need to test out the demo project in its original folder. Assume when you install the packages using `STM32CubeMX`, they are installed in the `C:\pjct_arm\_stm32cube\` folder. Then, we need to do the following steps:

- Navigate to `C:\pjct_arm\_stm32cube\STM32Cube_FW_F4_V1.25.2\Projects\STM32F412G-Discovery\Examples\DFSDM\DFSDM_AudioRecord` or `C:\pjct_arm\_stm32cube\STM32Cube_FW_L4_V1.16.0\Projects\32L476GDISCOVERY\Examples\DFSDM\DFSDM_AudioRecord`.
- Click on the `MDK-ARM` folder and then the `project.uvprojx` to start the project.
- Build the project to see if everything is ok. To reduce the clutter of the project directory, we can put the build files to a separate `Build` folder under the `DFSDM` folder beside the `DFSDM_AudioRecord` folder. This can be done by clicking the **Options for target...** button in Keil and then clicking the **Output** tab. Now, you can change the build folder by clicking **Select Folder of Object...**. Build the project. It should work fine. Load the built image to your board and then you can plug in a headphone in the headphone jack. Now, you can hear the noise picked by the onboard microphone. If you say something, you should be able to hear the echo of it.

Porting the demo project to a new project folder

Now, we can port the demo project to a new project folder to add new functionalities. Suppose we want to port the demo project to a new `lab08_stateMachine_and_audio` project. We need to perform the steps below.

- Create a new folder called `lab08b_stateMachine_and_audio`.
- Create three folders below the above folder:
  - `usr_src` for user source code.
  - `F412Discovery` (or `L476Discovery`) for ported project files.
- In the `F412Discovery` (or `L476Discovery`) folder, create the following folders:
  - `BSP` for board support packages. LED or Joystick operation functions are saved here.
  - `Drivers` for various drivers. Drivers for DFSDM or sound play functions are saved here.
  - `Inc` for include files.
  - `MDK-ARM` for the project management and start-up files.
  - `Src` for source files coming with the demo project.

- Now, we need to add files to these folders. This should be done according to the files used in the demo project. Open the demo project in Keil and we can see we have the following folders in the project pane:
    - `Drivers/STM32F4xx_HAL_Driver`
    - `Drivers/BSP/Components`
    - `Example/User`
    - `Drivers/BSP/STM32412G-Discovery`
    - `Example/MDK-ARM`
    - `Drivers/CMSIS`
- Copy all the files in the above folders to the newly created folders for the project.
- Modify the project management file to include the source files. Note that this step should be done carefully as missing one or two files will lead to errors in building the new project.
- Rebuild the project to see if it builds. If not, we need to check the error message and add the needed files.
- Add new functionalities to the new project.