

How to create a clean STM32CubeIDE Project ready to work in TouchGFXDesigner for a ST Demboard, starting from CubeMX.ioc generated from TouchGFXDesigner Demo.

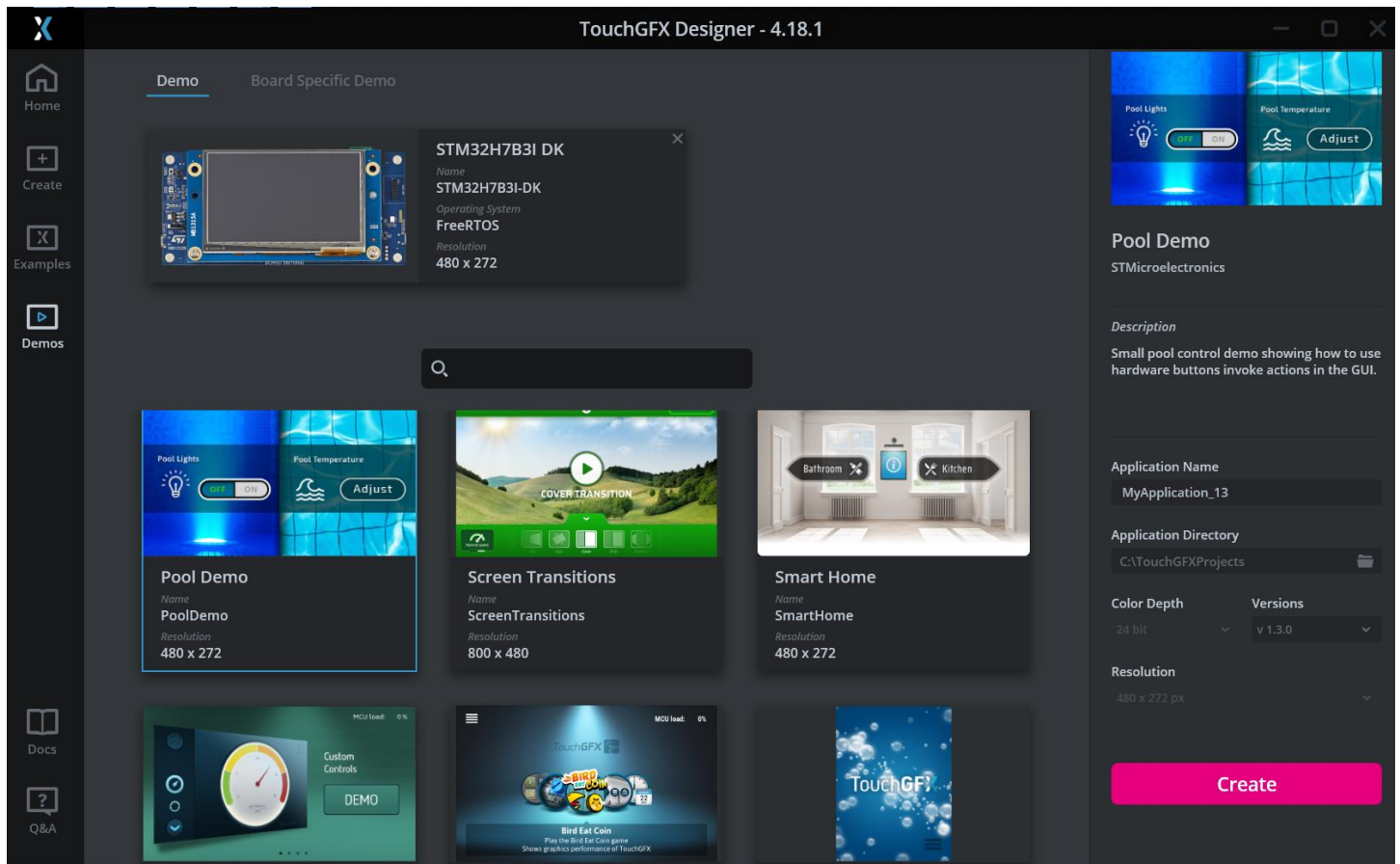
By **M.Airoidi** – For more info contact Airoidi to AVNET-SILICA ITALY

This document is hosted on:

- emcu.eu
- emcu-homeautomation.org

1_ From TouchGFXDesigner create a Demo for the demoboard ST that you want work with, in our case STM32H7B3I-DK.

Go to Demo click on '**Select Board Setup**' choose the demoboard to work with, and then choose the Demo.



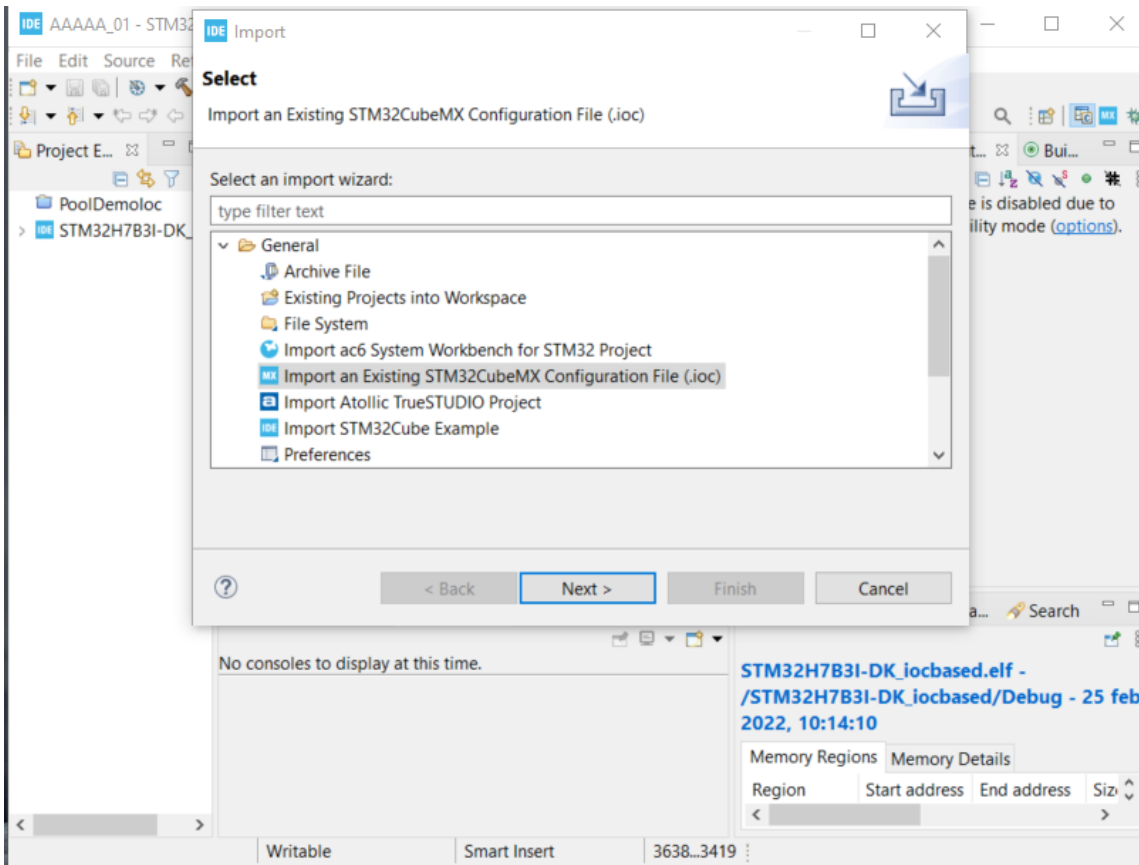
Shown here the Demos available with the version TouchGFXDesigner 4.18.1.

Take it one, change the project name if you want from 'MyApplication_13' to '**MyProjectTGFX**' and click on **Create** -> and then click-on:



If you left the Default destination, your project will be in C:\TouchGFXProject.

2_Run STM32CubeIDE go to File-> Import -> Import an Existing STM32CubeMX Configuration File (.ioc)



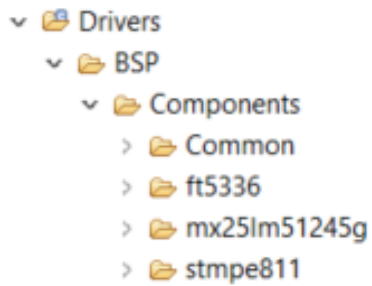
Select the File.ioc into the root of DemoProjectTGFX generated from TouchGFXDesigner, from which you want to start.

2_ When was imported and created NewProject, go to **NewProject/ToucGFX**, run **ApplicationTemplate.touchgfx.part** to create an application TouchGFXDesigner, then generate the code, come back in CubeIDE refresh Project and compile it, no errors should happen.

Copy the folder DemoProjectTGFX /Driver/BSP in NewProject/Driver.

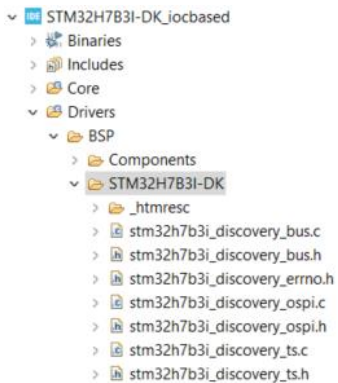
In reality the BSP element fundamental to let the display of the demoboard works are only few

2.1_ In the sub Folder BSP/components are needed only the folders Common, ft5336, mx25lm51245g e stmpe811.

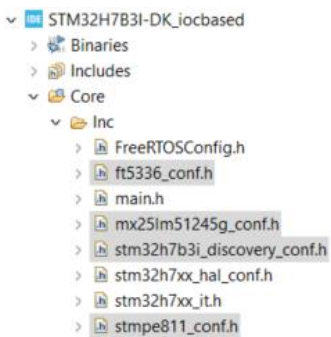


In the folder Common is needed only the file ts.h.

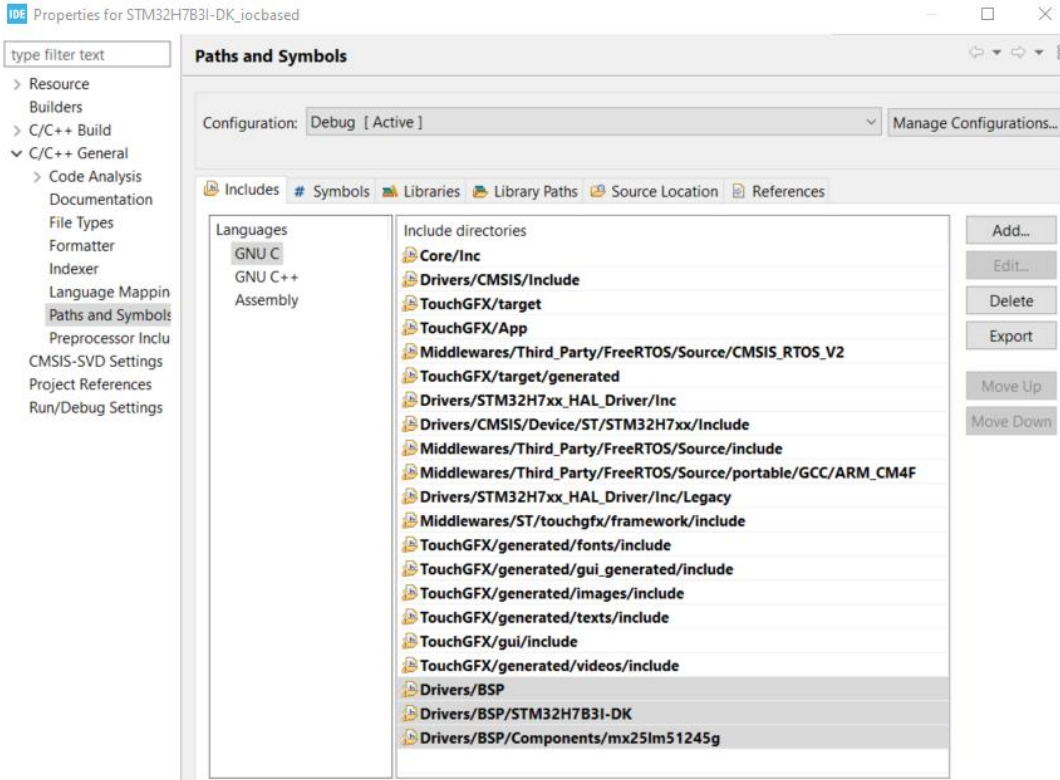
2.2_ In the subfolder BSP/STM32H7B3I-DK are needed these files:



3_ In the folder NewProject/Core/Inc add the followed highlighted files, that can be copied from DemoProjectTGFX /Core/Inc:



4_ Fix the dependencies and the Include pats: right click on the project -> properties -> C/C++General -> Paths and Symbols. Add the dependencies highlighted here:



Resolve any conflicts on dependencies, acting both on the `#include` of the Files and through the 'Paths and Symbols'.

5_ Replace the files .cpp and .hpp in NewProject /ToucGFX/target with those found in DemoProjectTGFX/ToucGFX/target. Is easy that the dependence in the file STM32TouchController.cpp

```
#include <stm32h7b3i_discovery_ts.h>
```

Need to be fixed, for example with:

```
#include "../Drivers/BSP/STM32H7B3I-DK/stm32h7b3i_discovery_ts.h"
```

6_ Complete the following routine for OCTOSpi initialization in the main.c file:

```
static void MX_OCTOSPI1_Init(void)
```

In our case with demoboard STM32H7B3I-DK the OCTOSPI configuration to add is

```
/* USER CODE BEGIN OCTOSPI1_Init 2 */
BSP_OSPI_NOR_Init_t Flash;
Flash.InterfaceMode = BSP_OSPI_NOR_OPI_MODE;
Flash.TransferRate = BSP_OSPI_NOR_DTR_TRANSFER;
BSP_OSPI_NOR_DeInit(0);
int32_t RetVal = BSP_OSPI_NOR_Init(0, &Flash);
if(RetVal != BSP_ERROR_NONE)
{
    Error_Handler();
}
RetVal = BSP_OSPI_NOR_EnableMemoryMappedMode(0);
if(RetVal != BSP_ERROR_NONE)
{
    Error_Handler();
}
/* USER CODE END OCTOSPI1_Init 2 */
```

Don't forget to add the its #include too, at the start of main.c

```
/* Private includes -----*/
/* USER CODE BEGIN Includes */
#include "stm32h7b3i_discovery_ospi.h"
/* USER CODE END Includes */
```

7_ In NewProject /ToucGFX run ApplicationTemplate.touchgfx.part to create the TouchGFX Application and then generate the code. Now you grafichs are in NewProject /ToucGFX with the name NewProject.touchgfx.

PS:

Everytime are made some changes to File.ioc from STM32CubeMX or File.touchgfx from TouchGFXDesigner will be necessary in STM32CubeID refresh the project and rebuild all it.

ST TouchGFX Wiki: <https://support.touchgfx.com/4.18/docs/introduction/welcome>

ST Video Tutorial su TouchGFX (MOOC): <https://www.youtube.com/c/stmicroelectronics/playlists>

Link Usefull:

<https://www.youtube.com/watch?v=NdGgtKwQ6fg>

Part.1: embadded display: <https://www.youtube.com/watch?v=yiHjOH7zJP0>

Part.2: work with external RAM: <https://www.youtube.com/watch?v=XnSg3UJfSFc>