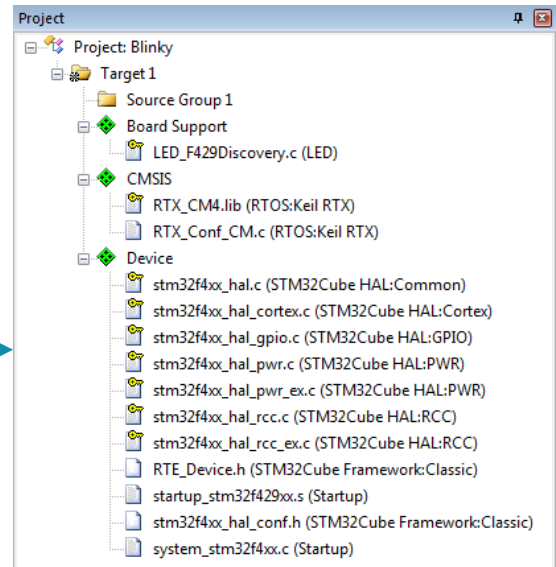


## Abstract

This tutorial shows how to create the Blinky project using the STMicroelectronics STM32F429I-Discovery Kit.

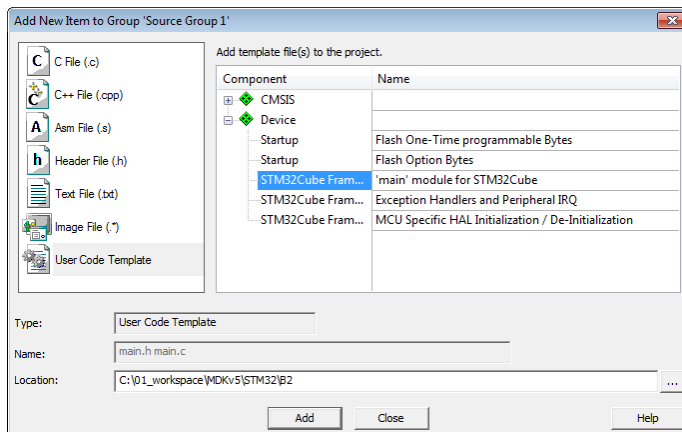
### Create a New Project for the Discovery Board

1. In the main  $\mu$ Vision menu, select **Project**  $\rightarrow$  **New  $\mu$ Vision Project...**  
The 'Create New Project' window opens up.
2. Create a new directory called **Blinky** and enter **Blinky** for the File name. Press **Save**.
3. In the 'Select Device for Target' window select **STM32F429ZITx** and press **OK**.
4. In the 'Manage Run-Time Environment' window select the following Software Components:
  - a. **CMSIS:RTOS (API):Keil RTX**
  - b. **Board Support(STM32F429I-Discovery): LED (API):LED**
  - c. **Device:STM32Cube Framework (API):Classic**
  - d. When done, press **Resolve** and afterwards **OK**.
5. The Project window should look like this



### Add user code templates *main.c* and *Thread.c*

6. Right-Click **Source Group 1** and select **Add New Item to Group 'Source Group 1'...**
7. In the upcoming window, select **User Code Template** and then expand **Device**. Select '**main**' module for **STM32Cube** and click **Add**:



8. Repeat the process, expand **CMSIS** and choose **CMSIS-RTOS Thread**. You now should see a *main.c* and a *Thread.c* file below the Source Group 1.

### Configure CMSIS-RTOS RTX

9. Open *RTX\_Conf\_CM.c*, select the **Configuration Wizard** tab and press **Expand All**.
10. Change the **RTOS Kernel Timer input clock frequency [Hz]** to **16800000** as the Discovery kit runs on 168 MHz.

## Configure the Target Options

11. Connect the Mini-USB cable to **USB ST-LINK**
12. Click on  or press **ALT+F7**
13. Select the **Debug** tab and choose **ST-Link Debugger**. Press **Settings**.
14. On the Debug tab, set the **Port** to **SW**. Click **OK** twice.



## Add the Blinky code

15. Change *main.c* as follows. After line 44:

```
43  /* Includes -----*/
44  #include "main.h"
45  #include "stm32f4xx.h"           // Device header
46  #include "Board_LED.h"         // ::Board Support:LED
47  extern int Init_blink_LED (void);
```

16. Before line 115 add:

```
115  osKernelInitialize ();           // initialize CMSIS-RTOS
116  LED_Initialize();
117  Init_blink_LED();
118  osKernelStart ();               // start thread execution
```

17. Change *Thread.c* as follows:

```
#include <cmsis_os.h>               // CMSIS RTOS header file
#include "Board_LED.h"             // ::Board Support:LED

void blink_LED (void const *argument); // thread function
osThreadId tid_blink_LED;          // thread id
osThreadDef (blink_LED, osPriorityNormal, 1, 0); // thread object

int Init_blink_LED (void) {

    tid_blink_LED = osThreadCreate (osThread(blink_LED), NULL);
    if(!tid_blink_LED) return(-1);

    return(0);
}

void blink_LED (void const *argument) {

    while (1) {
        LED_On (0);
        osDelay(500);
        LED_Off (0);
        osDelay(500);
        osThreadYield();
    }
}
```

## Build the Project and run it on the Target

18. Go to **File** → **Save All**
19. Go to **Project** → **Build Target** (or press **F7**)
20. Go to **Flash** → **Download** to flash the project to the target
21. Go to **Debug** → **Start/Stop Debug Session** (or press **CTRL+F5**)
22. Go to **Debug** → **Run** (or press **F5**) to run the project on the target.  
LED **PG13** will start flashing.

