# NXP LPC4330 Xplorer: Blinky Lab

MDK Version 5 Tutorial

Spring 2015, V 1.0

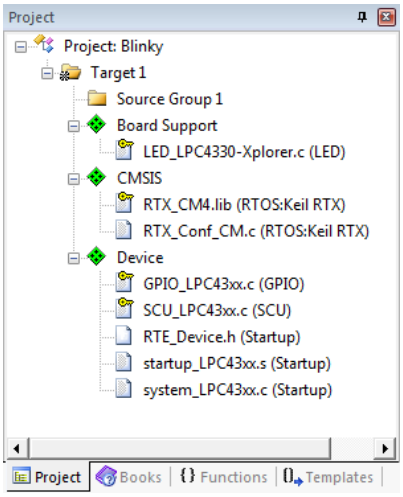
ARM KEIL Microcontroller Tools

## Abstract

This tutorial shows how to create the Blinky project for the Cortex-M4 in the LPC4330 on the NXP LPC4330 Xplorer development board.

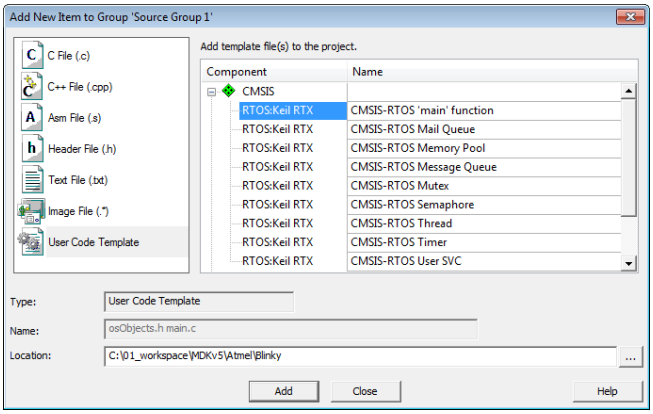
### Create a New Project for the Xplorer Board

1. In the main µVision menu, select **Project → New µVision 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 **LPC4330:Cortex-M4** 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(LPC4330-Xplorer):LED (API):LED**
   c. 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 **CMSIS**. Select **CMSIS-RTOS 'main' function** and click **Add**:




8. Repeat the process 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 **180000000** as the LPC4330 runs on 180 MHz.

### Add the Blinky code

11. Change *main.c* as follows:

```
#define osObjectsPublic                     // define objects in main module
#include "osObjects.h"                      // RTOS object definitions
#include "LPC43xx.h"                        // Device header
#include "Board_LED.h"                      // ::Board Support:LED

extern int Init_blink_LED (void);

int main (void) {
  osKernelInitialize ();                    // initialize CMSIS-RTOS

  LED_Initialize();
  Init_blink_LED();

  osKernelStart ();                         // start thread execution
  while(1);
}
```

12. 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();
  }
}
```

## Debug Adapter

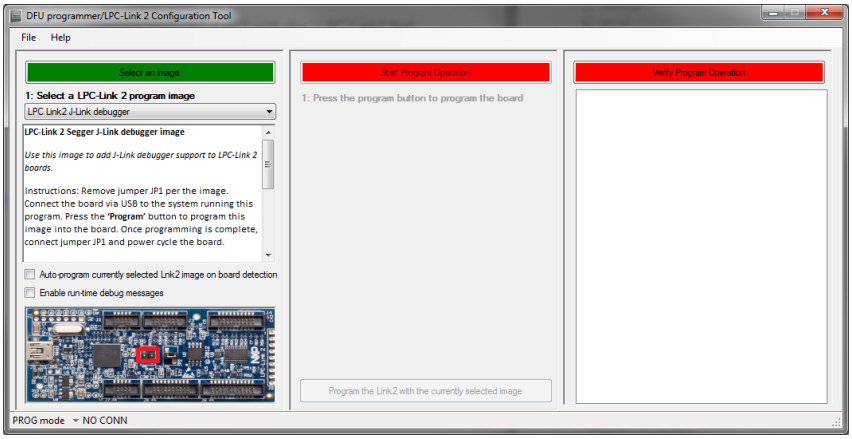
In this workshop we are using the **LPC-Link2 debug adapter** with J-Link firmware. You will need a Mini-USB cable to connect the LPC-Link2 with the PC that runs the development tools. The LPC-Link2 Debug Adapter should be configured as described below.

## Download and Install J-Link Software & Documentation Pack for Windows

Visit www.segger.com/jlink-software.html and download the latest version of the J-Link software and documentation pack for Windows. The ZIP file contains an EXE file that needs to be installed on your computer before the configuration of the LPC-Link2 that is described in the next step.

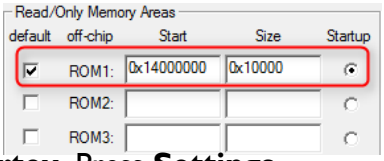
## Configure the LPC-Link2 as J-LINK debugger

Visit www.lpcware.com/lpclink2 to obtain the latest LPC-Link Configuration Tool. After installation, run the tool and follow the on-screen instructions to program your LPC-Link2 with the "LPC-Link2 J-Link debugger" firmware.

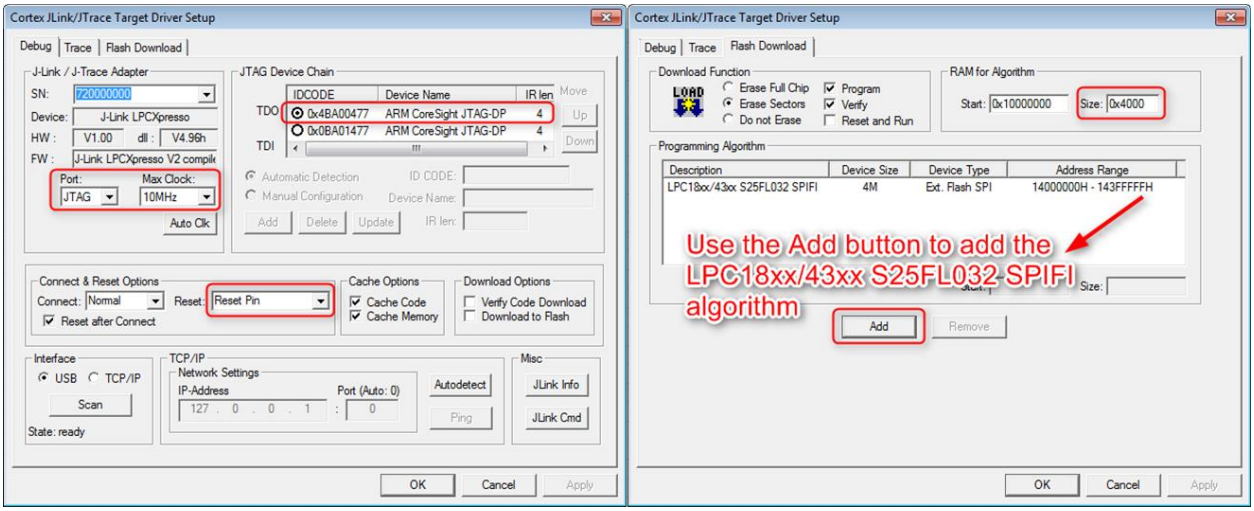



**Note:** This software requires the .NET framework to be present on your PC.

## Configure the Target Options

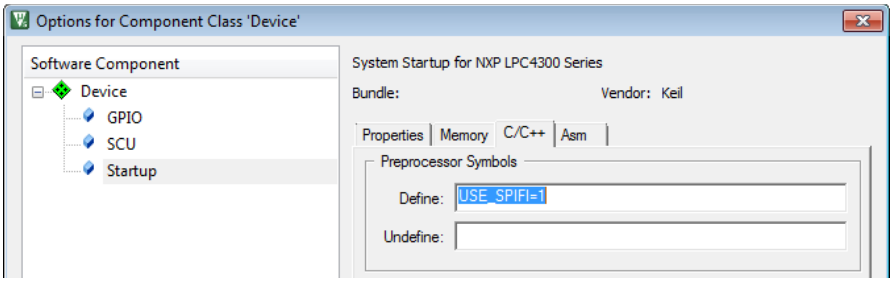



13. Click on  or press **ALT+F7**
14. Enter the following in the **Read/Only Memory Areas:**
15. Select the **Debug** tab and choose **J-LINK / J-TRACE Cortex**. Press **Settings**.
16. Configure **Debug** and **Flash Download** as follows:



## Build the Project and run it on the Target

17. Right-click on *startup_LPC43xx.s (Startup)* and choose **Options for Component Class 'Device'**
18. Select the **C/C++** tab and enter `USE_SPIFI=1` in the **Define** box and click **OK**:




19. Go to **File → Save All**
20. Go to **Project → Build Target** (or press **F7**)
21. Connect the Mini-USB cable to the LPC-Link 2 and the Micro-USB cable to a USB connector on the LPC4330-Xplorer board
22. Connect the two boards with the flat cable (make sure the red mark on the cable is on the right side of the connectors of each board)
23. Go to **Flash → Download** to flash the project to the target
24. Go to **Debug → Start/Stop Debug Session** (or press **CTRL+F5**)
25. Go to **Debug → Run** (or press **F5**) to run the project on the target.
    LED **D3** will start flashing.

## More information

For a detailed description on how to setup multi-core projects on the LPC4430, please refer to AN 272: http://www.keil.com/appnotes/docs/apnt_272.asp