



# **Getting Started and Blinky Example**

This **Quick Start Guide** shows how to install the Keil software development tools and run a variety of example projects on the MCBTWRK60 board containing the Freescale **MK60N512VMD100** device.

## Installing the Software

The MCBTWRK60 works with the Keil **MDK-ARM** (Microcontroller Development Kit) which features the industry-standard ARM<sup>®</sup> compiler, the  $\mu$ Vision4 IDE/Debugger and analysis tools.

All examples shown in this quick start guide will work with the MDK-Lite Edition:

Download the latest version of MDK-ARM from:

## www.keil.com/demo/eval/arm.htm

- Double click the downloaded file and follow the instructions to install MDK-ARM into your root directory
- The evaluation version does not require a license key

**Note:** MDK-ARM will be installed in your root directory by default. This ensures all included examples work correctly.

# Connecting to the MCBTWRK60

MDK-ARM supports ULINK2, ULINK*pro* and OS-JTAG downloading and debugging for the MCBTWRK60 board.

To connect via ULINK2 or ULINKpro to your PC you need:

- USB A-miniB cable to power the board.
- ULINK2 or ULINKpro Debug Adapter with USB A-B cable

To connect via OS-JTAG to your PC you need:

USB A-miniB cable (provides power & debug connection).

## Using a ULINK Adapter or OS-JTAG

The ULINK family of debug adapters connect your PC's USB port to the Cortex Debug connector on your target board allowing you to download and analyze embedded programs running on your target hardware.

ULINK2 supports JTAG, Serial Wire Debug (SWD), and Serial Wire Viewer (SWV) for on-the-fly debug and analysis. ULINK*pro* supports these interfaces plus ETM Instruction Trace.

The **ULINK2** and **ULINK***pro* adapters connect to the MCBTWRK60 using the Cortex Debug connector.



The **OS-JTAG** JTAG interface from P&E Micro is included on the MCBTWRK60 board. The USB cable provides the power and the debug connection from the PC to the MCBTWRK60.



## **Blinky Example**

This example demonstrates the ease of downloading and debugging an application on a target board.

It can be found at:

### C:\Keil\ARM\Boards\Freescale\TWR-K60N512\Blinky

To use this example



- Open the Blinky.uv2 project file
   Project Open
- hpile and link the Blinky application
   Project Build
- Flash Download

- The LEDs on the target are controlled by the potentiometer setting. As the setting increases, the LEDs scroll faster.
- 🛚 🙋 🛛 Start debug mode

Using the debugger commands, you may:

- Single step through code
- Set breakpoints
- Run the application
- Review variables in the watch window
- Reset the device to re-run the application
- Use the yellow arrow (program counter) to view the current assembler or C statement

## www.keil.com

# Using the MCBTWRK60 Evaluation Board



## Blinky example with RTX

This example implements a step-motor driver application using the RTX real-time operating system. Four LEDs blink to simulate the activation of the four output driver stages.

#### It can be found at:

## C:\Keil\ARM\Boards\Freescale\TWR-60N512\RTX\_Blinky

To use this example:

- Start µVision
- Open the Blinky project file

#### **Project - Open**

 Compile and link the application, then program it into on-chip Flash ROM

**Project - Build Target** 

#### Flash - Download

### **Press RESET** button

The LEDs on the target blink from left to right, and back again. Turn the potentiometer dial to vary the speed of blinking. You may debug the example project using the  $\mu$ Vision debugger.

- 🛚 🧕 Start debug mode
  - Set breakpoints

A breakpoint can be a simple halt at an instruction, or a complex combination of boolean conditions

•  $\boxed{\Xi}$  Run the application

The code runs to the end, or to the next breakpoint

# **Creating a New Project**

MDK makes it easy to create new programs for the K60N512 by providing the Device Database, Startup Code and Configuration Wizard to help you select your target, and configure the tools.

Start a new project

### Project - New µVision Project

Assign a project file name and save to your preferred location

 $\mu\text{V}\textsc{ision}$  opens up the Device Database for you to select your targeted device.

Scroll down to vendor Freescale, and select a device

elect Device for Target 'Target 1' CPU | Vendor: Freescale Semiconductor Device: MK60N512VMD100 Toolset: ARM Data base Description Core features 32-bit ARM Cotex-M4 core (up to 180MHz CPU Clock) DSP Support. MPU Nested vectored intemupt contr. (NVIC) Async: wake-up internupt contr. (AWIC) Debug & trace capability - 2 pin serial wire debug (SWD) - 16EET 1149.1 Joint Test Action Group (JTAG) - IEEET 1149.7 compact JTAG (cJTAG) - IEEET 1149.7 compact JTAG (cJTAG) - IEEE 1149.7 compact JTAG (cJTAG) - Trace port interface unit (TPU) - Rash patch and breakpoint (FPB) - Data watchoomit and trace (DWT) MK10X128VMD100 🔺 MK10X256VMD100 MK20N512VMD100 MK20X128VMD100 CI MK20X256VMD100 MK30N512VMD100 MK30X128VMD100 MK30X256VMD100 MK40N512VMD100 MK40X128VMD100 Data watchpoint and trace (DWT) Instrumentation trace macrocell (ITM) MK40X256VMD100 MK60N512VMD100 ystem and power management SW/HW watchdog with external monitor pir MK60X256VMD100 ~ Cancel Help

Detailed information about the RTX tasks and configuration can be seen in the **Kernel Aware** dialog windows.

## Debug - OS Support

The **RTX Tasks and System** window displays system resource and active task information.

Property	Valu	e								
🖃 System	Item	0 		Value						
	Time	r Number:		0						
	Tick 1	limer:		10.000	) mSec					
	Rour	nd Robin Timeout:		50.000	) mSec					
	Stack	c Size:		200						
	Tasks	s with User-provided St	ack:	0						
	Stack	Overflow Check:		Yes						
	Task	Usage:		Availa	ble: 7, Used	: 5				
	User	Timers:		Availa	ble: 0, Used	: 0				
				-						
<ul> <li>Tasks</li> </ul>	ID	Name	Priori	ity	State	Delay	Event Value	Event Mask	Stack Load	
	255	os_idle_demon	0		Running				0%	
	6	clock	1		Wait_AND		0x0000	0x0100	32%	
	5	phaseD	1	1	Wait_AND		0x0000	0x0001	32%	
	4	phaseC	1	1	Wait_DLY	30	0x0000	0x0001	36%	
	3	phaseB	1	2	Wait_AND		0x0000	0×0001	32%	
	2	phaseA	1	1	Wait_AND		0x0000	0x0001	32%	

The **Event Viewer** displays detailed task switching information.

	Min Time: 0.543362 s						Max Time: 6.356558 s						Range: 5.000000 s							Grid: 0.2500			
init phaseA phaseB phaseC								1	3				-	3	• •		-				Ę		
	1.		- Ti	me:				1.0	use 5949	8 5	5		0.5	446	15 s			0.5	248	84 s	- 13	905:	18 1
phaseD clock			1			ł.		ų.		•	ł.		1	4		1		h	ж е				
ICO Idle																							
laie		• •	• •	• •			• •		*	•		• •				• •		• •		• •	1	6 ( A	
0.50				; ;	;	1.1	: :	11	1	1	3	000	000 s	;	: :	: :	1	: :	ł	: :	÷	5	.500

 $\mu Vision$  can automatically import the device's startup code into the project workspace, to save time, and get you started quickly.

To add files to your new project, expand the folder **Target 1** in the **Project** window and right click on **Source Group 1** to open the context menu. Then use **Add Files to Group** to add existing source files to the project.

New source files may be created for this project using File - New

Compile and debug following the steps shown for the example above. Ensure that the debug driver is selected and configured.

To select and configure the ULINK adapter or OS-JTAG:

Click on the Options for Target icon

#### On the Debug tab,

-	
For Debug Adapter	Select
Keil ULINK2	ULINK Cortex Debugger
Keil ULINKpro	ULINK Pro Cortex Debugger
P&E Micro OS-JTAG	Pemicro OSJTAG

#### then click **Settings** to configure the debug session

		×
Linker D	ebug Utilities	
🖲 Use:	ULINK Cortex Debugger	▼ Settings
✓ Load Initializatio	ULINK Cortex Debugger ULINK Pro Cortex Debugger Altera Blaster Cortex Debugger Stellaris ICDI Signum Systems JTAGiet	o main()
Restore	Cortex-M/R J-LINK/J-Trace ST-Link Debugger RDI Interface Driver	Edit

 Do the same under the Utilities tab to configure Flash download

# www.keil.com