

Application Note: 201

Setting Up ULINK2 for LogicPD LH7A404 Card Engine



Abstract

This application note provides instructions for connecting and setting up the LogicPD LH7A404 Card Engine evaluation board for use with the ULINK2 USB/JTAG adapter and the MDK-ARM development tools. It also provides step-by-step instructions for building, downloading and debugging a simple μ Vision example project on that board.

Contents

Abstract.....	1
You will need:.....	1
Hardware Setup	1
Software Setup.....	2
BLINKY Example Program	3
Editing BLINKY	3
Building BLINKY	3
Debugging BLINKY.....	4

You will need:

To successfully complete the procedures in this application not, you will need the following:

- Zoom SDK baseboard
- LH7A404 Card Engine
- 5 VDC power supply
- ULINK2 USB-JTAG adapter
- USB A/B cable
- Keil MDK-ARM Tools ([Evaluation](#) or Licensed Version)
- The LH7A404.ZIP file containing the BLINKY example project

Hardware Setup

This procedure shows you how to connect the Zoom SDK baseboard set to your PC via the ULINK2 USB/JTAG adapter, and set the appropriate jumpers for downloading and debugging via JTAG.

1. Install the LH7A404 Card Engine on the Zoom SDK baseboard according to the LogicPD User's Guide.
2. Connect the 20-pin ribbon cable from the ULINK2 JTAG adapter to the **J36** header on the Zoom SDK baseboard (Figure 1). This connector is not keyed. Make sure pin 1 on the cable (red tracer) is oriented toward pin 1 on the header.

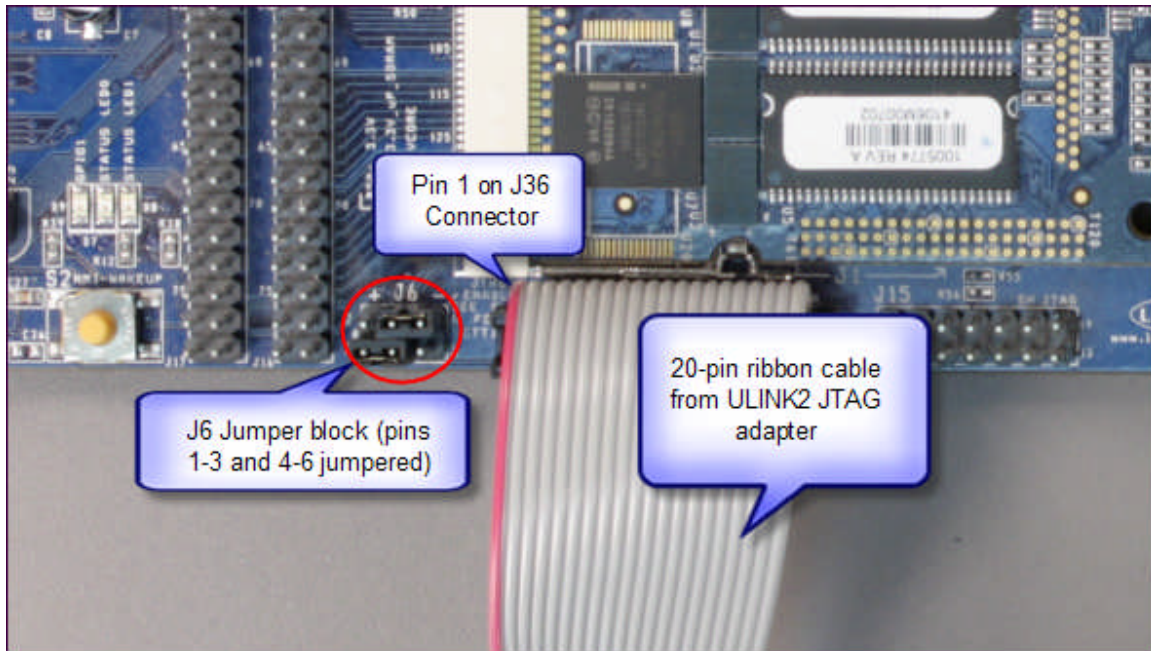


Figure 1 Jumper Settings for the Zoom SDK Baseboard

3. On the Zoom SDK baseboard, at jumper block **J6** make sure pins 1-3 and pins 4-6 are jumpered (normal JTAG mode).
4. Connect the 5VDC power supply to the Zoom SDK baseboard.
5. Connect a standard USB A/B cable between the ULINK2 adapter and an unused USB port on your PC.

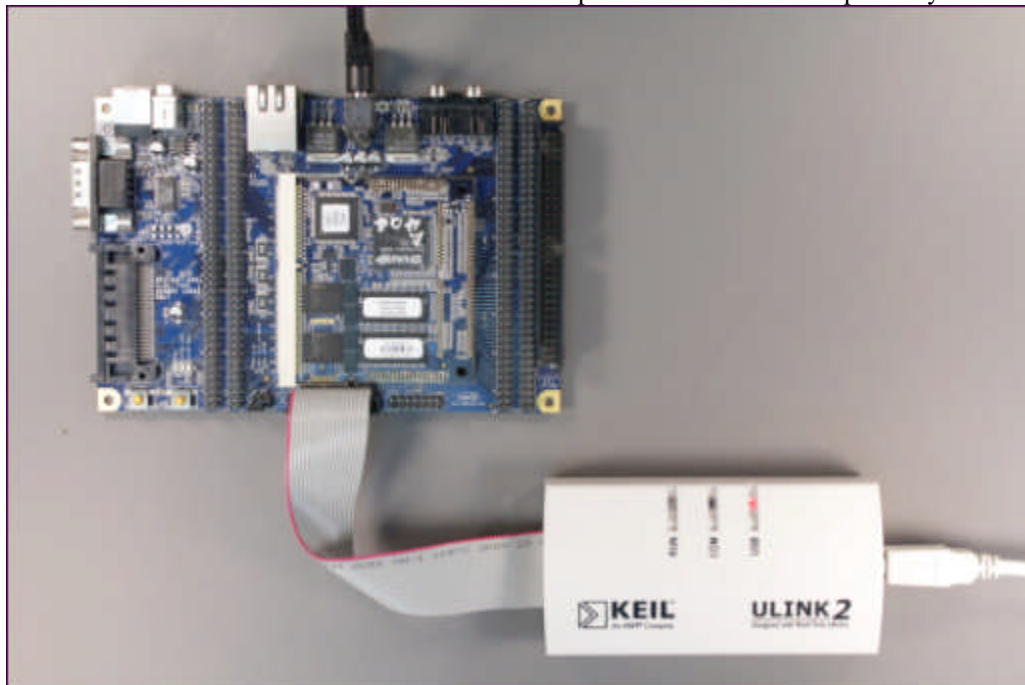


Figure 2 ULINK2 Connected to Zoom SDK Baseboard

Software Setup

1. Install the Keil MDK-ARM Development tools (either the [Evaluation](#) or Licensed version) on your PC.
2. Extract the BLINKY project from the LH7A404.ZIP file to a folder on your local hard drive.

BLINKY Example Program

The BLINKY example program is a simple application that blinks the GPIO0 and GPIO1 LEDs continuously.

1. To load the BLINKY example project select **Open Project** from the **Project** menu and open **BLINKY.UV2** from the folder chosen in the Software Setup section, Step 2.

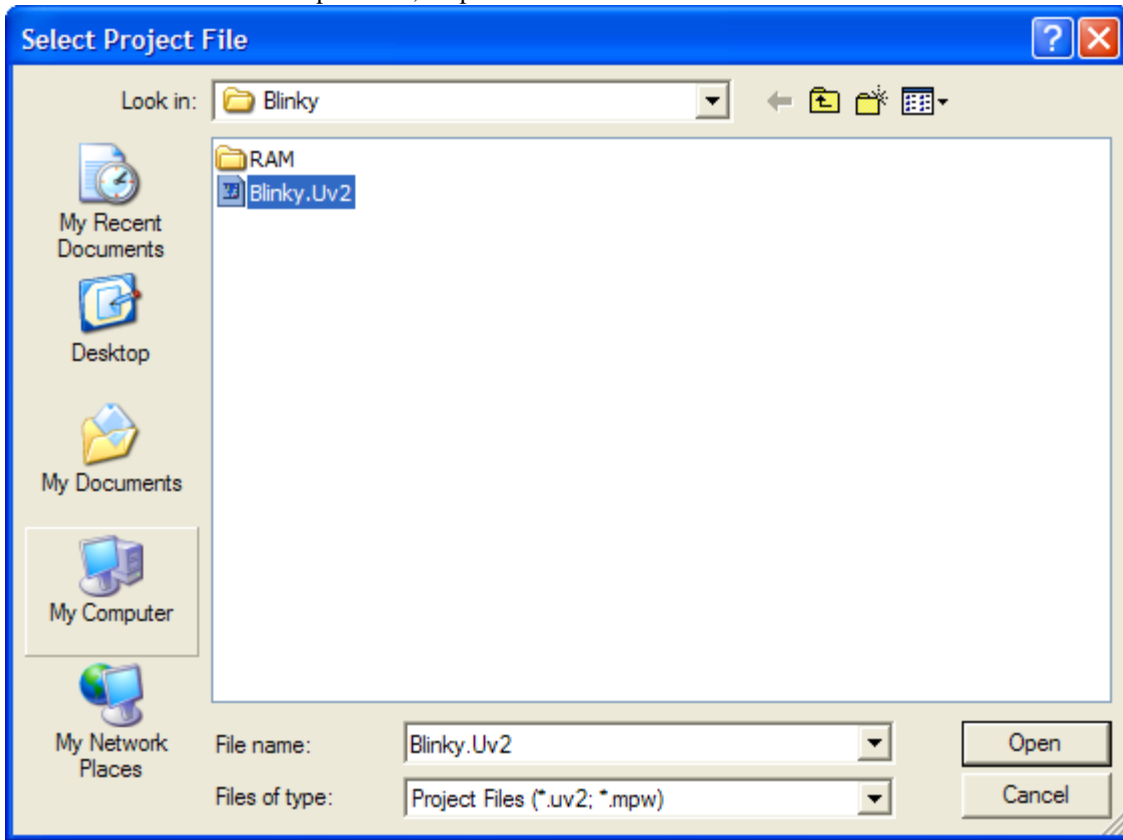


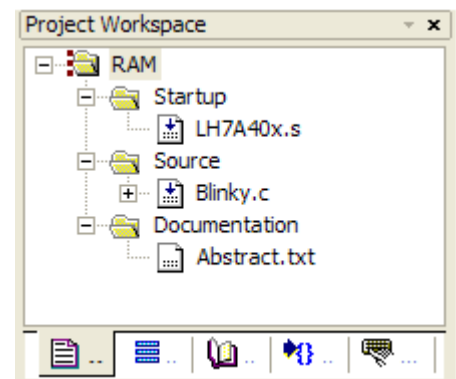
Figure 3 Select Project File Dialog

2. When the BLINKY project opens, the μ Vision **Project Workspace** window displays the source files that make up the project.
 - **LH7A40x.s** contains the CPU startup code for the Sharp LN7A404 device.
 - **Blinky.c** contains the application code that toggles the LEDs.

The **Abstract.txt** file contains documentation about the project.

Editing BLINKY

You may edit Blinky.c or review the source code. Double click on Blinky.c in the Files page of the **Project Workspace** window. μ Vision loads and displays the contents of Blinky.c in an editor window.



Building BLINKY

Build the BLINKY example program using the following procedure. The executable files are placed in an output folder, ready for downloading.

Setting Up ULINK2 for LogicPD LH7A404 Card Engine



When you are ready to compile and link your project, use the **Build Target** command from the **Project** menu or the Build toolbar. On the Build toolbar, select the **RAM** target for your program.

µVision translates and links the source files and creates an absolute object module that you may load into the µVision debugger for testing. The status of the build process displays in the **Build** page of the **Output Window**.

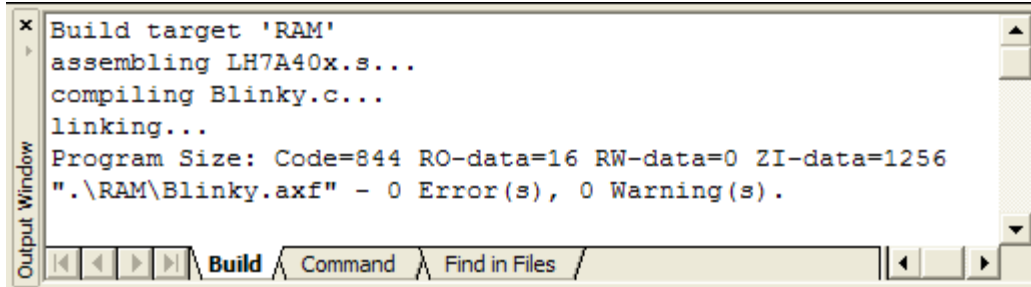


Figure 4 Output Window After Build

Once the BLINKY application is built, you are ready to start debugging.

Debugging BLINKY



Use the **Start/Stop Debug Session** toolbar button or the µVision command **Debug — Start/Stop Debug Session** to connect the debugger to the Keil ULINK2 Adapter. The CPU stops and connects to the Embedded ICE of the LN7A404 device.

When the debugger starts, the application program is downloaded to the LN7A404 target board's RAM and the program executes then halts at the start of the `main ()` function. From here you can use the debugging buttons or commands to verify the application executes properly.

For more details on the debugging features, refer to the [µVision User's Guide](#).