

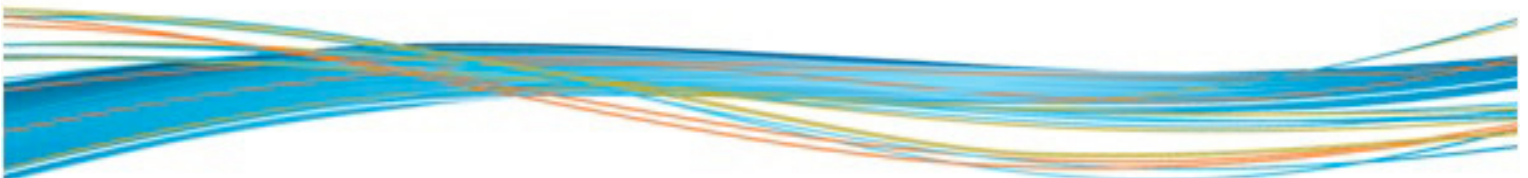
# USER MANUAL



LPC4357-EVB: Development Platform for the Dual-Core Cortex-M4/M0 LPC4357

*Preliminary*

From  
element14





## Table of Contents

<b>INTRODUCTION</b> .....	<b>5</b>
<b>1 QUICK START</b> .....	<b>5</b>
1.1 UNPACKING .....	5
1.2 CONNECTING CABLES.....	5
1.3 JUMPER SETTING .....	6
1.4 BOARD PARAMETER .....	7
1.4.1 <i>Physical Size</i> .....	7
1.4.2 <i>Peripherals</i> .....	8
1.5 IDE AND DEBUGGER INSTALLATION .....	9
<b>2 BLINKY LED DEMO EXAMPLE</b> .....	<b>13</b>
2.1 BUILDING THE PROJECT .....	13
2.2 DOWNLOAD THE PROJECT.....	14
2.2.1 <i>Set up the Debug configuration</i> .....	14
2.2.2 <i>Download</i> .....	15
2.2.3 <i>Debug</i> .....	16
<b>3 MAJOR FUNCTIONAL MODULES</b> .....	<b>17</b>
3.1 PROCESSOR .....	17
3.2 MEMORY .....	18
3.3 AUDIO.....	19
3.4 USBHOST AND USB DEVICE .....	20
3.5 SD CARD.....	20
3.6 USER KEYS.....	20
3.7 JOYSTICK .....	21
3.8 LCD.....	21
3.9 TOUCH SCREEN .....	22
<b>4 SOFTWARE EXAMPLES FOR TESTING PERIPHERALS</b> .....	<b>23</b>
4.1 GPIO.....	23
4.1.1 <i>GPIO Key</i> .....	23
4.1.2 <i>GPIO Led Blinky</i> .....	23
4.2 DUAL CORE.....	23
4.2.1 <i>Int Demo</i> .....	23
4.2.2 <i>Mailbox Demo</i> .....	24
4.3 I <sup>2</sup> S AUDIO .....	24

4.4	LCD DEMO .....	24
4.5	USB CDC .....	25
4.6	I <sup>2</sup> C MASTER: ACCELEROMETER DEMO.....	25
4.7	SD CARD: READ AND WRITE DEMO.....	26
4.8	USB HOST .....	28
<b>6</b>	<b>ESD PRECAUTIONS AND PROPER HANDLING PROCEDURES .....</b>	<b>29</b>

## LIST OF FIGURES

FIGURE 1-1: LPC4357-EVB BOARD .....	5
FIGURE 1-2 JUMPERS (USB1 BOOT MODE SHOWN) .....	6
FIGURE 1-3 BOARD TOP SIDE.....	7
FIGURE 1-4 BOARD BOTTOM SIDE.....	7
FIGURE 1-5: LPC4357-EVB BLOCK DIAGRAM .....	8
FIGURE 1-6: KEIL MDK INSTALLATION 1 .....	9
FIGURE 1-7: KEIL MDK INSTALLATION 2 .....	10
FIGURE 1-8: KEIL MDK INSTALLATION 3 .....	10
FIGURE 1-9: KEIL MDK INSTALLATION 4 .....	11
FIGURE 1-10: KEIL MDK INSTALLATION 5 .....	11
FIGURE 1-11: KEIL MDK INSTALLATION 6 .....	12
FIGURE 1-12: KEIL MDK INSTALLATION COMPLETED .....	12
FIGURE 2-1: INFLASH CONFIGURATION.....	13
FIGURE 2-2: SUCCESSFUL BUILD .....	14
FIGURE 2-6: CONFIGURATION ICON .....	14
FIGURE 2-7: DEBUG CONFIGURATION.....	14
FIGURE 2-8: DEBUG ADAPTER SETTINGS .....	15
FIGURE 2-3: DOWNLOAD .....	15
FIGURE 3-1: SDRAM MODULE .....	18
FIGURE 3-2: NOR FLASH MODULE .....	18
FIGURE 4-1: VIRTUAL COM DEMO.....	25
FIGURE 4-2: ACCELEROMETER DEMO .....	26

## LIST OF TABLES

TABLE 1-1 BOOT MODE JUMPER SETTINGS.....	6
--	---

## Introduction

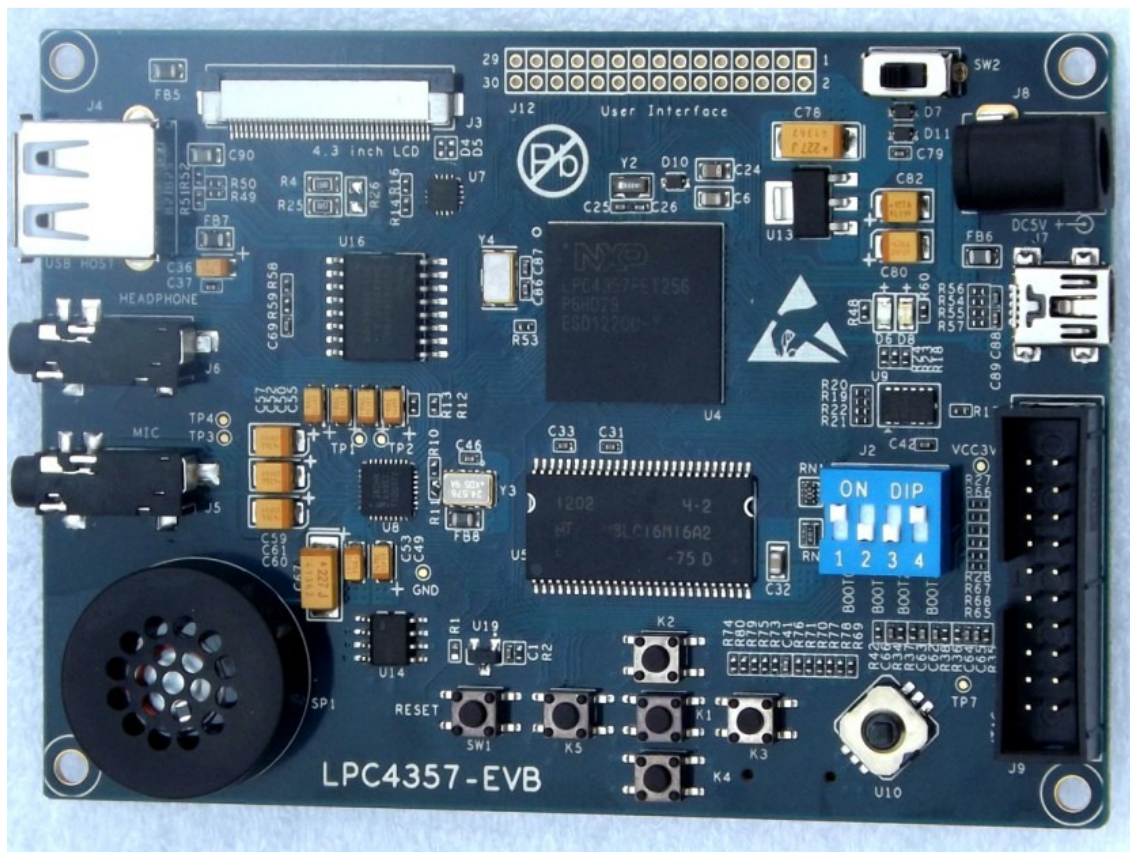
This document provides information on the LPC4357-EVB, a development platform for the NXP dual-core Cortex-M4 and Cortex-M0 controller, brought to you exclusively from element14 .

## 1 Quick Start

### 1.1 Unpacking

- LPC4357-EVB Board
- USB Cable

Figure 1-1: LPC4357-EVB Board



### 1.2 Connecting cables

- ✚ Power supply: USB power (J7) or 5V DC power adapter at J8
- ✚ JTAG Debugger Connector : J9. Connect a Keil ULink adapter (or similar debugger) here.

### 1.3 Jumper Setting

Figure 1-2 Jumpers (USB1 Boot Mode Shown)

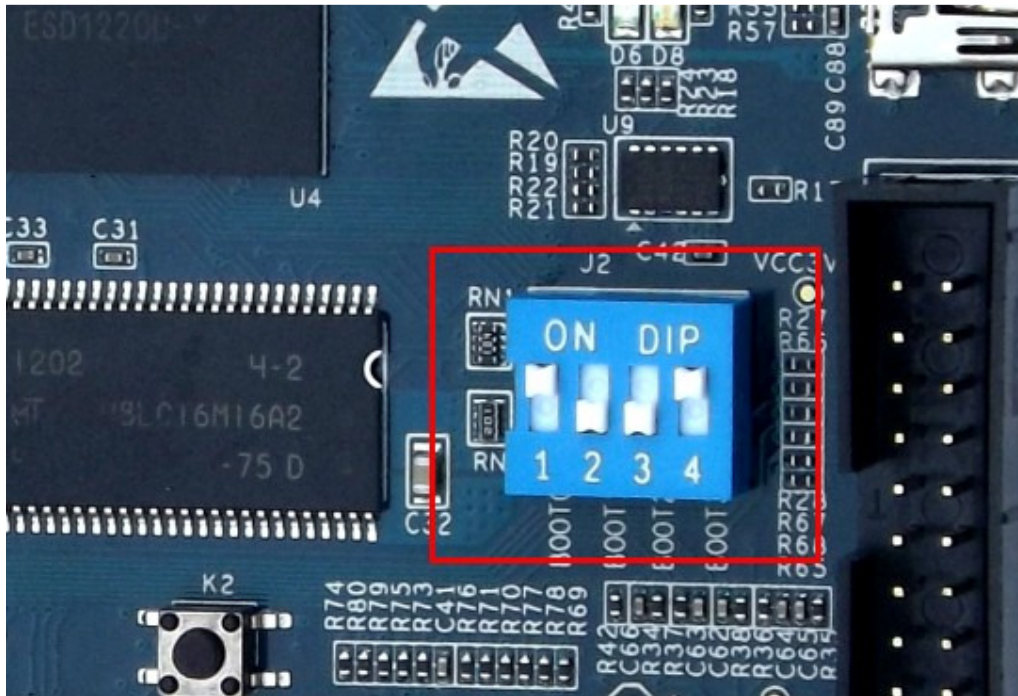


Table 1-1 Boot mode jumper settings

Boot mode	1	2	3	4	Description
<b>USART0</b>	UP	UP	UP	UP	Boot from device connected to USART0 using pins P2_0 and P2_1. For flash parts, enter UART ISP mode.
<b>SPIFI</b>	DN	UP	UP	UP	Boot from Quad SPI flash connected to the SPIFI interface on P3_3 to P3_8
<b>EMC 8-bit</b>	UP	DN	UP	UP	Boot from external static memory (such as NOR flash) using CS0 and an 8-bit data bus.
<b>EMC 16-bit</b>	DN	DN	UP	UP	Boot from external static memory (such as NOR flash) using CS0 and a 16-bit data bus.
<b>EMC 32-bit</b>	UP	UP	DN	UP	Boot from external static memory (such as NOR flash) using CS0 and a 32-bit data bus.
<b>USB0</b>	DN	UP	DN	UP	Boot from USB0.
<b>USB1</b>	UP	DN	DN	UP	Boot from USB1.
<b>SPI(SSP)</b>	DN	DN	DN	UP	Boot from SPI flash connected to the SSP0 interface on P3_3 (function SSP0_SCK), P3_6 (function SSP0_MISO), P3_7 (function SSP0_MOSI), and P3_8 (function SSP0_SSEL).
<b>USART3</b>	UP	UP	UP	DN	Boot from device connected to USART3 using pins P2_3 and P2_4. For flash parts, enter UART ISP mode.

## 1.4 Board Parameter

- Supply Voltage: 5 Volts DC (Provided by power adapter)
- Supply Current: 0.5A
- Board Size: 105cm\*75cm
- Weight: 50g

### 1.4.1 Physical Size

Figure 1-3 Board top side

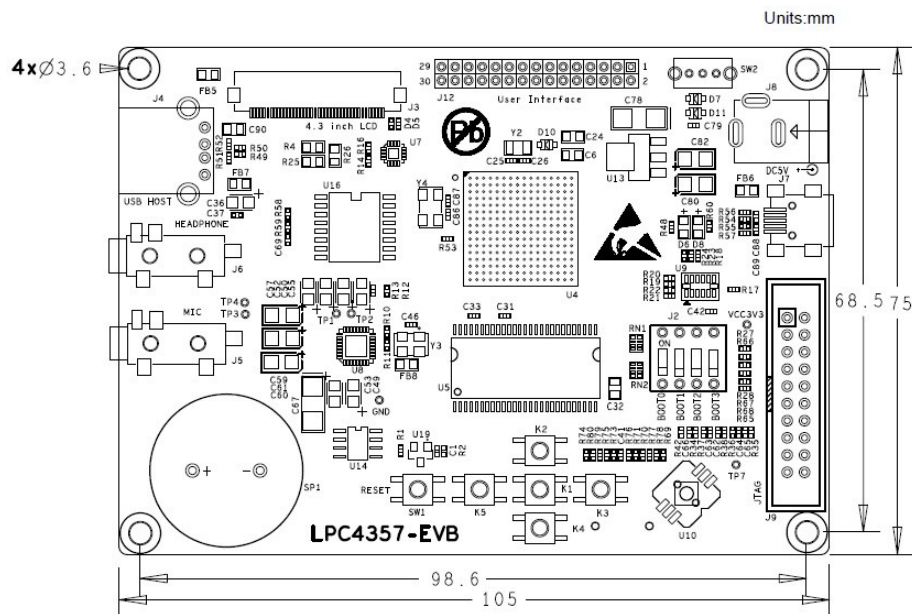


Figure 1-4 Board bottom side

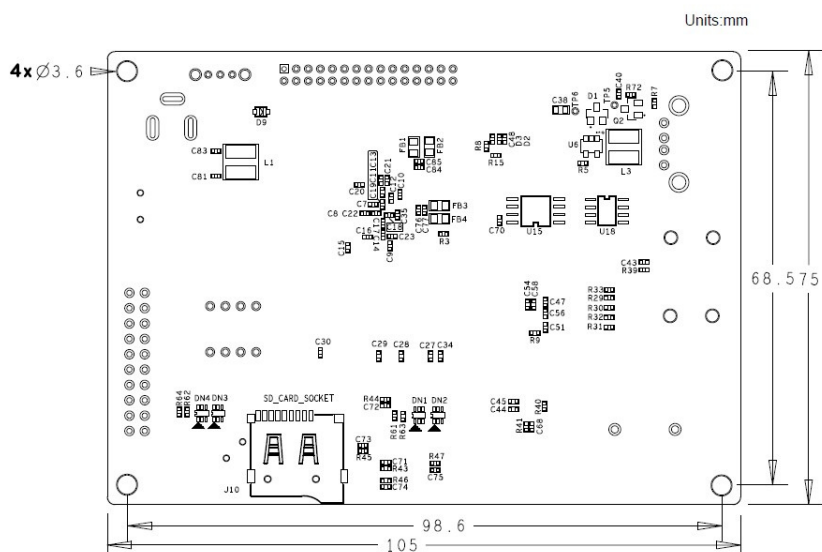
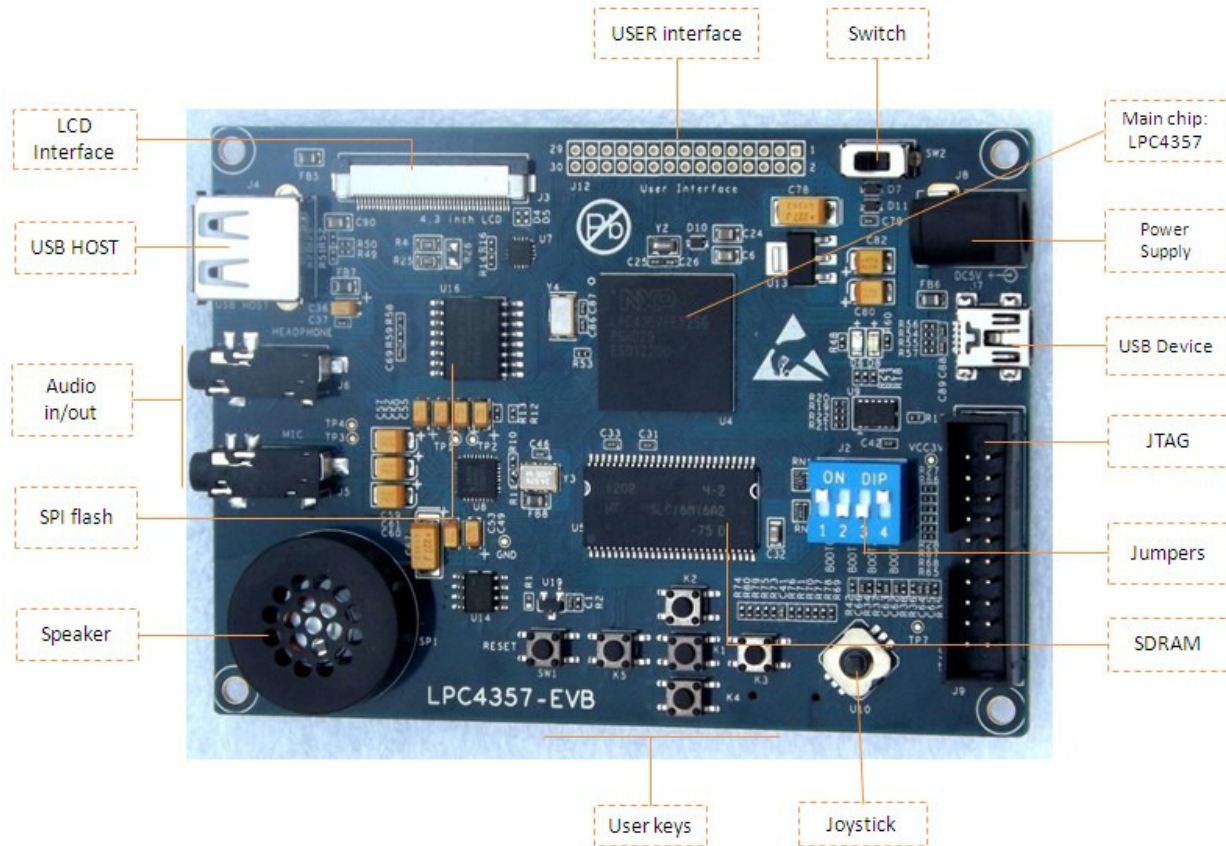


Figure 1-5: LPC4357-EVB block diagram



## 1.4.2 Peripherals:

1. LCD interface for a QVGA LCD module (sold as an optional accessory)
2. Audio codec and audio jacks
3. Two USB ports, one HS (High speed USB host) port and one FS (Full Speed device) port
4. 3-axis accelerometer from Analog Devices
5. Five user input switches
6. Joystick (Five way switch)
7. One user LED
8. One ISP switch and one reset switch
9. Boot select switch
10. Crystals for controller, RTC and audio codec
11. On board USB host power switch
12. 20-pin Cortex debug header



## 1.5 IDE and Debugger installation

- Recommended PC system configuration as below:
  - o 2.0GHz (or higher) of the CPU
  - o 512M RAM
  - o USB interfaces
  - o A serial interface
  - o Windows XP operating system
- Debugger: ULINK2 or equivalent
- IDE: KEIL MDK v4.53 Integrated Development Environment. You can download from [www.element14.com/lpcmm](http://www.element14.com/lpcmm)

The following steps show the steps for installation of Keil MDK v4.53.

Figure 1-6: Keil MDK Installation 1

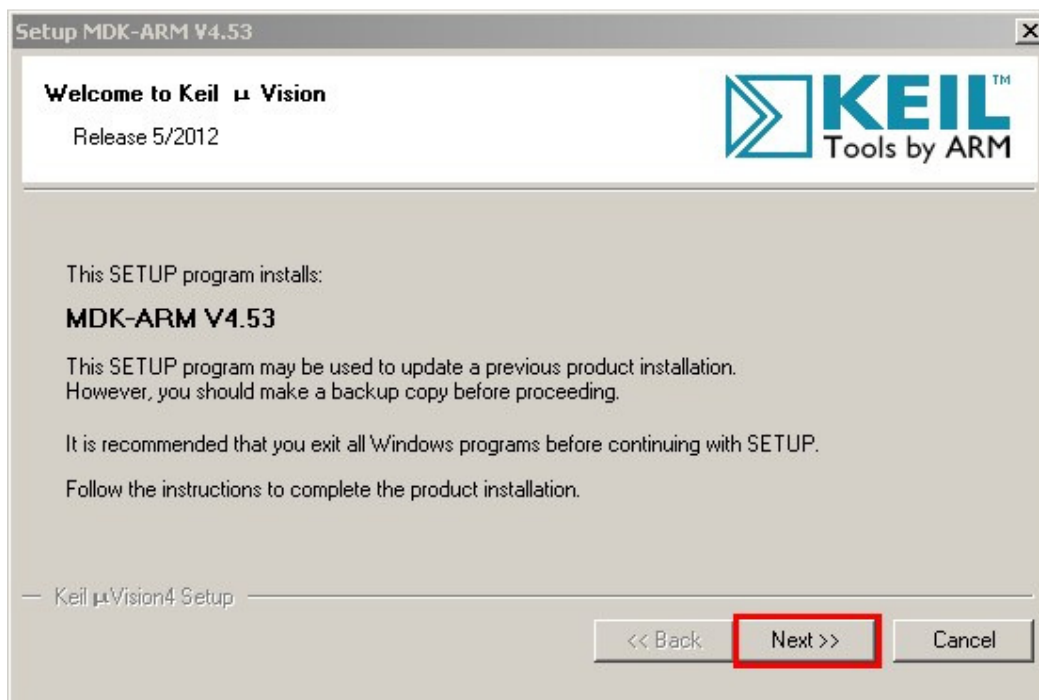


Figure 1-7: Keil MDK Installation 2

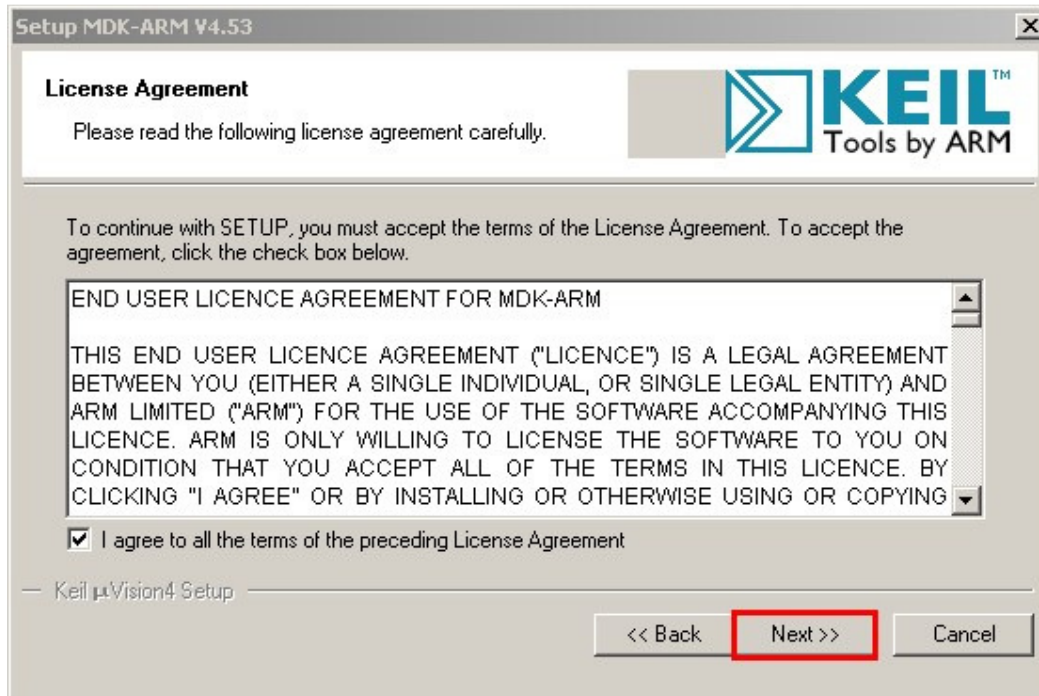


Figure 1-8: Keil MDK Installation 3

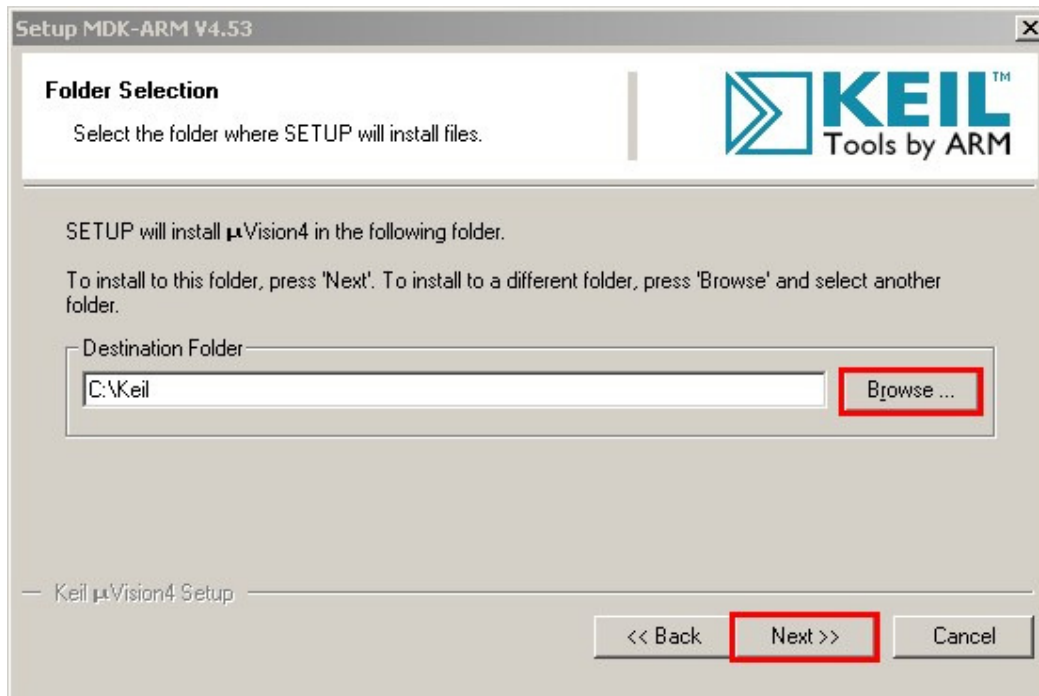


Figure 1-9: Keil MDK Installation 4

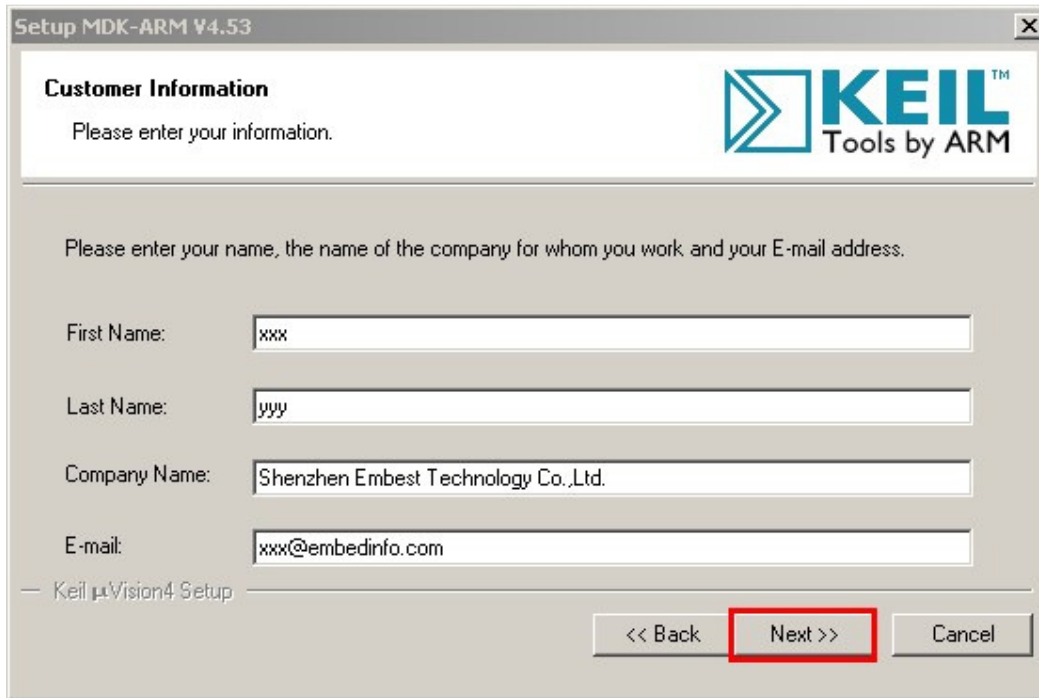


Figure 1-10: Keil MDK Installation 5

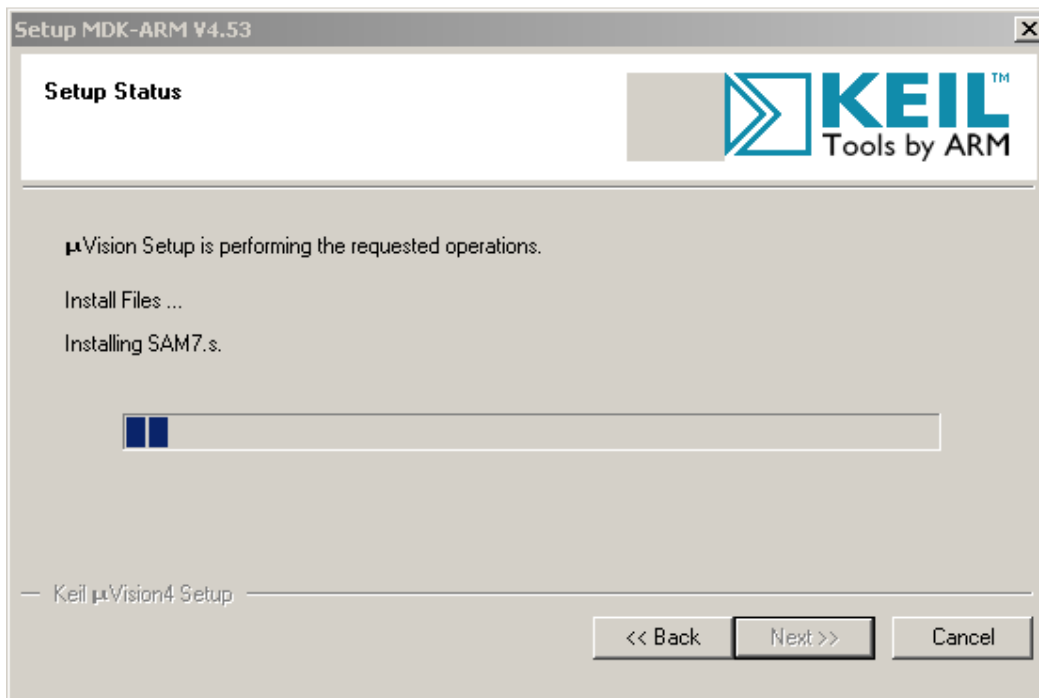


Figure 1-11: Keil MDK Installation 6

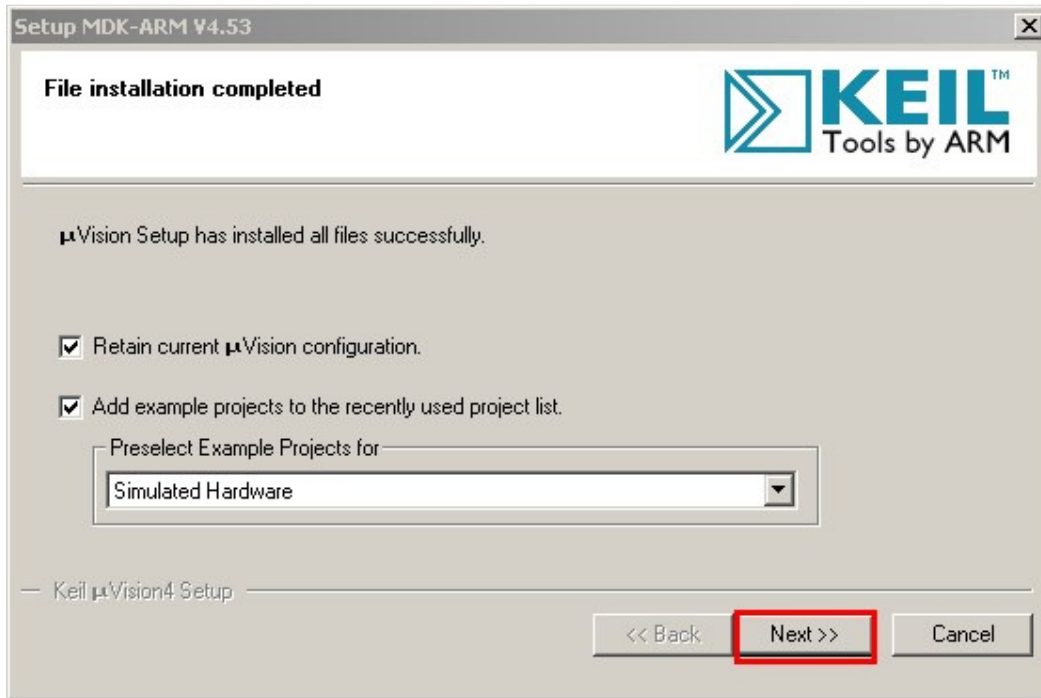
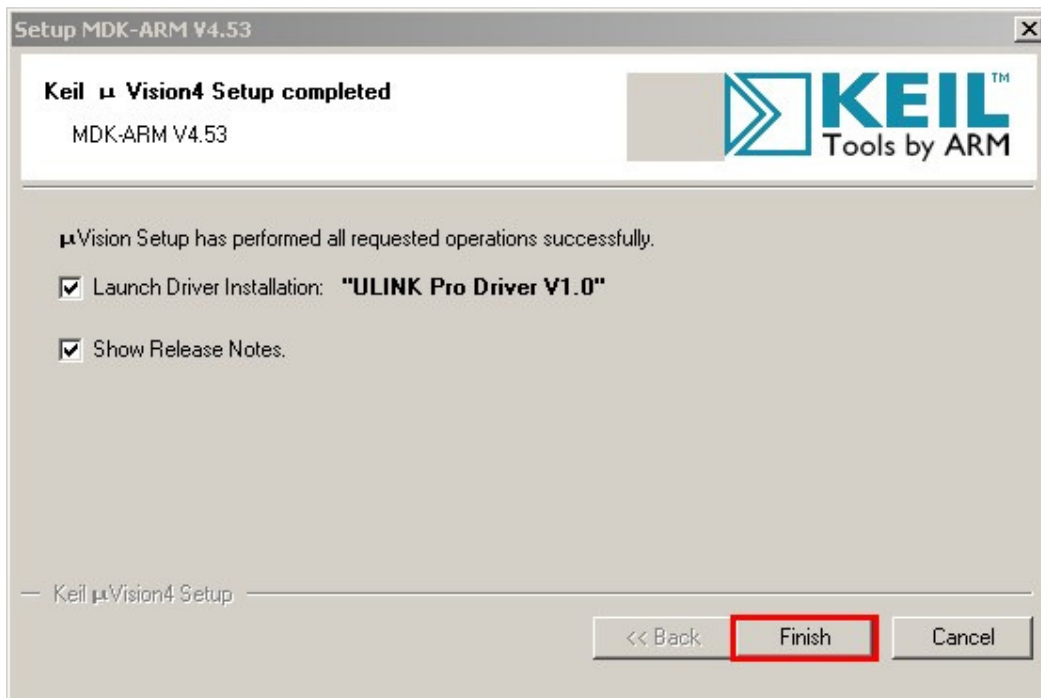


Figure 1-12: Keil MDK Installation Completed



## 2 Blinky LED Demo Example

All software and examples can be downloaded from [www.element14.com/lpcmm](http://www.element14.com/lpcmm)

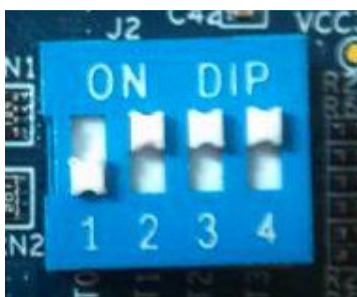
Here we choose a simple LED\_Blinky as the running demo.

- Double click to open the project. (path: \GPIO\Gpio\_LedBlinky\Keil\Gpio\_LedBlinky.uvproj)

### 2.1 Building the project

There are 3 kinds of configuration for download or debug: Inflash (internal flash memory), SPIflash (external SPI memory) and InterSRAM (Internal SRAM).

**Note:** The board boots up from internal flash as priority as compared to external memory (including SPI FLASH ). To boot from external SPI Flash, you must erase the internal flash memory and set the jumpers as below. For other boot settings please refer to [chapter 1.3](#)



For this example we will use the Inflash configuration.

Choose Inflash module, and click the “Build” button or “F7”.

Figure 2-1: InFlash configuration

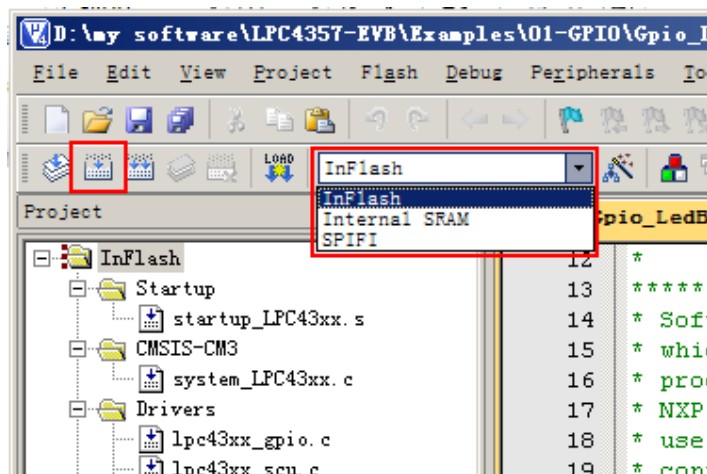


Figure 2-2: Successful Build

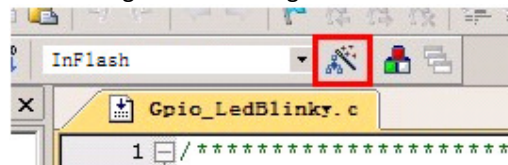
```
Build Output
compiling lpc43xx_gpio.c...
compiling lpc43xx_scu.c...
compiling lpc43xx_cgu.c...
compiling Gpio_LedBlinky.c...
linking...
Program Size: Code=3218 RO-data=1554 RW-data=84 ZI-data=2052
FromELF: creating hex file...
User command #1: D:\Keil\ARM\BIN\ElfDwT.exe .\InFlash\example.axf BASEADDRESS (0x1A000000)
ELFDWT - Signature Creator V1.1
COPYRIGHT Keil - An ARM Company, Copyright (C) 2012
*** Updated Signature over Range[32] (0x1A000000 - 0x1A000018): @0x1A00001C = 0x43FFF08A
*** Processing completed, no Errors.
".\InFlash\example.axf" - 0 Error(s), 0 Warning(s).
```

## 2.2 Download the project

### 2.2.1 Set up the Debug configuration

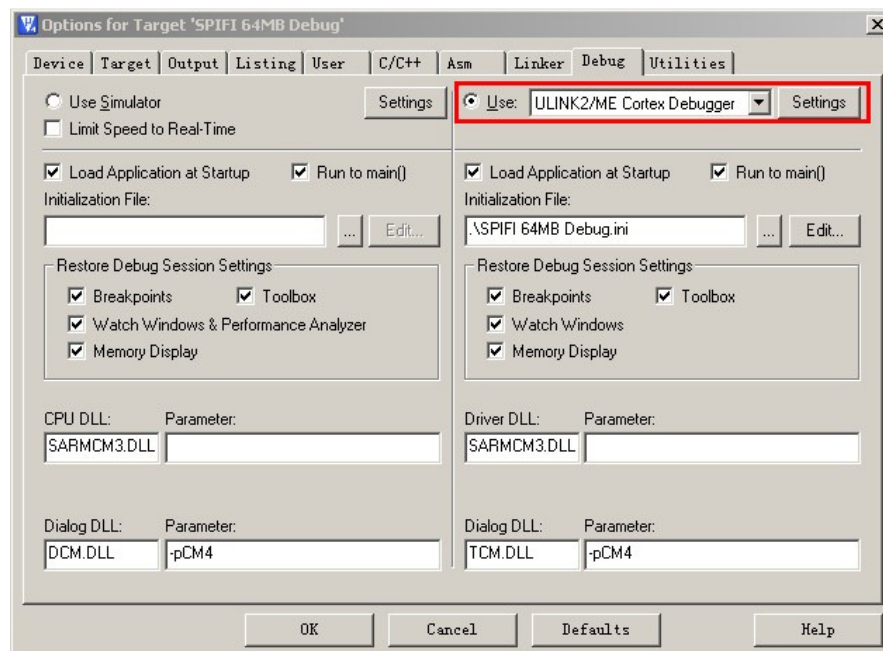
- Click on the Configuration icon as shown below

Figure 2-3: Configuration icon



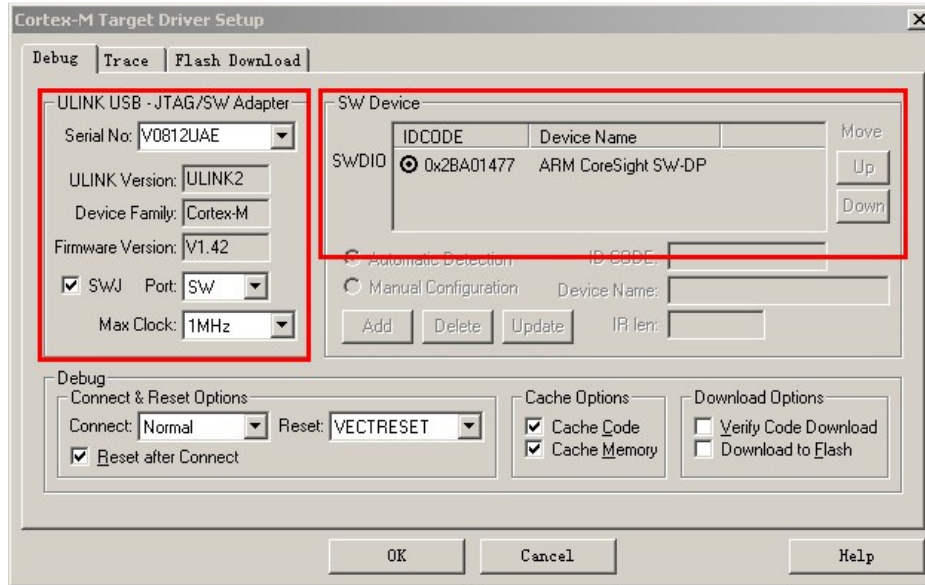
- Under the “Debug” tab; select ULINK2 as the debugger. Click on “Settings”

Figure 2-4: Debug configuration



- Under “Settings” ensure the debug adapter settings are as shown below. Click “OK” to accept.

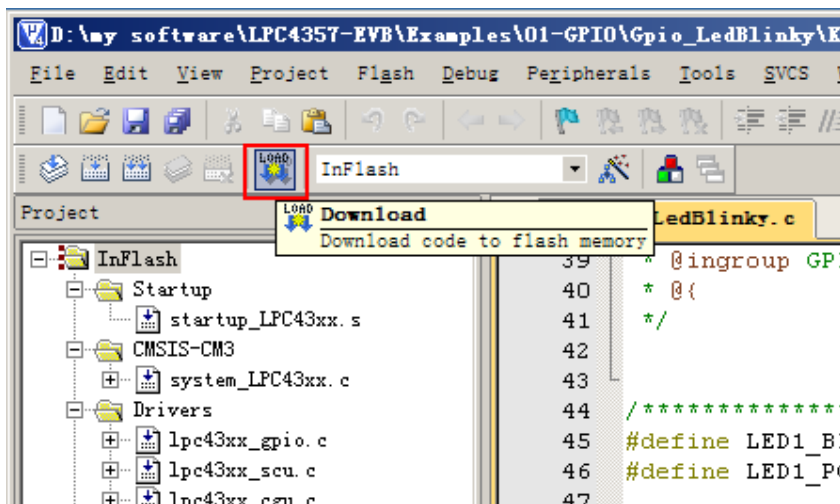
Figure 2-5: Debug Adapter Settings



## 2.2.2 Download

Download the code by clicking on the icon as shown below

Figure 2-6: Download



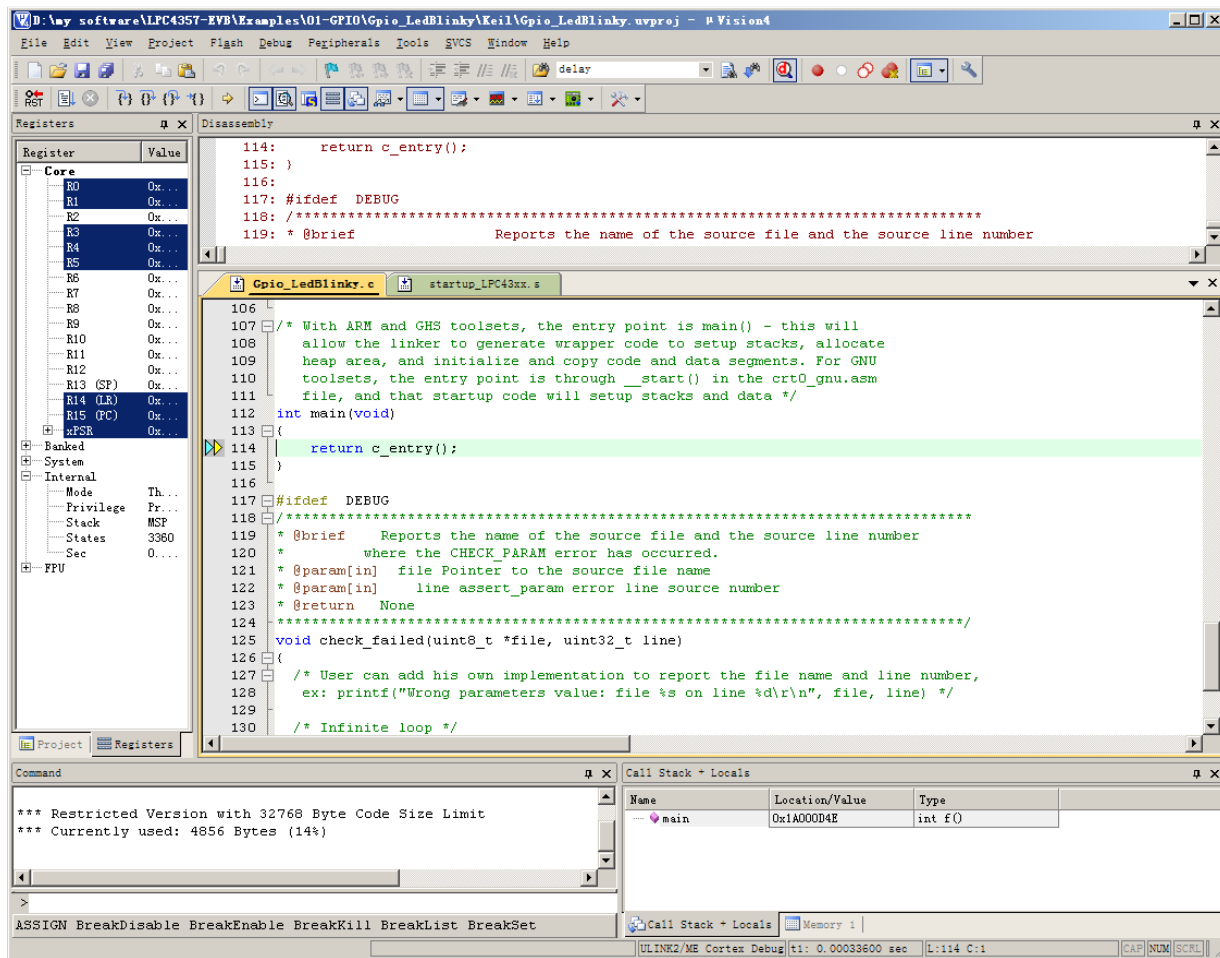
Successfully download as below:

```
Load "D:\my software\LPC4357-EVB\Examples\01-GPIO\Gpio
Erase Done.
Programming Done.
Verify OK.
```

- Reset the board. You can see the LED D6 is blinking.

### 2.2.3 Debug

Click the debug button  after building the example. As below:



The screenshot shows the Keil uVision4 IDE interface. The main window displays the source code for `Gpio_LedBlinky.c`. The code includes a `main` function that calls `return c_entry();` and a `check_failed` function. The `main` function is currently selected and highlighted in green. The `Registers` window on the left shows the core registers (R0-R15) and system registers. The `Command` window at the bottom left shows the output of the linker, indicating a restricted version with a 32768 byte code size limit and 4856 bytes currently used. The `Call Stack + Locals` window at the bottom right shows the current call stack with the `main` function at the top, located at `0x1A000D4E` with type `int f0`. The status bar at the bottom indicates the target is `ULINK2/MB Cortex Debug` and the current location is `L:114 C:1`.



### 3 Major Functional Modules

#### 3.1 Processor

The LPC43xx are ARM Cortex-M4 based microcontrollers for embedded applications which include an ARM Cortex-M0 coprocessor, up to 1 MB of flash, up to 264 KB of SRAM, advanced configurable peripherals such as the State Configurable Timer (SCT) and the Serial General Purpose I/O (SGPIO) interface, two High-speed USB controllers, Ethernet, LCD, an external memory controller, and multiple digital and analog peripherals.

The LPC43xx operate at CPU frequencies of up to 204 MHz.

The ARM Cortex-M4 is a next generation 32-bit core that offers system enhancements such as low power consumption, enhanced debug features, and a high level of support block integration. The ARM Cortex-M4 CPU incorporates a 3-stage pipeline, uses a Harvard architecture with separate local instruction and data buses as well as a third bus for peripherals, and includes an internal pre fetch unit that supports speculative branching.

The ARM Cortex-M4 supports single-cycle digital signal processing and SIMD instructions. A hardware floating-point processor is integrated in the core.

The ARM Cortex-M0 coprocessor is an energy-efficient and easy-to-use 32-bit core which is code- and tool-compatible with the Cortex-M4 core. The Cortex-M0 coprocessor, designed as a replacement for existing 8/16-bit microcontrollers, offers up to 204 MHz performance with a simple instruction set and reduced code size.

For more information, please refer to: <http://www.element14.com/community/docs/DOC-51065>



### 3.2 Memory

The LPC4357 processor features an external memory controller (EMC) and SPI Flash Interface (SPIFI). On the evaluation board, two kinds of memory devices are connected:

- One 256Mbits SDRAM using the EMC interface.
- One 256Mbits Spansion NOR Flash using the SPIFI interface

Figure 3-1: SDRAM module

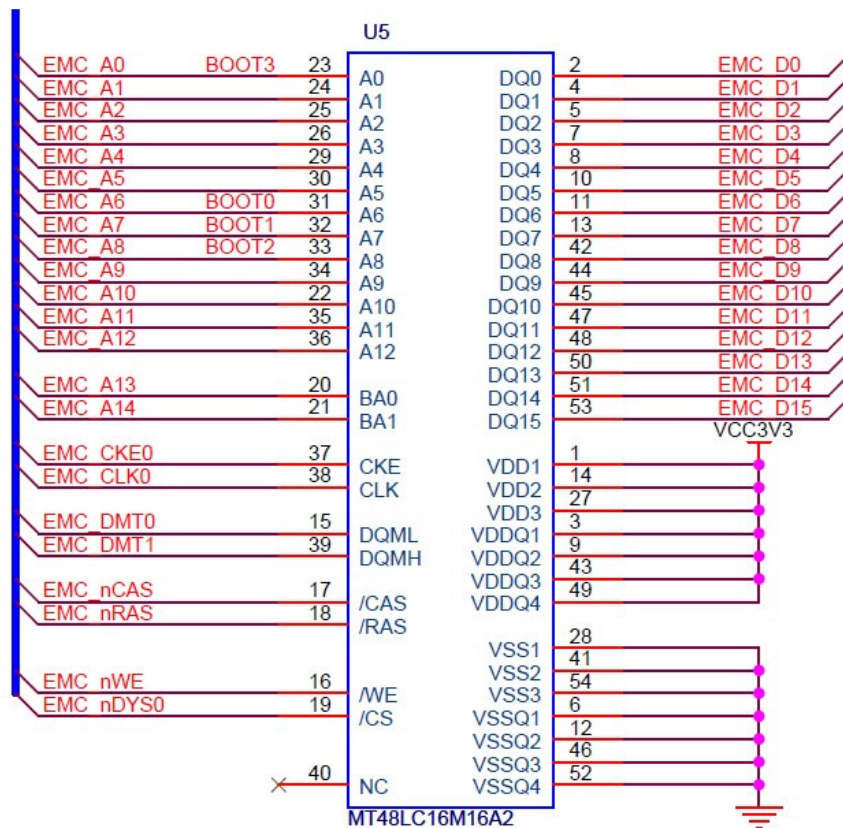
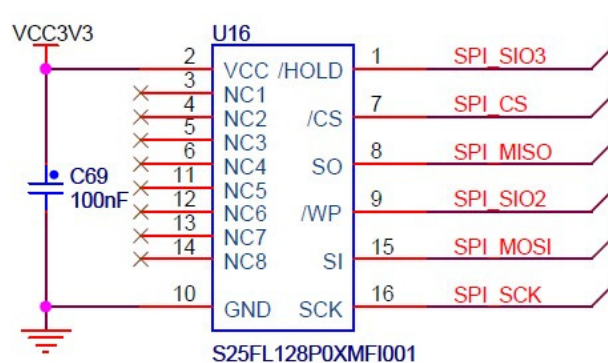
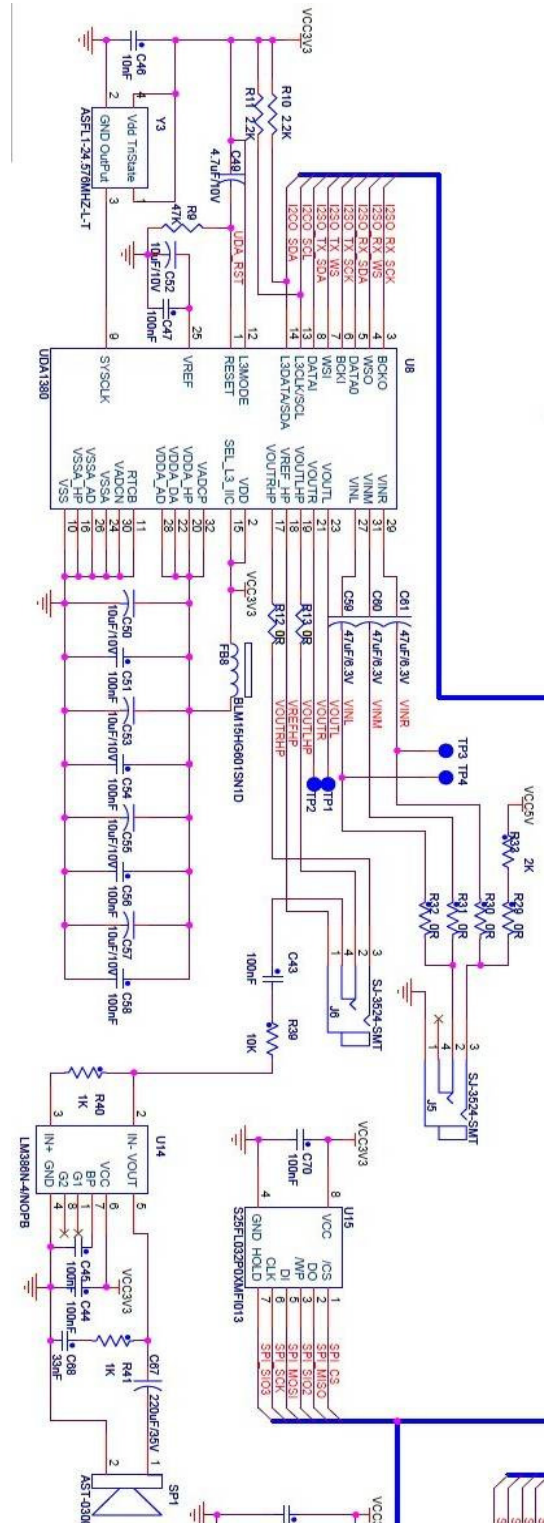


Figure 3-2: NOR Flash module



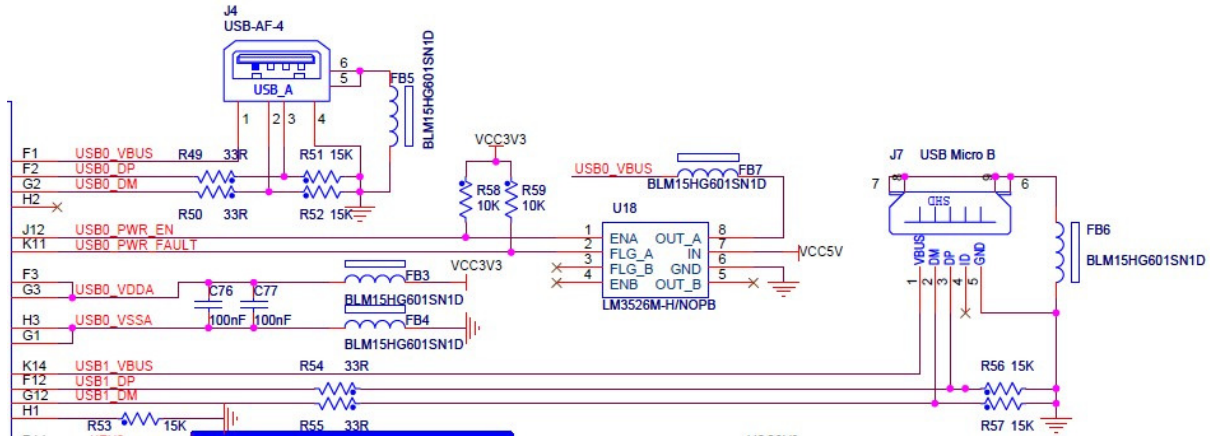
### 3.3 Audio

The processor is configured in IIS mode to interface with the stereo audio coder-decoder - [UDA1380](#). Audio input can be provided through J5, and output either on the headphone out (J6) or Speaker(SP1).

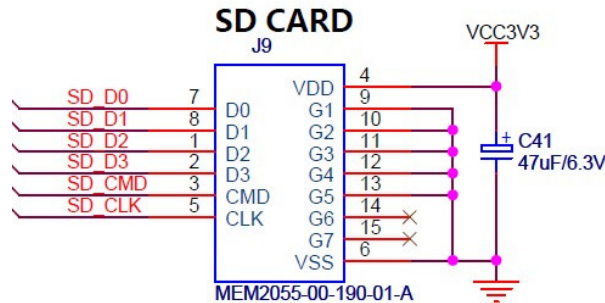


### 3.4 USBHOST and USB Device

USB0 can be configured as USB0 Host/Device/OTG mode, and the USB1 can be configured as Host/Device.

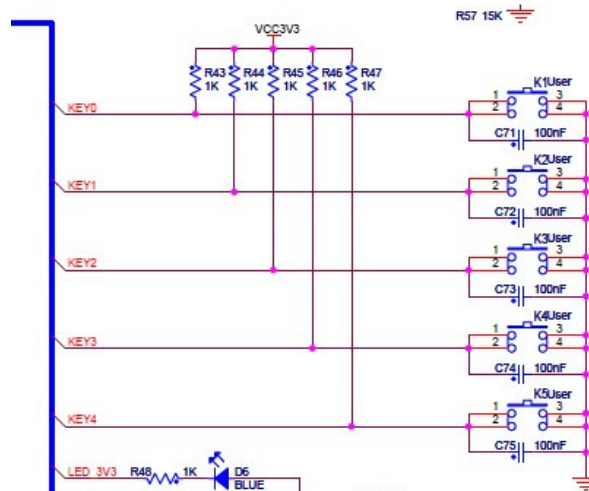


### 3.5 SD card



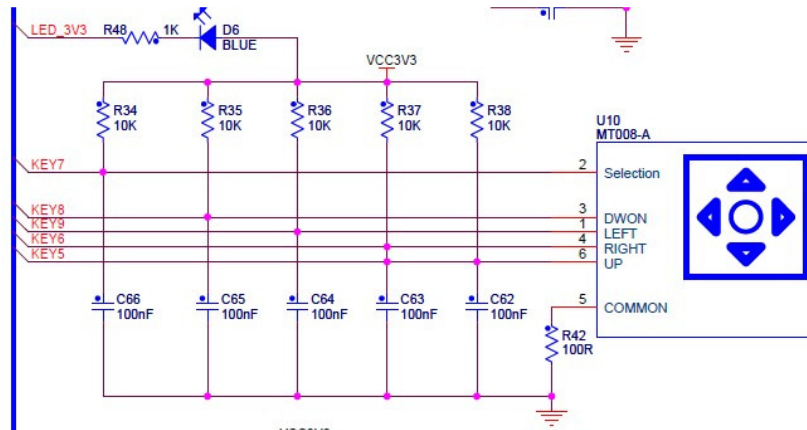
### 3.6 User keys

The board offers five user keys, controlled by GPIOs.



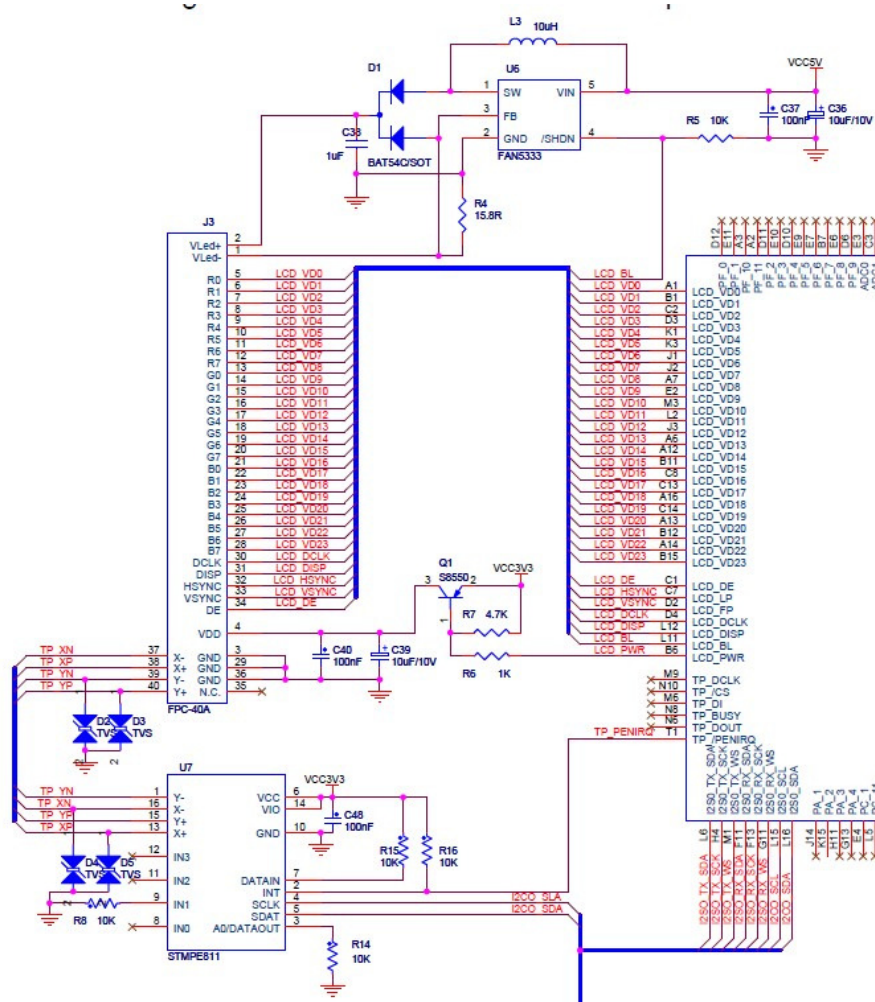
### 3.7 Joystick

The kit equips a Joystick, controlled by GPIOs.



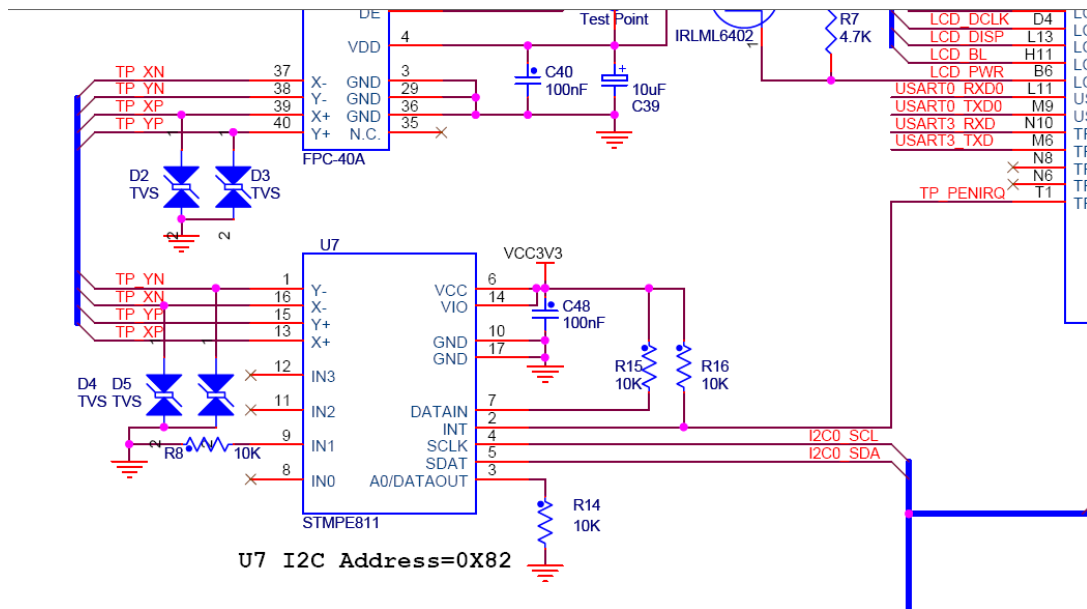
### 3.8 LCD

A 4' 480\*272 LCD is optional module. Here we offer its hardware interface.



### 3.9 Touch Screen

Touch Screen control module.



## 4 Software examples for testing Peripherals

We offer relevant software to test the hardware functionality on the board. All the projects are running in Keil-MDK v4.53. All software and examples can be downloaded from [www.element14.com/lpcmm](http://www.element14.com/lpcmm)

### 4.1 GPIO

#### 4.1.1 GPIO Key

- The project describes how to use keys through GPIO controller.
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - JTAG Debugger Connection : One end is connected to JTAG interface on the board (J9), the other to PC.
  - Turn on the switch (SW2) if you choose the power line support.
  - [Build](#) the project, and [Download](#) the image to the board
- Result:
  - Reset the board. Press any key (K1~K5 or joystick), the LED D6 is on, and off if you release.

#### 4.1.2 GPIO Led Blinky

- The project describes how to use LEDs through GPIO controller.
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - JTAG Debugger Connection : One end is connected to JTAG interface on the board (J9), the other to PC.
  - Turn on the switch (SW2) if you choose the power line support.
  - [Build](#) the project, and [Download](#) the image to the board
- Result:
  - Reset the board. The LED D6 is blinking.

### 4.2 DUAL CORE

#### 4.2.1 Int Demo

- The project demonstrates how the dual cores communicate through interruption.
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - JTAG Debugger Connection : One end is connected to JTAG interface on the board (J9), the other to PC.
  - Turn on the switch (SW2).
  - [Build](#) the project, and [Download](#) the image to the board  
**Note:** first build M0 image, and then M4. After building, download M4 image into the board.
- Result:
  - Reset the board. The LED D6 is blinking.

## 4.2.2 Mailbox Demo

- The project describes how the dual cores communicate through mailbox function.
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - Headphone connection: use a earphone to connect the board (J6).
  - USB Stick: 2GB capacity
    - Copy the psy.wav to the USB Stick root directory. (You could use your own .wav file, but ensure to rename it as psy.wav)
  - JTAG Debugger Connection : One end is connected to JTAG interface on the board (J9), the other to PC.
  - Turn on the switch (SW2).
  - [Build](#) the project, and [Download](#) the image to the board  
**Note:** first build M0 image, and then M4. After building, download M4 image into the board.
- Result:
  - Reset the board. Then insert the USB Stick, and listen to the audio output from either the speaker or the headphone.

## 4.3 I<sup>2</sup>S Audio

- The project describes how to use I2S controller to play audio files.
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - Headphone connection: use a earphone to connect the board (J6).
  - JTAG Debugger Connection : One end is connected to JTAG interface on the board (J9), the other to PC.
  - Turn on the switch (SW2)
  - [Build](#) the project, and [Download](#) the image to the board
- Result:
  - Reset the board. You can hear the warning sound from either the speaker or the headphone.

## 4.4 LCD Demo

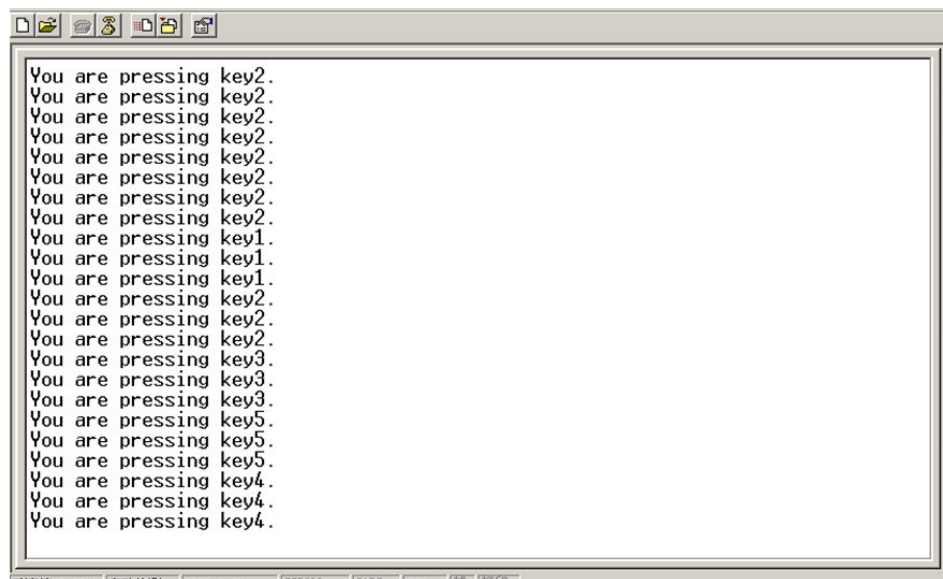
- The project describes the LCD display.
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - LCD connection: connect the optional LCD to the board.
  - JTAG Debugger Connection : One end is connected to JTAG interface on the board (J9), the other to PC.
  - Turn on the switch (SW2)
  - [Build](#) the project, and [Download](#) the image to the board
- Result:
  - Reset the board. You can see a beautiful picture displayed.



## 4.5 USB CDC

- The project describes how to make USB device as a virtual COM, working as a serial port.
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - USB device connection: connect the board to the computer through the USB device interface (J7).
  - JTAG Debugger Connection : One end is connected to JTAG interface on the board (J9), the other to PC.
  - Turn on the switch (SW2)
  - [Build](#) the project, and [Download](#) the image to the board
- Result:
  - Reset the board
  - Windows will prompt to install a driver. The .inf file (lpc18xx-vcom) lies in USB\_CDC\Keil\lpc18xx-vcom.inf
  - When installation is complete, the virtual COM will show up in the device manager.
  - Configure the terminal (eg. Hyperterminal) as below:
    - **Baud rate:** 9600; **Data bits:** 8; **Parity:** None; **Stop bits:** 1; **Flow control:** none
  - Press key K1~K5, terminal will show the information as below:

Figure 4-1: Virtual COM demo

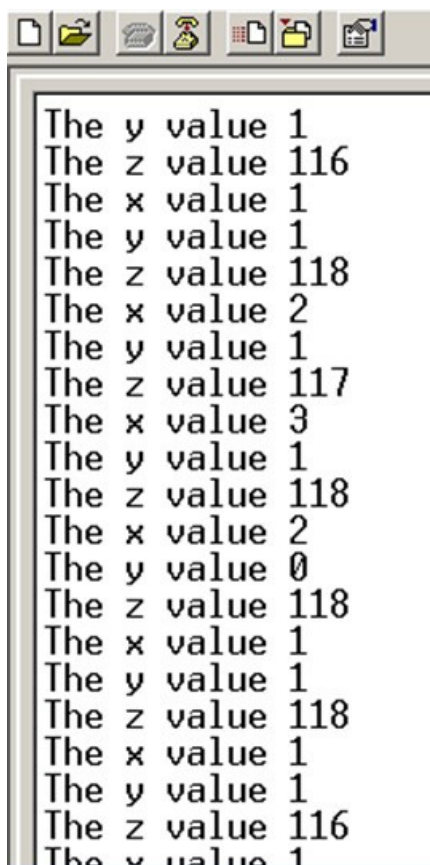


## 4.6 I<sup>2</sup>C Master: Accelerometer demo

- The project describes how to read coordinate on the ADXL345 registers.
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - USB device connection: connect the board to the computer through the USB device interface (J7).
  - JTAG Debugger Connection: One end is connected to JTAG interface on the board (J9), the other to PC .
  - Turn on the switch (SW2) if you choose the power line support.

- [Build](#) the project, and [Download](#) the image to the board
- Result:
  - Reset the board
  - Windows will prompt to install a driver. The .inf file (lpc18xx-vcom) lies in USB\_CDC\Keil\lpc18xx-vcom.inf
  - When installation is complete, the virtual COM will show up in the device manager.
  - Configure the terminal (eg. Hyperterminal) as below:
    - **Baud rate:** 9600; **Data bits:** 8; **Parity:** None; **Stop bits:** 1; **Flow control:** none
  - Rotate and move the board, terminal will show the coordinate information as below:

Figure 4-2: Accelerometer demo



#### 4.7 SD Card: Read and Write demo

- The project describes how to read and write the SD card.
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - USB device connection: connect the board to the computer through the USB device interface (J7).
  - SD card connection: put a SD card in the SD card interface (J10).
  - JTAG Debugger Connection: One end is connected to JTAG interface on the board (J9), the other to PC .
  - Turn on the switch (SW2) if you choose the power line support.
  - [Build](#) the project, and [Download](#) the image to the board

- Result:
  - Reset the board
  - Windows will prompt to install a driver. The .inf file (lpc18xx-vcom) lies in USB\_CDC\Keil\lpc18xx-vcom.inf
  - When installation is complete, the virtual COM will show up in the device manager.
  - Configure the terminal (eg. Hyperterminal) as below:
    - **Baud rate: 9600; Data bits: 8; Parity: None; Stop bits: 1; Flow control: none**
  - Press K1, terminal will show the information as below:

SD card demo is running.

the number we write in is : a

the number we read from is: a

the number we write in is : a

the number we read from is: a

.....

.....

the number we write in is : a

the number we read from is: a

the number we write in is : a

the number we read from is: a

Verify success!

## 4.8 USB HOST

- The project describes how to read and write the USB stick
- Setup:
  - Power supply: USB device (J7) or power line (J8) connection.
  - USB device connection: connect the board to the computer through the USB device interface (J7).
  - JTAG Debugger Connection: One end is connected to JTAG interface on the board (J9), the other to PC .
  - Turn on the switch (SW2) if you choose the power line support.
  - USB Host connection: put a U stick in the USB Host interface (J4).
  - Turn on the switch if you choose the power line support.
  - Build the project, as [chapter 2.5](#).
  - Download the image to the board, as [chapter 2.6](#).
- Result:
  - Reset the board
  - Windows will prompt to install a driver. The .inf file (lpc18xx-vcom) lies in USB\_CDC\Keil\lpc18xx-vcom.inf
  - When installation is complete, the virtual COM will show up in the device manager.
  - Configure the terminal (eg. Hyperterminal) as below:
    - **Baud rate:** 9600; **Data bits:** 8; **Parity:** None; **Stop bits:** 1; **Flow control:** none
  - Press K1, and insert the USB stick again. The terminal will show the information as below:

```

trew - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
Mass Storage Host Demo running.
Device Attached.
Total LUNs: 1 - Using first LUN in device.
Vendor "Generic ", Product "Flash Disk"
Mass Storage Device Enumerated.
Waiting until ready...
128
0
Retrieving Capacity...
7782399 blocks of 512 bytes.

Contents of first block:
FA B8 00 00 8E D0 BC 00 7C 8B F4 50 07 50 1F FB
FC BF 00 06 B9 00 01 F3 A5 EA 1E 06 00 00 BE BE
07 80 3C 80 74 02 CD 18 56 53 06 BB 00 7C B9 01
00 BA 00 00 B8 01 02 CD 13 07 5B 5E B2 80 72 0B
BF BC 7D 81 3D 55 53 75 02 B2 00 BF EB 06 88 15
8A 74 01 8B 4C 02 8B EE EB 15 BE 9B 06 AC 3C 00
74 0B 56 BB 07 00 B4 0E CD 10 5E EB F0 EB FE BB
00 7C B8 01 02 CD 13 73 05 BE B3 06 EB DF BE D2
06 BF FE 7D 81 3D 55 AA 75 D3 BF 24 7C BE EB 06

8A 04 88 05 8B F5 EA 00 7C 00 00 49 6E 76 61 6C
69 64 20 70 61 72 74 69 74 69 6F 6E 20 74 61 62
6C 65 00 45 72 72 6F 72 20 6C 6F 61 64 69 6E 67
20 6F 70 65 72 61 74 69 6E 67 20 73 79 73 74 65
已连接 0:01:06 自动检测 9600 8-N-1 SCROLL CAPS NUM 插 打印
  
```

## 6 ESD PRECAUTIONS AND PROPER HANDLING PROCEDURES

This section includes the precautions for mechanical handling and static precautions to be taken to avoid ESD damage:

- Avoid carpets in cool, dry areas. Leave development kits in their anti-static packaging until ready to be installed.
- Dissipate static electricity before handling any system components (development kits) by touching a grounded metal object, such as the system unit unpainted metal chassis.
- If possible, use antistatic devices, such as wrist straps and floor mats.
- Always hold a evaluation board by its edges. Avoid touching the contacts and components on the board.
- Take care when connecting or disconnecting cables. A damaged cable can cause a short in the electrical circuit.
- Prevent damage to the connectors by aligning connector pins before you connect the cable. Misaligned connector pins can cause damage to system components at power-on.
- When disconnecting a cable, always pull on the cable connector or strain-relief loop, not on the cable itself.

# X-ON Electronics

Authorized Distributor

*Click to view similar products for [NXP manufacturer](#).*

Other Similar products are found below :

[PCF8583P](#) [CWALSDVLPRFL](#) [MCIMX6QSDB](#) [74HC4050N](#) [74HC4514N](#) [74HC590PW112](#)

[MIMXRT1050EVK](#) [FRDMK66F](#) [FRDMMCLVBLDC](#) [HEF4028BPN](#) [RAPPID567XFSW](#)

[BUK7628100A118](#) [BUK9K32100EX](#) [IMXEBOOKDC4](#) [OM13017598](#)

[OM7690BFU730F598](#) [P1025RDBPA](#) [P4080DSPC](#) [LFVBBM34U1A](#) [SCC2692AC1N40](#)

[T1023RDBPC](#) [FRDMKL28Z](#) [FRDMMCLVMTR](#) [FRDMMCLVPMSM](#) [MC1378](#)