

Linux LPC17XX
BSP (Board Support Package) Guide for the
Embedded Artists LPC1788-DEV-KIT Board

Release 1.12.0

Table of Contents

1. OVERVIEW	3
2. PRODUCT CONTENTS	3
2.1. SHIPPABLE HARDWARE ITEMS	3
2.2. DOWNLOADABLE HARDWARE MATERIALS	3
2.3. DOWNLOADABLE SOFTWARE MATERIALS	3
2.4. DOWNLOADABLE DOCUMENTATION MATERIALS	3
3. SOFTWARE FUNCTIONALITY	4
3.1. SUPPORTED FEATURES.....	4
3.2. NEW AND CHANGED FEATURES	5
3.3. KNOWN PROBLEMS & LIMITATIONS.....	5
4. HARDWARE SETUP	5
4.1. HARDWARE INTERFACES	5
4.2. JUMPERS	5
4.3. BOARD CONNECTIONS	6
5. LPC1788-DEV-KIT BOARD LINUX SOFTWARE SET-UP	6
5.1. U-BOOT INSTALLATION.....	6
5.2. U-BOOT ENVIRONMENT	6
5.3. ETHERNET MAC ADDRESS.....	7
5.4. NETWORK CONFIGURATION	7
5.5. LOADING LINUX IMAGES	8
5.6. U-BOOT BUILD.....	9
6. FURTHER MATERIALS	9
7. SUPPORT	9

1. Overview

This document is a Linux LPC17XX BSP (Board Support Package) Guide for the Embedded Artists LPC1788-DEV-KIT board, Release 1.12.0.

The BSP provides a software development environment for evaluation and development of Linux on the Cortex-M3 processor core of the NXP LPC17XX microcontroller using the Embedded Artists LPC1788-DEV-KIT board as a hardware platform.

2. Product Contents

This product includes the following components.

2.1. Shippable Hardware Items

The following hardware items are shipped to customers of this product:

1. None - THIS IS A SOFTWARE-ONLY PACKAGE. Please purchase the LPC1788-DEV-KIT board from Embedded Artists or its distributors.

2.2. Downloadable Hardware Materials

The following hardware materials are available for download from Emcraft's web site to customers of this product:

1. None - Please obtain hardware materials pertinent to the LPC1788-DEV-KIT board directly from Embedded Artists.

2.3. Downloadable Software Materials

The following software materials are available for download from Emcraft's web site to customers of this product:

1. `u-boot.hex` - prebuilt U-Boot file in the format suitable for installation into embedded Flash of Cortex-M3 on the LPC1788-DEV-KIT board;
2. `networking.uImage` - prebuilt Linux image ready to be loaded to the LPC1788-DEV-KIT board;
3. `linux-LPC-1.12.0.tar.bz2` - Linux LPC17XX software development environment, including:
 - a) U-Boot firmware;
 - b) Linux kernel;
 - c) `busybox` and other target components;
 - d) Linux-hosted cross-development environment;
 - e) Framework for developing multiple projects (embedded applications) from a single installation, including sample projects allowing to kick-start software development for Linux LPC17XX.

2.4. Downloadable Documentation Materials

The following documentation materials are available for download from Emcraft's web site to customers of this product:

1. `linux-cortexm-um-1.12.0.pdf` - Linux Cortex-M User's Manual;
2. `linux-EA-LPC1788-bsp-1.12.0.pdf` - Linux LPC17XX BSP (Board Support Package) Guide for the Embedded Artists LPC1788-DEV-KIT Board (this document).

3. Software Functionality

3.1. Supported Features

The following list summarizes the features and capabilities of Linux LPC17XX, Release 1.12.0:

- U-Boot firmware:
 - U-Boot v2010.03;
 - Target initialization from power-on / reset;
 - Runs from the internal eNVM and internal SRAM (no external memory required for standalone operation);
 - Serial console;
 - Ethernet driver for loading images to the target;
 - Serial driver for loading images to the target;
 - Device driver for built-in Flash (eNVM) and self-upgrade capability;
 - Device driver for storing environment and Linux images in external Flash;
 - Autoboot feature, allowing boot of OS images from Flash or other storage with no operator intervention;
 - Persistent environment in Flash for customization of target operation;
 - Sophisticated command interface for maintenance and development of the target.
- Linux:
 - uClinux kernel v2.6.33;
 - Boot from compressed and uncompressed images;
 - Ability to run critical kernel code from integrated Flash of LPC17XX;
 - Serial device driver and Linux console;
 - Ethernet device driver and networking (`ping`, `NFS`, `Telnet`, `FTP`, `ntpd`, etc.);
 - `busybox` v1.17;
 - POSIX `pthread`s;
 - Process-to-kernel and process-to-process protection using the Memory Protection Unit (MPU) of the LPC17XX core;
 - Hardened exception handling; an exception triggered by a process affects only the offending process;
 - Loadable kernel modules;
 - Secure shell (`ssh`) daemon;
 - Web server;
 - MTD-based Flash partitioning and persistent JFFS2 Flash file system for external Flash;
 - Device driver for the DMA interface;
 - Device driver for the USB host interface;
 - SD Card device driver;
 - I²C device driver;
 - I2S device driver;
 - RTC device driver;
 - Framebuffer device driver;

- SPI controller master-mode device driver;
- Support for RS-485 in the serial device driver;
- Watchdog device driver.
- Development tools:
 - ARMv7-optimized GNU toolchain from CodeSourcery (2010q1) is used for development of U-Boot, Linux and user-space applications (toolchain must be downloaded separately from the CodeSourcery web site);
 - Cross GDB for debugging user-space applications;
 - `mkimage` tool used by the Linux kernel build process to create a Linux image bootable by U-Boot.
- Development environment:
 - Linux-hosted cross-development environment;
 - Development of multiple projects (embedded applications) from a single installation;
 - `hello` sample project ("Hello, world!" single-process configuration);
 - `networking` sample project (basic shell, networking and Flash management tools demonstration);
 - `developer` sample project (template project that can be used to jump-start development of custom user-space applications and loadable kernel modules).

3.2. New and Changed Features

This section lists new and changed features of this release:

- None.

3.3. Known Problems & Limitations

This section lists known problems and limitations of this release:

1. `CONFIG_KERNEL_IN_ENVM` requires disabling `CONFIG_ARM_UNWIND` and `CONFIG_EARLY_PRINTK`.
ID: RT 74683.
Workaround: When enabling `CONFIG_KERNEL_IN_ENVM` in the kernel, disable `CONFIG_ARM_UNWIND` and `CONFIG_EARLY_PRINTK`.
2. Debugging of multi-threaded applications using GDB is not supported.
ID: RT 77243.
Workaround: None. This issue will be resolved in future releases of the product.

4. Hardware Setup

This section explains how to set up the Embedded Artists LPC1788-DEV-KIT board.

4.1. Hardware Interfaces

For a description of the hardware interfaces provided by the LPC1788-DEV-KIT board, refer to detailed technical documentation and other associated materials available from Embedded Artists.

4.2. Jumpers

The following jumpers must be configured on the OEM Base Board:

Board	Jumper	Configuration	Notes
OEM Base Board	JP20	1-2 open, 3-4 open	To disable ISP mode to be enabled by the terminal program
OEM Base Board	JP29	1-2 open, 3-4 closed	To disconnect the DAC from the on-board speaker, in order to get rid of the noise from the speaker

4.3. Board Connections

To power the LPC1788-DEV-KIT board up, simply connect it to a PC / notebook by plugging a mini-USB cable into the J25 mini-USB connector on the base board. As soon as the connection to the PC has been made, the various on-boards LEDs should lit up, indicating that the board is up and running.

On the PC side, the USB link provides a serial console device to the LPC1788-DEV-KIT. The software installed on the board is configured for a 115.2 K terminal. On the Linux host, the serial console is available using a `/dev/ttyUSBn` device.

To provide network connectivity to the board, connect it into your LAN by plugging a standard Ethernet cable into the 10/100 Ethernet connector. The board is pre-configured with an IP address of 192.168.0.2.

5. LPC1788-DEV-KIT Board Linux Software Set-up

5.1. U-Boot Installation

Installation of the Linux LPC17XX U-Boot binary image file `u-boot.hex` for the LPC1788-DEV-KIT board is accomplished using the FlashMagic tool available from Embedded Systems Academy.

Please follow the procedure described at

<http://www.lpcware.com/content/project/lpc177x8x-u-boot-port/lpc177x8x-u-boot-deployment>.

5.2. U-Boot Environment

When the LPC1788-DEV-KIT board is reset, the Linux bootstrap will proceed to boot the U-Boot firmware from the built-in Flash printing the following output to the serial console:

```
U-Boot 2010.03-linux-cortexm-1.12.0 (Dec 06 2013 - 17:19:37)

CPU : LPC178x/7x series (Cortex-M3)
Freqs: SYSTICK=108MHz,EMCCLK=54MHz,PCLK=54MHz
Board: EA-LPC1788 rev 1
DRAM: 32 MB
Flash: 4 MB
*** Warning - bad CRC, using default environment

In:    serial
Out:   serial
Err:   serial
Net:   LPC178X_MAC
Hit any key to stop autoboot:  0
EA-LPC1788>
```

U-boot makes use of the so-called environment variables to define various aspects of the system functionality. Parameters defined by the U-boot environment variables include: target IP address, target MAC address, address in RAM where a Linux bootable images will be loaded, and many more. To examine the current settings of the environment variables, run `printenv` from the U-Boot command interface.

The reason for the warning about the "bad CRC" is that U-Boot is configured to store its environment variables in the external Flash. However, this being the first time when you boot Linux LPC17XX on the development board, obviously there is no U-Boot environment programmed to the external Flash. U-Boot goes to the external Flash, fails to find its environment there, prints the warning message and resorts to using the default environment integrated into the U-Boot image at build time.

U-Boot provides a command called `saveenv` that stores the up-to-date run-time environment to the persistent storage, which will be the external Flash for the U-Boot configuration used on the LPC1788-DEV-KIT board. You need to call `saveenv` any time when you want to copy current settings of the environment variables to the persistent storage in Flash. This is how you can write the current U-Boot environment to the external Flash:

```
EA-LPC1788> saveenv
Saving Environment to Flash...
...
EA-LPC1788>
```

Reset the LPC1788-DEV-KIT board and check that there is no warning about the bad CRC in the boot-up messages. This is expected since now U-Boot successfully finds its environment in the external Flash:

```
EA-LPC1788> reset
resetting ...

U-Boot 2010.03-linux-cortexm-1.12.0 (Dec 06 2013 - 19:43:45)
...
Hit any key to stop autoboot: 0
EA-LPC1788>
```

5.3. Ethernet MAC Address

In Linux LPC17XX, the MAC address of the Ethernet interface is defined by the `ethaddr` U-Boot environment variable. The value of the MAC address can be examined from the U-Boot command line monitor as follows:

```
EA-LPC1788> printenv ethaddr
ethaddr=C0:B1:3C:88:88:88
EA-LPC1788>
```

The default U-Boot environment for the LPC1788-DEV-KIT board sets `ethaddr` to a fixed MAC address. This address should work for you in a general case, however if you have more than two LPC1788-DEV-KIT boards in your LAN, use of the same address on multiple boards may result in packet collisions in your LAN and overall may render your LAN mal-functioning.

If you have more than one LPC1788-DEV-KIT boards in your LAN, you have to assign a unique MAC address to each board.

The MAC address can be changed by modifying the `ethaddr` variable as follows:

```
EA-LPC1788> setenv ethaddr C0:B1:3C:88:88:89
```

Don't forget to store your update in the persistent storage using `saveenv` so it is remembered across resets and power cycles.

5.4. Network Configuration

You will have to update the network configuration of your board to match settings of your local environment.

Typically, all you have to allow loading images over network from a TFTP server is update the U-Boot environment variables `ipaddr` (the board IP address) and `serverip` (the IP address of the TFTP server). Here is how it is done.

Update `ipaddr` and `serverip`:

```
EA-LPC1788> setenv ipaddr 192.168.0.2
EA-LPC1788> setenv serverip 192.168.0.1
```

and then save the updated environment to the external Flash so that your changes are persistent across resets/power cycles.

5.5. Loading Linux Images

At this point, you are able to load Linux bootable images to the board over TFTP and either boot them directly or install them to the external Flash to allow booting Linux from Flash on power-up/reset.

On the host, activate the Linux LPC17XX development environment and build the networking project:

```
-bash-3.2$ . ACTIVATE.sh
-bash-3.2$ cd projects/networking/
-bash-3.2$ make
...
-bash-3.2$
```

Copy the Linux bootable image to the TFTP download directory:

```
-bash-3.2$ cp networking.uImage /tftpboot/vlad/
-bash-3.2$
```

To load the image directly, use the `netboot` U-Boot macro:

```
EA-LPC1788> setenv image vlad/networking.uImage
EA-LPC1788> run netboot
...
TFTP from server 172.17.0.1; our IP address is 172.17.5.100
Filename 'vlad/networking.uImage'.
...
Loading: #####
#####
#####
done
Bytes transferred = 2084704 (1fcf60 hex)
...
Image Name: Linux-2.6.33-arm1
Image Type: ARM Linux Kernel Image (uncompressed)
...
Verifying Checksum ... OK
Loading Kernel Image ... OK
OK

Starting kernel ...

Linux version 2.6.33-arm1 (vlad@ocean.emcraft.com) (gcc version 4.4.1 (Sourcery G++ Lite
2010q1-189) ) #1 Fri Dec 06 15:43:44 MSK 2013
...
```

To load the image into the Flash, use the `update` U-Boot macro:

```
EA-LPC1788> setenv image vlad/networking.uImage
EA-LPC1788> run update
...
TFTP from server 172.17.0.1; our IP address is 172.17.5.100
Filename 'vlad/networking.uImage'.
...
Loading: #####
#####
#####
done
Bytes transferred = 2084704 (1fcf60 hex)
..... done
Un-Protected 32 sectors

..... done
Erased 32 sectors
```



```
Copy to Flash... done
EA-LPC1788>
```

Reset the board and verify that the newly programmed image boots on the target in the autoboot mode:

```
EA-LPC1788> reset
resetting ...

U-Boot 2010.03-linux-cortexm-1.12.0 (Dec 06 2013 - 17:19:37)
...
Starting kernel ...
...
init started: BusyBox v1.17.0 (Dec 06 2013 - 17:19:37)
~ #
```

5.6. U-Boot Build

The BSP distribution comes with U-Boot pre-built for the LPC1788-DEV-KIT board. If however you need to re-build U-Boot for your board, please follow the instructions below:

1. Install the Linux LPC17XX distribution to the development host, as described in the Linux Cortex-M User's Manual.
2. From the top of the Linux LPC17XX installation, activate the Linux LPC17XX cross-compile environment by running `. ACTIVATE.sh`.
3. Go to the U-Boot source directory (`cd u-boot/`).
4. Run the following commands:

```
[psl@pvr u-boot]$ make ea-lpc1788_config
Configuring for ea-lpc1788 board...
[psl@pvr u-boot]$ make -s
[psl@pvr u-boot]$ make -s u-boot.hex
```

6. Further Materials

Refer to *Linux Cortex-M User's Manual* for detailed information on the software architecture of the Linux LPC17XX distribution.

Visit Emcraft Systems' web site at www.emcraft.com to obtain additional materials related to Linux LPC17XX.

7. Support

We appreciate your review of our product and welcome any and all feedback. Comments can be sent directly by email to:

a2f-linux-support@emcraft.com

The following level of support is included with your purchase of this product:

- Email support for installation, configuration and basic use scenarios of the product during 3 months since the product purchase;
- Free upgrade to new releases of the downloadable materials included in the product during 3 months since the product purchase.

If you require support beyond of what is described above, we will be happy to provide it using resources of our contract development team. Please contact us for details.