

WB FIRMWARE UPDATE UTILITY PROCEDURE

Application Note

v1.1

INTRODUCTION

The goal of this document includes the following:

- Explain the firmware update procedure for the Laird WB45NBT.
- List the requirements for a remote firmware update.
- Detail the flash partition map of the WB45NBT memory.
- Provide sample code for the DHCP bootfile to be used in a remote update.

OVERVIEW

This document applies to both the WB40NBT and the WB45NBT.

The Laird WB40NBT firmware update provides the capability to change the OS image residing in a WB40NBT/WB45NBT module locally via a USB flash drive or over a wireless connection.

To boot into Linux, there are four images involved:

- Bootstrap
- U-Boot
- Linux Kernel
- Root File System

Any combination of these images may be updated using the firmware update procedure. This is accomplished with a script named *fw_update*, which resides on the WB40NBT/WB45NBT module. This script can be run in either of two different ways:

- **Automatic:** A platform init-script runs during boot-up and may call *fw_update* if triggered by a flag-file: `/tmp/platform_update_request`. This method requires a network connection so that images can be downloaded from a remote server.
- **Manual:** The *fw_update* can be invoked on the console terminal via CLI.

The *fw_update* script can use image files from locally attached media or by using a remote server on a network.

When performing an update via the network, an *update_pending* notification is sent to the server, if enabled. An Update Configuration Service, if running on the server, responds by sending configuration data back to the WB. This data is then written to the `u_boot-env` to be acted upon after reboot by the platform init-script.

An optional Post-Update procedure may be realized by including extra programs or scripts to be activated by the platform init-script via an Update Configuration Service. The platform init-script can also perform cleanup of these extra items to ensure that they are actually only run once. There is flexibility here-in for running test programs or some final configuration setup, so that the update procedure for manufacturing functions properly.

FLASH PARTITION MAP

The flash partition layout allows for two kernel images (kernel-a and kernel-b), as well as two file system images (rootfs-a and rootfs-b). This allows an alternate image to be updated without disturbing the currently running system. The currently mounted rootfs are also one of these respective a/b images. Additionally, should something go awry with the newly updated image, the previous image is still available as a fallback. The fw_update script always programs the alternate set of images.

Optionally, the fw_select script can be used to switch between the respective kernel-a/b and rootfs-a/b images. Run this script without options to see the currently running system. Use '-h' for additional options.

The WB40NBT flash partition map is as follows:

Image	Partition	Start	End	Size	MTD	Type
bootstrap.bin	bootstrap	0x00000000	0x0001FFFF	128 KB	/dev/mtd0	raw binary
u-boot.bin	u_boot	0x00020000	0x0007FFFF	384 KB	/dev/mtd1	raw binary
n/a	u_boot-env	0x00080000	0x0009FFFF	128 KB	/dev/mtd2	u-boot env
kernel.bin	kernel-a	0x000A0000	0x0029FFFF	2 MB	/dev/mtd3	kernel image
kernel.bin	kernel-b	0x002A0000	0x0049FFFF	2 MB	/dev/mtd4	kernel image
rootfs.bin	rootfs-a	0x004A0000	0x0221FFFF	29.5 MB	/dev/mtd5	UBI
rootfs.bin	rootfs-b	0x02220000	0x03F9FFFF	29.5 MB	/dev/mtd6	UBI
n/a	logs	0x03FA0000	0x03FFFFFF	384 KB	/dev/mtd7	raw

The WB45NBT flash partition map is as follows:

Image	Partition	Start	End	Size	MTD	Type
bootstrap.bin	bootstrap	0x00000000	0x0001FFFF	128 KB	/dev/mtd0	raw binary
u-boot.bin	u_boot	0x00020000	0x0007FFFF	384 KB	/dev/mtd1	raw binary
n/a	u_boot-env	0x00080000	0x0009FFFF	128 KB	/dev/mtd2	u-boot env
n/a	redund-env	0x000C0000	0x000DFFFF	128 KB	/dev/mtd3	u-boot env
kernel.bin	kernel-a	0x000E0000	0x0035FFFF	2.5 MB	/dev/mtd4	kernel image
kernel.bin	kernel-b	0x00360000	0x005DFFFF	2.5 MB	/dev/mtd5	kernel image
rootfs.bin	rootfs-a	0x005E0000	0x02BDFFFF	38.9 MB	/dev/mtd6	UBI
rootfs.bin	rootfs-b	0x02BE0000	0x051DFFFF	38.9 MB	/dev/mtd7	UBI
n/a	user	0x07FA0000	0x07FFFFFF	46.8 MB	/dev/mtd8	raw
n/a	logs	0x03FA0000	0x03FFFFFF	384 KB	/dev/mtd9	raw

FIRMWARE UPDATE METHODOLOGY

Automatic Remote Update

Note: This describes default behavior of the WB40NBT/WB45NBT firmware as supplied by Laird.

After associating with an AP, the WB40NBT/WB45NBT sends out a DHCP request. The DHCP server must respond with an IP address, and additionally a Bootfile-Name. The WB40NBT/WB45NBT detects if it has received this optional DHCP information. If it does, it proceeds with running the fw_update script using the information in the Bootfile to update its flash memory. If the md5sums of the components listed in the Bootfile match what is already on the device, then nothing is updated and the WB40NBT/WB45NBT continues to run normally.

After the updates are complete, the bootloader becomes aware of the new kernel and/or filesystem and loads these new images on the next reboot. The WB40NBT/WB45NBT reboots into the newly updated system automatically after the update process is complete.

Manual Remote Update

The fw_update script can be invoked from the CLI. Access to the CLI can be obtained either through the serial port or over a network connection through the use of SSH. By default, an SSH server is enabled on the WB40NBT/WB45NBT. The username is *root* and the password is *summit*.

When starting the fw_update script manually, the URL for the Bootfile must be supplied as an argument. For example:

```
fw_update -url http://192.168.1.10/wb40/fw.txt
```

Note: The WB40NBT/WB45NBT requires a valid IP address before it can fetch the updates from the network. If the module is configured as a bridge (which does not have an IP address), then it must first be configured with a valid IP address on its network interface before running the update program. This can be done using 'udhcpc -i eth1', or by using ifconfig.

After the fw_update script has completed its process, the bootloader becomes aware of the new kernel and/or filesystem. The new images are loaded on the next reboot. Issue the 'reboot' command to reboot the WB40NBT/WB45NBT. When invoked manually, the fw_update script does NOT automatically reboot the module after it has completed updating the device.

FIRMWARE UPDATE LIST (BOOTFILE)

The fw.txt file contains a list of image URLs and their respective MD5SUMs. This file can be supplied via a server's DHCP bootfile option, for remote updates.

The fw_update script tries to use each listed file in this list. Any lines beginning with a '#' are "commented-out" and are ignored.

The format is as follows:

```
http://192.168.1.10/wb40/bootstrap.bin, 06c73bb2c38f6bd0ecde4b2308595adc  
http://192.168.1.10/wb40/u-boot.bin, c55870dce9de3347950987b135a9fcd3  
http://192.168.1.10/wb40/kernel.bin, 5f7f5ebb4717250cdf565e20049debbe  
http://192.168.1.10/wb40/rootfs.bin, 479d3828243c3712efd9a14ecfe9b96e
```

It is not required to list all four images in this file. If only a subset of these images is required to be upgraded, then only list those particular image files in this file. Each line has two elements:

- URL for each image file
- MD5SUM of each image file

REMOTE UPDATE REQUIREMENTS

To utilize the Remote Update capability of the WB40NBT/WB45NBT, three pieces of hardware must be in place:

- Wi-Fi Access Point
- Windows or Linux server running DHCP and either HTTP or FTP
- WB40NBT/WB45NBT unit to be updated

Wi-Fi Access Point

If this is the first time the WB40NBT/WB45NBT module is being updated, then the AP must be configured for open access. This is so the module can obtain a wireless network connection without having been pre-programmed with

a specific SSID and password. Ensure that there are no other APs around that the module may inadvertently attach to during this update procedure.

Subsequent updates can be arranged to occur with a prescribed, secured AP, but the corresponding profile must first be created in the unit using the CLI.

Windows or Linux Server

The server needs to offer DHCP and either HTTP or FTP file services.

The DHCP server needs to be configured to send an additional parameter called a 'DHCP Option Parameter'^[1]. The DHCP Option specifies a 'Bootfile-Name' (DHCP option #67). The Bootfile-Name is a URL of a text file that further describes the images to be updated (the 'Bootfile').

The fw_update script uses the 'wget' utility to fetch the images from the file server over the network. It supports both HTTP and FTP protocols.

LIMITATIONS

- This updating method is destructive to existing data on the device, as existing images in the flash memory of the WB module, are overwritten by new images.
- There should only be one DHCP server present, when using the automatic Remote Update method.
- Any interruption during the procedure can result in a failed update, and possibly in a WB module that does not boot (the sam-ba utility can be used to recover).
- Currently, the update procedure cannot successfully write the bootstrap image, for the WB45NBT (this issue is being worked on).

REFERENCES

[1] <http://www.iana.org/assignments/bootp-dhcp-parameters/bootp-dhcp-parameters.xml>

REVISION HISTORY

Revision	Date	Description	Approved By
1.0	24 Oct 2013	Initial Release	John Imboden
1.1	27 April 2016	Removed references to <i>connecting to an open AP by default</i> in the Automatic Remote Update section.	Mark Calhoun

© Copyright 2016 Laird. All Rights Reserved. Patent pending. Any information furnished by Laird and its agents is believed to be accurate and reliable. All specifications are subject to change without notice. Responsibility for the use and application of Laird materials or products rests with the end user since Laird and its agents cannot be aware of all potential uses. Laird makes no warranties as to non-infringement nor as to the fitness, merchantability, or sustainability of any Laird materials or products for any specific or general uses. Laird, Laird Technologies, Inc., or any of its affiliates or agents shall not be liable for incidental or consequential damages of any kind. All Laird products are sold pursuant to the Laird Terms and Conditions of Sale in effect from time to time, a copy of which will be furnished upon request. When used as a tradename herein, *Laird* means Laird PLC or one or more subsidiaries of Laird PLC. Laird™, Laird Technologies™, corresponding logos, and other marks are trademarks or registered trademarks of Laird. Other marks may be the property of third parties. Nothing herein provides a license under any Laird or any third party intellectual property right.