SHORT RANGE

ELLA-W1 Wi-Fi Direct

Setting up and testing Wi-Fi Direct in ELLA-W1 series

Application Note

Abstract

This document describes how to setup and test Wi-Fi Direct in the ELLA-W1 series modules using Marvell proprietary drivers and wpa_supplicant.

Document Information		
Title	ELLA-W1 Wi-Fi Direct	
Subtitle	Setting up and testing Wi-Fi Direct in ELLA-W1 series	
Document type	Application Note	
Document number	UBX-15014432	
Revision, date	R02	02-July-2015
Document status	Advance Information	

Document status information	
Objective Specification	Document contains target values. Revised and supplementary data will be published later.
Advance Information	Document contains data based on early testing. Revised and supplementary data will be published later.
Early Production Information	Document contains data from product verification. Revised and supplementary data may be published later.
Production Information	Document contains the final product specification.

This document applies to the following products:

Product name	Type number	Firmware version	PCN / IN
ELLA-W1 series		**Industrial driver release** Package: SD-UAPSTA-BT-FM-8787-FC18-MMC-14.66.35.p57-M3X14484_AX-GPL Firmware version: 14.66.35.p57 Driver version: M3X14484	N/A

Contents

1 Overview

Wi-Fi Direct or Wi-Fi P2P is a standard defined by the Wi-Fi Alliance that enables devices to connect to each other without requiring a dedicated wireless access point. Instead, it lets the devices negotiate which side will take over the access point (AP) role (also referred to as Soft-AP) in a P2P group. The device acting as AP is called the P2P Group Owner (P2P GO), whereas the client devices are known as P2P clients.

To establish a P2P group, two P2P devices must discover each other and negotiate their roles. Once this is complete, they set up a secure communication using Wi-Fi Protected Setup (WPS) and finally the P2P GO is required to run a Dynamic Host Configuration Protocol (DHCP) server to provide the P2P clients with IP addresses. Once the P2P group is established, other P2P clients can join the group and even legacy Wi-Fi clients can connect to the P2P GO similar to traditional IEEE802.11 infrastructure network.

2 Setting up and testing

This section provides information on how to setup and test Wi-Fi Direct in the ELLA-W1 series modules using Marvell proprietary drivers and wpa_supplicant.

2.1 Prerequisites

2.1.1 Marvell drivers

The Linux reference drivers support the following two driver API options:

- Old Linux wireless extensions (WEXT) interface
- New cfg80211 configuration API

Instructions on how to compile the Marvell reference drivers can be found in the EVK-ELLA-W1 User Guide [1]. Make sure to compile the driver with the cfg80211 and Wi-Fi Direct options enabled in the Makefile (default configuration):

```
CONFIG_STA_SUPPORT=y
CONFIG_UAP_SUPPORT=y
CONFIG_WIFI_DIRECT_SUPPORT=y
```

The Linux kernel should be at least version 3.2 or *compat-wireless* (now named *backports*) is required to provide a recent version of the cfg80211 API.

The Wi-Fi Direct feature has been tested to work with the following driver and kernel versions:

Driver/firmware version	Linux kernel version
14.66.35.p57-M3X14484-GPL-(FP66)	3.17.4
14.66.35.p57-M3X14484-GPL-(FP66)	3.0.35 + compat-wireless-3.2.5-1

Currently, the Wi-Fi Direct using wpa_supplicant does not work with the Marvell automotive driver release based on the instructions from this application note. However, it has been successfully tested with proprietary Marvell tools.

2.1.2 wpa_supplicant

The Wi-Fi Direct functionality has been tested to work with wpa_supplicant versions >=2.0. To enable P2P support in the wpa_supplicant, the following configuration options should be set in the wpa_supplicant `.config` file:

```
CONFIG_CTRL_IFACE=y
CONFIG_DRIVER_NL80211=y
CONFIG_WPS=y
CONFIG_WPS2=y
CONFIG_P2P=y
CONFIG_AP=y
```

2.2 Testing

For setting up a Wi-Fi Direct connection between two devices that has an ELLA-W1 module connected to each of them, follow the steps mentioned below on both the devices unless mentioned otherwise:

1. Load the driver and make sure that cfg80211 for AP and STA and enhanced P2P are enabled:

   ```
   modprobe sd8787 cfg80211_wext=0xf p2p_enh=1
   ```

2. Prepare a configuration file `p2p.conf` for wpa_supplicant with the following settings:

   ```
   ctrl_interface=/var/run/wpa_supplicant
   device_name=ELLA1
   ```

```
device_type=10-0050F204-1
p2p_no_group_iface=1
config_methods="keypad push_button virtual_display"
persistent_reconnect=1
p2p_go_ht40=1
country=US

# optional, can be useful for monitoring, forces
# wpa_supplicant to use only channel 1 rather than
# 1, 6 and 11:
#p2p_listen_reg_class=81
#p2p_listen_channel=1
#p2p_oper_reg_class=81
#p2p_oper_channel=1

# If you need to modify the group owner intent, 0-15, the higher
# number indicates preference to become the GO. You can also set
# this on p2p_connect commands.
#p2p_go_intent=15
```

3. Run *wpa_supplicant* with the configuration file on both the devices:

```
wpa_supplicant -i wfd0 -D nl80211 -c p2p.conf -B
```

4. Run *wpa_cli* on both the sides and start P2P device discovery:

```
wpa_cli
> p2p_find
OK
```

5. The devices should discover each other and you should see something as shown below:

5a. Device 1:

```
<3>P2P-DEVICE-FOUND 02:06:c6:43:85:3d p2p_dev_addr=02:06:c6:43:85:3d pri_dev_type=10-
0050F204-1 name='ELLA2' config_methods=0x188 dev_capab=0x25 group_capab=0x0
```

5b. Device 2:

```
<3>P2P-DEVICE-FOUND 02:06:c6:47:df:7b p2p_dev_addr=02:06:c6:47:df:7b pri_dev_type=10-
0050F204-1 name='ELLA1' config_methods=0x188 dev_capab=0x25 group_capab=0x0
```

6. Now you can issue a *p2p_connect* on both sides to the MAC address of the other device using WPS Push Button Configuration and the desired GO intent. The devices will start the group owner negotiation process and once this is completed successfully, the P2P GO will start an AP and the WPS provisioning phase, so that the P2P Client can securely connect to it. The full logs for both sides should be as shown in the following example:

6a. Device 1 – P2P GO:

```
> p2p_find
OK
<3>P2P-DEVICE-FOUND 02:06:c6:43:85:3d p2p_dev_addr=02:06:c6:43:85:3d pri_dev_type=10-
0050F204-1 name='ELLA2' config_methods=0x188 dev_capab=0x25 group_capab=0x0
p2p_connect 02:06:c6:43:85:3d pbc go_intent=10
> OK
<3>P2P-FIND-STOPPED
<3>P2P-GO-NEG-SUCCESS
<3>CTRL-EVENT-CONNECTED - Connection to 02:06:c6:47:df:7b completed [id=0 id_str=]
<3>WPS-ENROLLEE-SEEN 02:06:c6:43:85:3d f01edef3-ce65-53d3-8efb-1d82df4ae65b 10-0050F204-1
0x2388 4 1 [ELLA2]
```

```
<3>CTRL-EVENT-EAP-STARTED 02:06:c6:43:85:3d
<3>CTRL-EVENT-EAP-PROPOSED-METHOD vendor=0 method=1
<3>CTRL-EVENT-EAP-PROPOSED-METHOD vendor=14122 method=254
<3>WPS-REG-SUCCESS 02:06:c6:43:85:3d f01edef3-ce65-53d3-8efb-1d82df4ae65b
<3>P2P-GROUP-FORMATION-SUCCESS
<3>P2P-GROUP-STARTED wfd0 GO ssid="DIRECT-l6" freq=2412 passphrase="1WFLZW0r"
go_dev_addr=02:06:c6:47:df:7b
<3>WPS-SUCCESS
<3>CTRL-EVENT-EAP-FAILURE 02:06:c6:43:85:3d
<3>WPS-ENROLLEE-SEEN 00:00:00:00:00:00 f01edef3-ce65-53d3-8efb-1d82df4ae65b 10-0050F204-1
0x2388 0 0 [ELLA2]
<3>AP-STA-CONNECTED 02:06:c6:43:85:3d p2p_dev_addr=02:06:c6:43:85:3d
```

6b. Device 2 – P2P Client:

```
> p2p_find
OK
<3>P2P-DEVICE-FOUND 02:06:c6:47:df:7b p2p_dev_addr=02:06:c6:47:df:7b pri_dev_type=10-
0050F204-1 name='ELLA1' config_methods=0x188 dev_capab=0x25 group_capab=0x0
<3>P2P-GO-NEG-REQUEST 02:06:c6:47:df:7b dev_passwd_id=4
p2p_connect 02:06:c6:47:df:7b pbc go_intent=7
> OK
<3>P2P-FIND-STOPPED
<3>P2P-GO-NEG-SUCCESS
<3>CTRL-EVENT-SCAN-RESULTS
<3>WPS-AP-AVAILABLE-PBC
<3>Trying to associate with 02:06:c6:47:df:7b (SSID='DIRECT-l6' freq=2412 MHz)
<3>Associated with 02:06:c6:47:df:7b
<3>CTRL-EVENT-EAP-STARTED EAP authentication started
<3>CTRL-EVENT-EAP-PROPOSED-METHOD vendor=14122 method=1
<3>CTRL-EVENT-EAP-METHOD EAP vendor 14122 method 1 (WSC) selected
<3>WPS-CRED-RECEIVED
<3>WPS-SUCCESS
<3>P2P-GROUP-FORMATION-SUCCESS
<3>CTRL-EVENT-EAP-FAILURE EAP authentication failed
<3>CTRL-EVENT-DISCONNECTED bssid=02:06:c6:47:df:7b reason=3 locally_generated=1
<3>CTRL-EVENT-SCAN-RESULTS
<3>WPS-AP-AVAILABLE-PBC
<3>Trying to associate with 02:06:c6:47:df:7b (SSID='DIRECT-l6' freq=2412 MHz)
<3>Associated with 02:06:c6:47:df:7b
<3>WPA: Key negotiation completed with 02:06:c6:47:df:7b [PTK=CCMP GTK=CCMP]
<3>CTRL-EVENT-CONNECTED - Connection to 02:06:c6:47:df:7b completed [id=0 id_str=]
<3>P2P-GROUP-STARTED wfd0 client ssid="DIRECT-l6" freq=2412
psk=9299586149f553c6fa78170589ebd0f48052cbe5211a1cdc56f2e9295d209199
go_dev_addr=02:06:c6:47:df:7b
```

2.3 Assigning IP addresses

In Wi-Fi Direct, the IP addresses are assigned using the Dynamic Host Configuration Protocol. This means, that the P2P GO is required to start a DHCP server in order to provide the P2P clients with the IP addresses. In Linux for example, the packages *udhcp* (DHCP server/client) or *dnsmasq* (DHCP server) can provide such services. For the sake of simplicity, static IP addresses are assigned to the devices in this example. To do this, quit wpa_cli on both sides and use *ifconfig* to configure IP addresses for the *wfd0* network interfaces, for example, on device 1:

```
ifconfig wfd0 169.254.1.1
```

Device 2:

```
ifconfig wfd0 169.254.1.2
```

Now the devices should be able to ping each other.

2.3.1 Dynamic IP address configuration

The following example shows how to use the DHCP server/client included in Busybox, for dynamically assigning an IP address to the P2P client. On the P2P group owner side, create a configuration file for the DHCP server:

```
# udhcpd configuration file (udhcpd-wfd0.conf)
start 192.168.3.8
end 192.168.3.255
interface wfd0
pidfile /var/run/udhcpd-wfd0.pid
lease_file /var/lib/misc/udhcpd-wfd0.leases
option subnet 255.255.255.0
option lease 864000 # default: 10 days
```

Configure the local wfd0 network interface and start the DHCP server with:

```
ifconfig wfd0 192.168.3.1
udhcpd -S udhcpd-wfd0.conf
```

On the P2P client, start the DHCP client to assign an IP address to the P2P network interface with:

```
udhcpc -i wfd0
```

3 Performance measurements

3.1 Throughput

For measuring the performance of the Wi-Fi Direct link between the two devices in terms of TCP data throughput, the *iperf* network testing tool was used. The tests were carried out with two ELLA-W163-A evaluation kits on:

1. Texas Instruments Sitara AM3352 (ARM Cortex A8) platform running Linux 3.17.4 and
2. Freescale i.MX6 Solo (ARM Cortex A9) platform running Linux 3.0.35 with compat-wireless-3.2.5-1

The Wi-Fi Direct was set up according to the description above and the devices were connected to each other in the 5 GHz band with support for 802.11n high-throughput data rates (*ht40*) enabled. The maximum PHY data rate of the modules is 150 Mbits/s.

☞ Currently, the Linux 3.17.4 platform supports 802.11g data rates only as a P2P GO. To enable high-throughput data rates, the group owner intents were adjusted accordingly for this test, so that the other device becomes P2P GO.

The *iperf* tool was run on one device in server mode and the other device connected to it with four parallel TCP streams to measure the upload data throughput. Then, the iperf client/server roles were switched and the download throughput was measured.

The results of the *iperf* measurements are shown below:

```
# iperf -c 169.254.1.2 -P4 -t30
------------------------------------------------------------
Client connecting to 169.254.1.2, TCP port 5001
TCP window size: 16.0 KByte (default)
------------------------------------------------------------
[  3] local 169.254.1.1 port 57181 connected with 169.254.1.2 port 5001
[  5] local 169.254.1.1 port 57183 connected with 169.254.1.2 port 5001
[  4] local 169.254.1.1 port 57182 connected with 169.254.1.2 port 5001
[  6] local 169.254.1.1 port 57184 connected with 169.254.1.2 port 5001
[ ID] Interval       Transfer     Bandwidth
[  5]  0.0-30.0 sec  42.6 MBytes  11.9 Mbits/sec
[  6]  0.0-30.0 sec  42.8 MBytes  11.9 Mbits/sec
[  3]  0.0-30.1 sec  42.6 MBytes  11.9 Mbits/sec
[  4]  0.0-30.1 sec  42.5 MBytes  11.9 Mbits/sec
[SUM]  0.0-30.1 sec   170 MBytes  47.6 Mbits/sec
```

Listing 1: TCP throughput P2P GO to client

```
# iperf -c 169.254.1.1 -P4 -t30
------------------------------------------------------------
Client connecting to 169.254.1.1, TCP port 5001
TCP window size: 43.8 KByte (default)
------------------------------------------------------------
[  4] local 169.254.1.2 port 52708 connected with 169.254.1.1 port 5001
[  6] local 169.254.1.2 port 52710 connected with 169.254.1.1 port 5001
[  3] local 169.254.1.2 port 52707 connected with 169.254.1.1 port 5001
[  5] local 169.254.1.2 port 52709 connected with 169.254.1.1 port 5001
[ ID] Interval       Transfer     Bandwidth
[  4]  0.0-30.0 sec  47.0 MBytes  13.1 Mbits/sec
[  5]  0.0-30.0 sec  47.2 MBytes  13.2 Mbits/sec
[  6]  0.0-30.1 sec  46.9 MBytes  13.1 Mbits/sec
```

```
[  3]  0.0-30.1 sec  47.1 MBytes  13.1 Mbits/sec
[SUM]  0.0-30.1 sec   188 MBytes  52.5 Mbits/sec
```

Listing 2: TCP throughput P2P client to GO

3.2 Connection time

To measure the time taken to create a connection between the two P2P devices, the Wi-Fi Direct was set up using wpa_supplicant, with timestamps enabled in the logging output. The connection time was extracted by calculating the difference between recorded P2P events from the log files. The measured time include pure P2P connection establishment only and no higher layer protocol setup times are included.

Measurements for DHCPv4 network configuration using udhcp (busybox 1.22.1) were done separately and yielded an additional average setup time of **2.06 seconds** based on three consecutive measurements.

3.2.1 P2P connection between two ELLA-W1 series modules

The following devices were used for this measurement:

1. Texas Instruments Sitara AM3352 (ARM Cortex A8) platform running Linux 3.17.4 with an ELLA-W161
2. Fujitsu Lifebook S7220 (Intel Core2 Duo) running Linux 3.16.7 with an ELLA-W163

Configuration (commands applicable to wpa_supplicant's CLI):

- Connection roles (GO, client) were suggested by both peers using the go_intent option of the p2p_connect command (wpa_cli)
- GO (2) was not looking actively for Wi-Fi Direct peers, but it was discoverable for clients (p2p_listen)
- Wi-Fi Direct client was actively looking for new connections (p2p_find)

Average connection time in seconds based on five consecutive measurements is shown in the following table:

Events recorded on GC	Average time (s)
P2P-DEVICE-FOUND	1.17
P2P-GO-NEG-SUCCESS	0.63
P2P-GROUP-FORMATION-SUCCESS	1.51
P2P-GROUP-STARTED	0.27
SUM	**3.58**

3.2.2 P2P connection between ELLA-W1 series and a consumer device

Devices used:

1. Texas Instruments Sitara AM3352 (ARM Cortex A8) platform running Linux 3.17.4 with an ELLA-W161
2. Samsung GalaxyTab7

ELLA-W1 as GO

Configuration:

- GO: (1)
- GC: (2)

Average connection times in seconds, based on five consecutive measurements:

Events recorded on GC	Average time (s)
P2P-GO-NEG-SUCCESS	0.19
P2P-GROUP-FORMATION-SUCCESS	4.83
P2P-GROUP-STARTED	0.00
AP-STA-CONNECTED	0.27
SUM	**5.29**

ELLA-W1 series as GC

Configuration:

- GC: (1) looking for GO
- GO: (2)

Average connection times in seconds, based on five consecutive measurements:

Events recorded on GC	Average time (s)
P2P-DEVICE-FOUND	2.26
P2P-GO-NEG-SUCCESS	2.87
P2P-GROUP-FORMATION-SUCCESS	1.69
P2P-GROUP-STARTED	0.30
SUM	**7.12**

Appendix

A List of acronyms

Abbreviation / Term	Explanation / Definition
AP	Access Point
API	Application Programming Interface
CLI	Command Line Interface
DHCP	Dynamic Host Configuration Protocol
EAP	Extensible Authentication Protocol
GC	Group Client
GO	Group Owner
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
MAC	Medium Access Control
P2P	Peer to Peer
PHY	Physical Layer Device
STA	Station
TCP	Transmission Control Protocol
WEXT	Wireless Extensions
Wi-Fi	Wireless Local Area Networking
WPS	Wi-Fi Protected Setup

Related documents

[1] EVK-ELLA-W1 User Guide, Docu.No. UBX-15012877

Revision history

Revision	Date	Name	Status / Comments
R01	26-May-2015	mzes	Initial release
R02	2-July-2015	mzes	Included section 2.3.1-Dynamic IP address configuration. Updated sections 2.2 and 3.2.1.

Contact

For complete contact information visit us at www.u-blox.com.

u-blox Offices

North, Central and South America

u-blox America, Inc.
Phone: +1 703 483 3180
E-mail: info_us@u-blox.com

Regional Office West Coast:
Phone: +1 408 573 3640
E-mail: info_us@u-blox.com

Technical Support:
Phone: +1 703 483 3185
E-mail: support_us@u-blox.com

Headquarters
Europe, Middle East, Africa

u-blox AG
Phone: +41 44 722 74 44
E-mail: info@u-blox.com
Support: support@u-blox.com

Asia, Australia, Pacific

u-blox Singapore Pte. Ltd.
Phone: +65 6734 3811
E-mail: info_ap@u-blox.com
Support: support_ap@u-blox.com

Regional Office Australia:
Phone: +61 2 8448 2016
E-mail: info_anz@u-blox.com
Support: support_ap@u-blox.com

Regional Office China (Beijing):
Phone: +86 10 68 133 545
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office China (Chongqing):
Phone: +86 23 6815 1588
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office China (Shanghai):
Phone: +86 21 6090 4832
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office China (Shenzhen):
Phone: +86 755 8627 1083
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office India:
Phone: +91 959 1302 450
E-mail: info_in@u-blox.com
Support: support_in@u-blox.com

Regional Office Japan (Osaka):
Phone: +81 6 6941 3660
E-mail: info_jp@u-blox.com
Support: support_jp@u-blox.com

Regional Office Japan (Tokyo):
Phone: +81 3 5775 3850
E-mail: info_jp@u-blox.com
Support: support_jp@u-blox.com

Regional Office Korea:
Phone: +82 2 542 0861
E-mail: info_kr@u-blox.com
Support: support_kr@u-blox.com

Regional Office Taiwan:
Phone: +886 2 2657 1090
E-mail: info_tw@u-blox.com
Support: support_tw@u-blox.com