How to use a Toshiba FlashAir W-04 Wi-Fi SD card with the Mistica FPGA16

The point of this text is showing how a Toshiba FlashAir W-04 SD card can be used with the Mistica FPGA16. We will setup the card and will map it like a network drive in our laptop. Although the Mistica FPGA16 device is used in the entire text, any other device can be used.

Requirements:

- Mistica FPGA16
- Toshiba FlashAir W-04 card
- Laptop with Windows 10
- Local WPA2-PSK secured Wi-Fi network and Internet access
- FlashAirFWUpdateToolV4 tool

Before we start:

- Write down the SD card MAC address. This info is printed in the back side of the card. Example: **EC21E5563AC2**
- Write down the name (aka SSID) and the password of our local Wi-Fi network. Example: **MI-WIFI** and **p4sW00rdW1F1**
- Search for an unused IP in the local network address range. Example: **192.168.1.16** with mask **255.255.255.0**
- Write down the IP address of the local router (aka default gateway). Example: **192.168.1.1**
- Download the latest version of the FlashAirFWUpdateToolV4 tool. As I write these lines it's the 4.00.04 (dated on 1 August 2019) and can be downloaded from: https://jp.toshiba-memory.com/ww/support/download/flashair/w04/update02.htm

Steps:

1. We will prepare the SD card to be used with our FPGA. We will delete any pre-existing partitions and will create a new single one with the máximum size of the card that we will format using FAT32.

   Inside Windows 10, we will open the command prompt (pressing Win + X) and then will select **Command Prompt (Admin)**

We write **diskpart** and once inside DiskPart:

1. **list disk** to identify the disk where Windows find de SD card.
2. **select disk X** where X is the disk number, example: 1.
3. **clean** to delete all the card pre-existing content and partitions.
4. **list partition** to check thant no other partition still exists.
5. **create partition primary** to create a new partition using the full size of the SD card.
6. **list partition** to make sure that the partition has been successfully created and has the correct size. In my 16GB card, 14GB are shown.
7. **select partition 1** to access the partition.
8. **format quick fs=fat32 label=”W-04"** to format using FAT32 file system and aasign a label to the card.
9. **active** to actívate the partition.
10. **exit** to end the DiskPart tool.

At this point we should see the SD card in the Windows 10 file explorer.

2. We are going to créate the base config for the card. The easiest way is running the fimware updater because, in addition to updating, it install only the needed files we are going to use later. We can do this even if the card already has the latest firmware version.

   We open **FlashAirFWUpdateToolV4** and then:

   1. Select the network drive letter where tha SD card is (example: "D:\")
   2. Safely eject the SD card and press **Accept**. Insert the card again. The tool will analyse the card and confirm the current firmware version.
   3. Will be asked to update. We press **Accept**.
   4. For 3 times we will be asked to safely eject the SD card and insert it back again. This process will update the firmware, will copy the needed files and will check the card status.

3. We are going to configure the card. Originally the SD has 2 folders (**DCIM** and **SD_WLAN**, the second one hidden) and a file named **fwupdate.fbn** in the root path. Inside the **DCIM** folder there is another folder named **100__TSB** which contains the **FA000001.JPG** file. Inside the **SD_WLAN** folder there is a text file called **CONFIG** and that’s where the configuration is located.

We open the **CONFIG** file with the Notepad (or any other similar editor).

We will see this:

```
[Vendor]

CIPATH=/DCIM/100__TSB/FA000001.JPG
APPMODE=4
APPNETWORKKEY=********
VERSION=F15DBW3BW4.00.04
CID=02544d535731364754d2ad007b011b01
PRODUCT=FlashAir
VENDOR=TOSHIBA
```

[Vendor] marks the start of the section where the internal card paramaters are defined.

CIPATH shows the path where the photo camera folder feature is located. We leave this as is.

APPMODE selects if the card works as an isolated Wi-Fi access point where the other devices connect to access the files (APPMODE=4) or as a Wi-Fi device client which connects to the current Wi-Fi network (APPMODE=5). We want the second option so will change the value to 5.

APPNETWORKKEY contains the password of the isolated Wi-Fi network that the SD card broadcasts (if APPMODE=4) or the password used lo join our local Wi-Fi network (if APPMODE=4). The password must be written in plain text. However when we open back the CONFIG file, it will be shown encrypted (********) for security reasons.

VERSION, **CID**, **PRODUCT** and **VENDOR** contain information about the version, model and manufacturer of the card. We leave this as is.

Now we have to add some additional parameters.

Inside the **[Vendor]** section:

APPSID contains the name (aka SSID) of the Wi-Fi network we the SD card will connect. Example: APPSID=MI-WIFI

LOCK is used to tell the card to run the *welcome script* after it reboots next time. LOCK=0 is for running it and LOCK=1 is to avoid it. In our scenario: LOCK=1

DNSMODE defines where the own SD card's DNS resolution queries must be sent. DNSMODE=0 will send them to the IP defined in the **[WLANSD]** section (which is explained bellow). DNSMODE=1 will send them to the IP received from the DHCP server (usually the router). In our scenario, as we are going to configure an static IP address to be able to connect later via WebDAV, wil use DNSMODE=0.

APPAUTOTIME contains the inactivity time-out period (in milliseconds) after which the card's Wi-Fi is disabled for energy saving. In our scenario will use APPAUTOTIME=3600000 (1 hour).

MASTERCODE is the MAC address of the FlashAir SD card. Example: MASTERCODE=EC21E5563AC2

APPNAME defines the hostname of the card when working in server mode. It's unused when Works as a client (which is our case). Anyway, we will assign a name. Example: APPNAME=myflashair

UPLOAD is used to decide if files will be allowed to be uploaded to the SD card (UPLOAD=1) or we will only allow downloading from the card (UPLOAD=0). In our scenario will use UPLOAD=1

UPDIR indicates the path to the root folder of the SD card where files will be copied if UPLOAD=1. We will use: UPDIR=/

WEBDAV enables the WebDAV protocol, which is needed to map the card as a network drive under Windows 10. If WEBWAV=2, the protocol will run in read+write mode, which is just we want. So for us: WEBDAV=2

NOISE_CANCEL enables the air noise reduction feature. This will enhance stability but at the expense of lowering the card's Wi-Fi signal strenght. For our scenario we will make sure that there is a good Wi-Fi signal level in the location where the Mistica FPGA16 is, and then: NOISE_CANCEL=2

Now we will create a new configuration section called [WLANSD] where all the parameters related to the network will be defined.

Inside **[WLANSD]**:

ID defines the name that the card will have when running in client mode. Example: ID=FAW04

DHCP_Enabled decides if the rest of network parameters will be assigned from the a DHCP server, usually the router (DHCP_Enabled=YES) or if they will be defined inside the configuration file (DHCP_Enabled=NO). In our scenario will use DHCP_Enabled=NO

IP_Address defines an static IP address to be used by the card. In our example: IP_Address=192.168.1.16

Subnet_Mask defines the network mask. For us: Subnet_Mask=255.255.255.0

Default_Gateway is used to define the IP address of the router (or default gateway). In our scenario: Default_Gateway=192.168.1.1

Preferred_DNS_Server and **Alternate_DNS_Server** define the DNS servers that the SD card will use. We will configure the the Google DNS servers. So: Preferred_DNS_Server=8.8.8.8 y Alternate_DNS_Server=8.8.4.4

In summary, this is the final configuration our **CONFIG** file should have:

```
[WLANSD]

ID=FAW04
DHCP_Enabled=NO
IP_Address=192.168.1.16
Subnet_Mask=255.255.255.0
Default_Gateway=192.168.1.1
Preferred_DNS_Server=8.8.8.8
Alternate_DNS_Server=8.8.4.4

[Vendor]

CIPATH=/DCIM/100__TSB/FA000001.JPG
APPMODE=5
APPSSID=MI-WIFI
APPNETWORKKEY=p4sW00rdW1F1
VERSION=F15DBW3BW4.00.03
CID=02544d535731364754d2ad007b011b01
PRODUCT=FlashAir
```

```
VENDOR=TOSHIBA
LOCK=1
DNSMODE=0
APPAUTOTIME=300000
MASTERCODE=EC21E5563AC2

APPNAME=myflashair
UPLOAD=1
UPDIR=/
WEBDAV=2
NOISE_CANCEL=2
```

4. Before we continue, we will make sure that the FlashAir SD cards is connected to our network. So we insert tha card in the Mistica FPGA16 and from the Windows 10 command prompt throw a ping to the IP address defined in the CONFIG file (in our example, the 192.168.1.16).

   If we get successfully replies will mean everything's OK:

```
C:\Users\bplcon0>ping 192.168.1.16

Pinging 192.168.1.16 with 32 bytes of data:
Reply from 192.168.1.16: bytes=32 time=3ms TTL=255
Reply from 192.168.1.16: bytes=32 time=2ms TTL=255
Reply from 192.168.1.16: bytes=32 time=2ms TTL=255
Reply from 192.168.1.16: bytes=32 time=5ms TTL=255

Ping statistics for 192.168.1.16:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximun = 5ms, Average = 3ms
```

5. To end, we will map the SD card as a network unit in Windows 10. To do it, we open the file explorer. In the left side of the window we right-click **This PC** and select the option **Map network drive…** from the dropdown menu.

   In **Drive:** we select a letter for the network unit from the dropdown menu.
   In **Folder:** we write *http://<FLASHAIR_CARD_IP>/.* In our example: http://192.168.1.16/
   We check the **Reconnect at sign-in** option and uncheck **Connect using different credentials**.
   We press **Finish**.

   And that's all. We are done. The Wi-Fi SD card performs now as a regular network unit. In the Windows 10 file explorer will be shown with the label **DavWWWRoot**.

   In the case we find any problems with the above, we can try this to fix the Windows 10 WebDAV issues:

   1. Open the command prompt (pressing Win + X) and select **Run**. We write **services.msc** and press INTRO.
   2. In the right column, search for **WebClient** and double-click it. Then we select **Automatic** from **Startup type:** dropdown menu.
   3. Close the window.
   4. Ppen the command prompt (pressing Win + X) and select **Run**. We write **regedit** and press INTRO.
   5. In the left column, search for **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WebClient\Parameters** and we select it.
   6. In the right column, search for **BasicAuthLevel**, double-click it and change the value to **2**.
   7. Close the window.
   8. Reboot Windows 10.

Special greetings to the spanish **MiST/Mistica/SiDi Telegram group** and also to the **forofpga.es** and **retrowiki.es** forums communities.