

Mawlana Bhashani Science and Technology University

Department of Information and Communication Technology

3rd Year 2nd nd Semester, B.Sc. (Engg.) in ICT

Course Title: Telecommunication Engineering

Course Code: ICT-4101

Name: Zaber All Khaled ID: IT-17057

Assignment Name: Zodiac OpenFlow Switch.

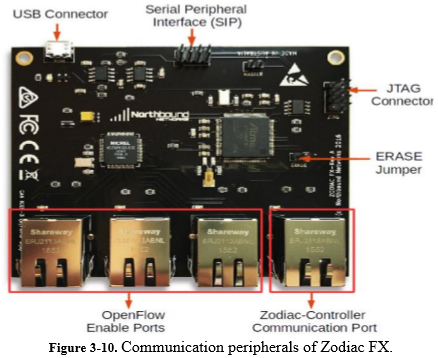
Assignment No: 01

Zodiac FX is the first OpenFlow switch designed to sit in a desk, not in a datacenter. Until now the power of Software Defined Networking (SDN) was only available to the administrators of large corporate networks. Even though there are numerous free or open source SDN controllers the one thing that was missing was a small, affordable OpenFlow switch. In this demo, we present Zodiac FX the world’s smallest OpenFlow Software Defined Network Switch.

The Zodiac FX is a 4 port network development board designed for

* Hobbyists
* Students
* Researchers
* embedded developers
* or anyone who requires a low cost network development platform.

Even though it was initially designed to allow affordable access to OpenFlow enabled hardware it’s open source firmware it can be used in any number of other applications. By providing the firmware source code users are free to not only create their own versions but also use it as a basis for a completely different type of device.



Static IP Addressing:

With static IP addressing, addresses are assigned manually, and have to be provisioned carefully so that each device has its own address—with no overlap. When we connect a new device, we would have to select the "manual" configuration option and enter in the IP address, the subnet mask, the default gateway and the DNS server(s).

Dynamic Host Configuration Protocol (DHCP): DHCP takes all of the manual work out of IP addressing. Generally, the device that's at the "top" of we home network—whether it's a standalone firewall or a router/gateway device or we Control home controller—will provide DHCP by default as a service on the network.

When DHCP is enabled, a new device connected to the network asks the DHCP server for an address, and the server assigns one from its pool of unused locations. The server itself tracks which addresses are used and which addresses are available, and keeps a record of which addresses have been assigned to the various devices. This ensures that addresses don't conflict with each other. However, it also means that, if a device goes offline, when it reconnects it may not have the same IP address it had before.

Mixing Configurations: It's entirely possible to mix static IP and DHCP addressing schemes. Since the default DHCP address range is between 100 and 149, we'll want to avoid all of the addresses between 192.168.1.100 and 192.168.1.149 when we're assigning static IP addresses. That leaves the ranges from 2-99 and from 150-254 wide open, which is usually plenty for most home networks.

Virtual Local Area Network (VLAN):

A VLAN is a group of devices on one or more LANs that are configured to communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments. There are two main reasons for the development of VLANs:

1. the amount of broadcast traffic .

2. increased security Broadcast traffic increases in direct proportion to the number of stations in the LAN.

The goal of the virtual LAN (VLAN) is the isolation of groups of users so that one group is not interrupted by the broadcast traffic of another. By segregating a group of devices to a particular VLAN, a switch will block broadcasts from devices in that VLAN to devices that are not in that VLAN instead of flooding it out every port.

VLANs also have the benefit of added security by separating the network into distinct logical networks. Traffic in one VLAN is separated from another VLAN as if they were physically separate networks. If traffic is to pass from one VLAN to another, it must be routed. Each VLAN is identified by a VLAN ID (VID), which is usually a number. They can reside on only a single switch, or they can be distributed throughout the entire network on each switch. Each VLAN is a broadcast domain. Each device in a VLAN, regardless of its physical location, can communicate directly with every other device in the same VLAN. However, they cannot communicate outside of the VLAN except through a router. A VLAN is usually created using physical ports.

Overview

The Zodiac FX is a 4 port network development board designed for hobbyists, students, researchers, embedded developers or anyone who requires a low cost network development platform. Even though it was initially designed to allow affordable access to OpenFlow enabled hardware it’s open source ﬁrmware it can be used in any number of other applications. By providing the ﬁrmware source code users are free to not only create their own versions but also use it as a basis for a completely different type of device. Some such applications may include:

• Router

• Bridge

• Load Balancer

• Web server

• VPN concentrator

• TOR client

• and many more.

Updating Firmware

The Zodiac FX ﬁrmware is stored within the ﬂash memory of the Atmel processor and is easily updated via the USB port. To update the ﬁrmware the existing code must ﬁrst be erased which will allow the CPU to boot in USB mode ready to receive the update. The ﬂashing procedure is as follows:

1. Download the required version of the ﬁrmware bin ﬁle which is available from the Zodiac FX forum (see appendix B). Alternately we can build we own version from the source code using Atmel Studio.

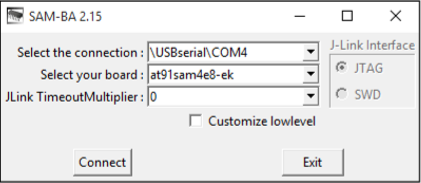
2. Disconnect the USB cable from the device.

3. While the device is powered off, CLOSE the ERASE jumper.

4. Reconnect the USB power source, wait 5 seconds and then disconnect it again. This will erase the ﬁrmware and reset the boot ﬂag to allow the ﬂash utility to communicate with the device.

5. Move the ERASE jumper back to the OPEN position. Be careful not to loose the jumper!

6. Connect the USB cable to power to the device again and open the SAM-BA utility (see list of links in appendix B).



7. Ensure the correct COM port is displayed and that the board type is “at91sam4e8-ek”, see ﬁgure below. Press the connect button. If the “Select the Connection” is blank then the existing ﬁrmware was not erased correctly, repeat steps 2-6.

8. In the “Download / Upload File” section click the folder icon for the “Send File Name” ﬁeld and select the appropriate bin ﬁle.

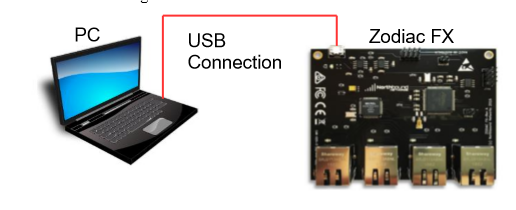
9. Click the “Send File” button to upload the ﬁrmware to the device.

10.When the “Lock region(s)” pop-up window appears select ”No”.

11. Under the “Scripts” section, select “Boot from Flash (GPNVM1)” from the drop down list and press the “execute” button.

12.Disconnect and reconnect the USB cable again to restart the device. The device will now load the updated ﬁrmware.

Connection:



Command Line Interface (CLI) :

The Command Line Interface provides the ability to conﬁgure setting and monitor the operation of the Zodiac FX. To simplify operations the CLI uses the concept of a “context”, this limits the available commands to only those available in the currently selected context. The are currently 3 available contexts:

* Base
* Conﬁg
* OpenFlow
* Debug

To enter the required context simply type the name of the context on the command line while at the base level. The return to the base level type “exit” The current context is shown in bracket between the device name and the prompt. for example Zodiac\_FX (conﬁg)#. The following sections describe the commands available within each context, please note that all commands are lowercase only.

Base:

* conﬁg - Enter the “conﬁg” context.
* openﬂow - Enter the “openﬂow” context.
* debug - Enter the “debug” context.
* show status - Displays the current device status.
* show ports - Displays information about each ethernet port including state, VLAN membership and trafﬁc statistics.
* show version - Display the ﬁrmware version.
* help - Display a list of available commands.

Conﬁg

* save - Saves the current conﬁguration to non-volatile memory.
* show conﬁg - Display the current device conﬁguration.
* show vlans - Displays a list of the currently conﬁgured VLANS.
* set name <name> - Sets the device name. Maximum of 16 characters, entries will be truncated.
* set mac-address <mac address> - Sets the MAC address of the device. The MAC address assigned to the device is located on a label on the underside of the device.
* set ip-address <ip address> - Sets the device IP address
* set netmask <netmask> - Set the device netmask
* set gateway <ip address> - Sets the default gateway of the device
* set of-controller <ip address> - Sets the IP address the OpenFlow controller
* set of-port <tcp port> - Sets the TCP port of the OpenFlow Controller
* set of-version <version> - Sets the device to only connect to an controller using the OpenFlow version speciﬁed. A value of 0 disables this function and allows the device to negotiate the version.
* add vlan <vlan id> <vlan name> - Creates a new vlan. Valid IDs are 1-4096 and names must be less then 16 characters.
* delete vlan <vlan id> - Deletes an existing vlan. set valn-type <type> - Set the vlan to either openﬂow or native.
* add vlan-port <vlan id> <port> - Assigns a ethernet port to the designated vlan. A port can only be a member of one vlan.
* delete vlan-port <port> - Remove the named Ethernet port from a vlan.
* factory reset - Conﬁgures and saves the conﬁguration back to the factory test conﬁguration, see appendix A for details.

Note: the MAC address is not reset and will remain at it’s current value and a restart is required to apply the changes. exit - Return the context back the base level.

OpenFlow

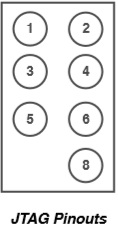
* show status - Displays the OpenFlow status.
* show ﬂows - Displays a list of the currently installed ﬂows.
* enable - Enables the Openﬂow functionality.
* disable - Disables the OpenFlow functionality. Disabling Openﬂow will clear the ﬂow tables
* exit - Return the context back the base level.

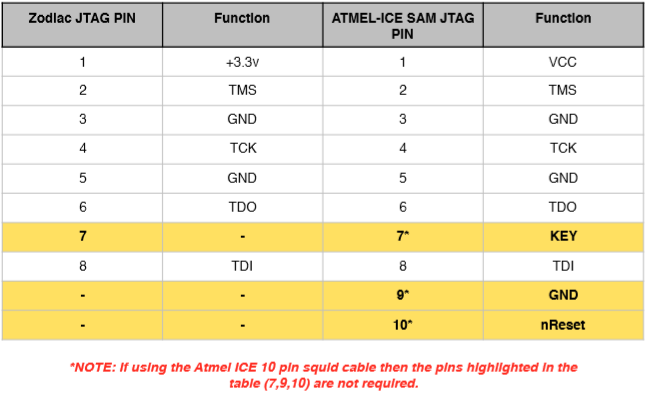
Debug

* read <register> - Display the value of the KSZ8795 register.
* write <register> <value> - Writes the value into the deﬁned KSZ8795 register.
* exit - Return the context back the base level.

JTAG Debugger

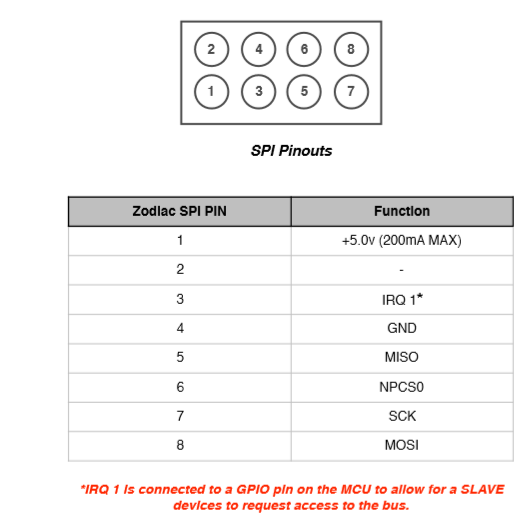
The following table describe the pin-outs of the Zodiac FX JTAG header and the ATMEL-ICE debugger SAM connector. If using the 10 pin squid cable that comes with the full ATMEL-ICE kit then the wires can be connected to the pins as shown below. Alternatively a Zodiac FX Developer kit can be purchased from the Northbound Networks store which includes the debugger and a speciﬁcally designed cable.





SPI Connector

The SPI connector allows the Zodiac FX to connect to other devices using the industry standard Serial Peripheral Interface (SPI). Information on the SPI standard can be found at https:// en.wikipedia.org/wiki/Serial\_Peripheral\_Interface\_Bus



Conclusion:

Zodiac Openflow Switch is first device of Software Define Networking. It’s very useful for researchers and students. Than develop a SND application using real traffic form hardware and discover power of Software Defined Networking. Even through there are numerous free or open source SND controllers.