Mawlana Bhashani Science and Technology University

Assignment

Assignment No:01

Assignment name: Zodiac OpenFlow Switch (Configure).

Course code: ICT-4101

Course title: Telecommunication Engineering

Date of Performance:10-10-2020

Date of Submission:19-10-2020

Submitted by

Name: Md. Rafatul Haque

ID:IT-17037

4th year 1st semester

Session: 2016-2017

Dept. of ICT

MBSTU.

Submitted To

Nazrul Islam

Assistant Professor

Dept. of ICT

MBSTU.

Assignment name: Zodiac OpenFlow Switch (Configure)

Objective:

- Configure and interact with Zodiac FX OpenFlow Switch.
- Exploring the Zodiac FX context.

Theory:

Zodiac FX Switch:

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.

Zodiac FX Description:

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 and many more. The main communication peripherals of Zodiac FX are sketched in Fig.1.

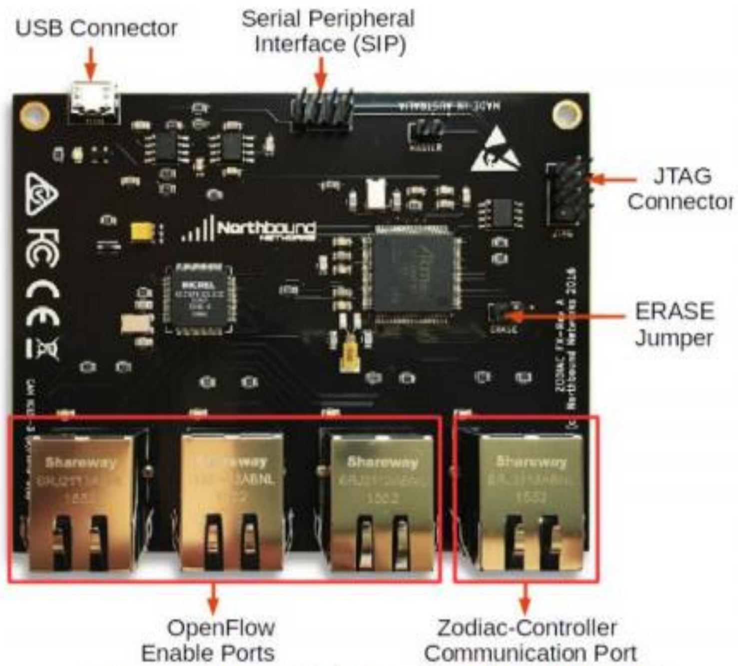

Figure 3-10. Communication peripherals of Zodiac FX.

The Zodiac FX provides many of the features of an OpenFlow switch costing thousands of dollars, yet is small enough to fit in the palm of your hand. Some of those amazing features include:

- 4 x 10/100 Fast Ethernet ports with integrated magnetics
- Command line interface accessible via USB virtual serial port
- Amtel ATSAM4E Cortex M4 processor
- Support for OpenFlow 1.0, 1.3 & 1.4
- 512 entry software flow table
- 64KB frame buffer with non-blocking store and forward
- 802.1q VLAN support for 64 groups from 4096 IDs
- Per port based 802.1x authentication
- 802.1w Rapid Spanning Tree Protocol (RSTP)
- 16 ACLs per port
- 2KB jumbo frame support
- QoS / CoS prioritisation with 802.1q tag insertion
- Auto MDIX with X-over detection
- Per port link and activity LEDs
- High speed SPI expansion header
- USB powered

- Ultra small size of only 10 cm x 8 cm

IP addressing:

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 you connect a new device, you 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 your home network—whether it's a standalone firewall or a router/gateway device or your 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, you'll want to avoid all of the addresses between 192.168.1.100 and 192.168.1.149 when you'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.

Methodology

Zodiac FX Command Line Interface (Z-CLI): The Zodiac CLI 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's, this limits the available commands to only those available in the currently selected context. There are currently four available contexts: Base, Config, OpenFlow and 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. The following sections describe the commands available within each context; please note that all commands are lower-case only.

1. **Base Functionalities:** The following commands are available in this context:

    - **config** Enter the config's context.
    - **openflow** Enter the OpenFlow's context.
    - **debug** Enter the debug's 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.

2. **Config Functionalities:** The following commands are available in this context:

- **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 than 16 characters.
- **delete vlan < vlan id >** Deletes an existing vlan.
- **set vlan-type < vlan id > < 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.
- **exit** Return the context back the base level.

3. **OpenFlow Functionalities:** The following commands are available in this context:

- **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.
- **clean flows** Disabling OpenFlow will clear the flow tables and
- **exit** Return the context back the base level.

4. **Debug Functionalities:** The following commands are available in this context:

- **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

Exercises

Exercise 4.1.4: Using the Base Functionalities command lines in Z-CLI, check the information about each Ethernet port. Which is the status of the port? Which ports are OpenFlow? Are all the ports in the same VLAN?

Zodiac_FX# show ports

Port 1

Status: DOWN

VLAN type: OpenFlow

VLAN ID: 100

Port 2

Status: DOWN

VLAN type: OpenFlow

VLAN ID: 100

Port 3

Status: DOWN

VLAN type: OpenFlow

VLAN ID: 100

Port 4

Status: DOWN

VLAN type: Native

VLAN ID: 200

Exercise 4.1.5: Using the OpeFLow Functionalities command lines in Z-CLI, check the information about flows? Is there any flow? If not, why?

Zodiac_FX(openflow)# show flows

No Flows installed!

There is no controller, therefore no flows are there.

Exercise 4.1.6: Using the Config Functionalities command lines in Z-CLI, check the Zodiac FX configuration, complete the following information:

Zodiac_FX(config)# show config

Configuration

Name: Zodiac_FX

MAC Address: 70:B3:D5:6C:D1:0B

IP Address: 10.0.1.99

Netmask: 255.255.255.0

Gateway: 10.0.1.1

OpenFlow Controller: 10.0.1.8

OpenFlow Port: 6633

Openflow Status: Enabled

Failstate: Secure

Force OpenFlow version: Disabled

Stacking Select: MASTER

Stacking Status: Unavailable

Is the MAC address equal to the one provided in the hardware? Yes

Display the status of the VLANS, in which VLAN the controller should be connected?

Zodiac_FX(config)# show vlans

VLAN ID   Name   Type

----

100   'Openflow'   OpenFlow

200   'Controller'   Native

Exercise 4.1.6: Connect the Ethernet cable between the Zodiac FX native port and the lab-PC as show in the following figure.

Show the status of the ports again using Z-CLI (same as Exercise 4.1.4), is the status of port 4 changed?

Zodiac_FX# show ports

Port 1

Status: DOWN

VLAN type: OpenFlow

VLAN ID: 100

Port 2

Status: DOWN

VLAN type: OpenFlow

VLAN ID: 100

Port 3

Status: DOWN

VLAN type: OpenFlow

VLAN ID: 100

Port 4

Status: UP

VLAN type: Native

VLAN ID: 200

In the lab PC open a terminal and ping the zodiac device (ping -c 5 10.0.1.99), does the ping works? No. If not, why? The Controller Ip need to be configured.

Exercise 4.1.7: Configuring lab PC to communicate with Zodiac FX. In order to establish the communication between Zodiac FX and Lab-PC using native port, Lab-PC has to be configured as Controller. Therefore, configure lab-PC with the following static IP address:

1. IP Address: 10.0.1.8

2. Netmask: 255.255.255.0

3. Gateway: 10.0.1.1

In the lab PC open a terminal, ping the zodiac device again (ping -c 5 10.0.1.99),

does the ping works? Yes

If yes, why? We configured the controller.

Section 4.2: Changing Zodiac configuration.

Exercise 4.2.1 Using the Config Functionalities command lines in Z-CLI, set Zodiac IP address to 10.0.1.(10+x), where x is the number of your Zodiac FX (check the box). Save the configuration in the non-voltaic memory.

Zodiac_FX(config)# set ip-address 10.0.1.8.11

IP Address set to 10.0.1.11

Zodiac_FX(config)# show config

Configuration

Name: Zodiac_FX

MAC Address: 70:B3:D5:6C:D1:0B

IP Address: 10.0.1.11

Netmask: 255.255.255.0

Gateway: 10.0.1.1

OpenFlow Controller: 10.0.1.8

OpenFlow Port: 6633

Openflow Status: Enabled

Failstate: Secure

Force OpenFlow version: Disabled

Stacking Select: MASTER

Stacking Status: Unavailable

Zodiac_FX(config)# save

Writing Configuration to EEPROM (200 bytes)

Done!

Exercise 4.2.2: In the lab PC open a terminal, ping the zodiac device again using the new IP address, does the ping works? Yes.

Exercise 4.2.1: Using the Config Functionalities command lines in Z-CLI, create a new vlan the name SDN_lab and type openflow (Vlan-ID=300). Then add the port 3 to the new vlan. Save the configuration in the non-voltaic memory.

Zodiac_FX(config)# add vlan 300 SDN_lab

Added VLAN 300 'SDN_Lab'

Zodiac_FX(config)# set vlan-type 300 openflow

VLAN 300 set as OpenFlow

Zodiac_FX(config)# delete vlan-port 100 3

Port 3 has been removed from VLAN 100

Zodiac_FX(config)# add vlan-port 300 3

Port 3 is now assigned to VLAN 300

When done display show vlans and show ports and ask your tutor for marking it.

Zodiac_FX(config)# show vlans

VLAN ID	Name	Type
100	'Openflow'	OpenFlow
200	'Controller'	Native
300	'SDN_Lab'	OpenFlow

Zodiac_FX# show vlan ports

Port 1

Status: DOWN

VLAN type: OpenFlow

VLAN ID: 100

Port 2

Status: DOWN

VLAN type: OpenFlow

VLAN ID: 100

Port 3

Status: DOWN

VLAN type: OpenFlow

VLAN ID: 300

Port 4

Status: UP

VLAN type: Native

VLAN ID: 200

Conclusion: In this assignment, we developed SDN Testbed and surveyed the design of OpenFlow and SDN Ryu Controller. It is observed a real programmable network by setting up a Ryu SDN controller and Zodiac FX for the SDN scenario. We presented all possible configurations that show how to build a real SDN scenario via a hardware switch Zodiac FX. Hence, it can say that SDN is more flexible than traditional network and main advantage of SDN is a cost efficient and programmable. Zodiac FX can be used to perform more experiments and examine the performance via Rya OpenFlow controller and other SDN controllers. It is observed that the major limitation of SDN controllers that if any active controller fails it can rapidly break down the entire network.