Name: Md.Sohag Hossain

ID : IT-17060

Assignment no: 01

Name of the assignment: Zodiac OpenFlow Switch (Configure)

Objectives:

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

Theory:

Introducing the Zodiac FX

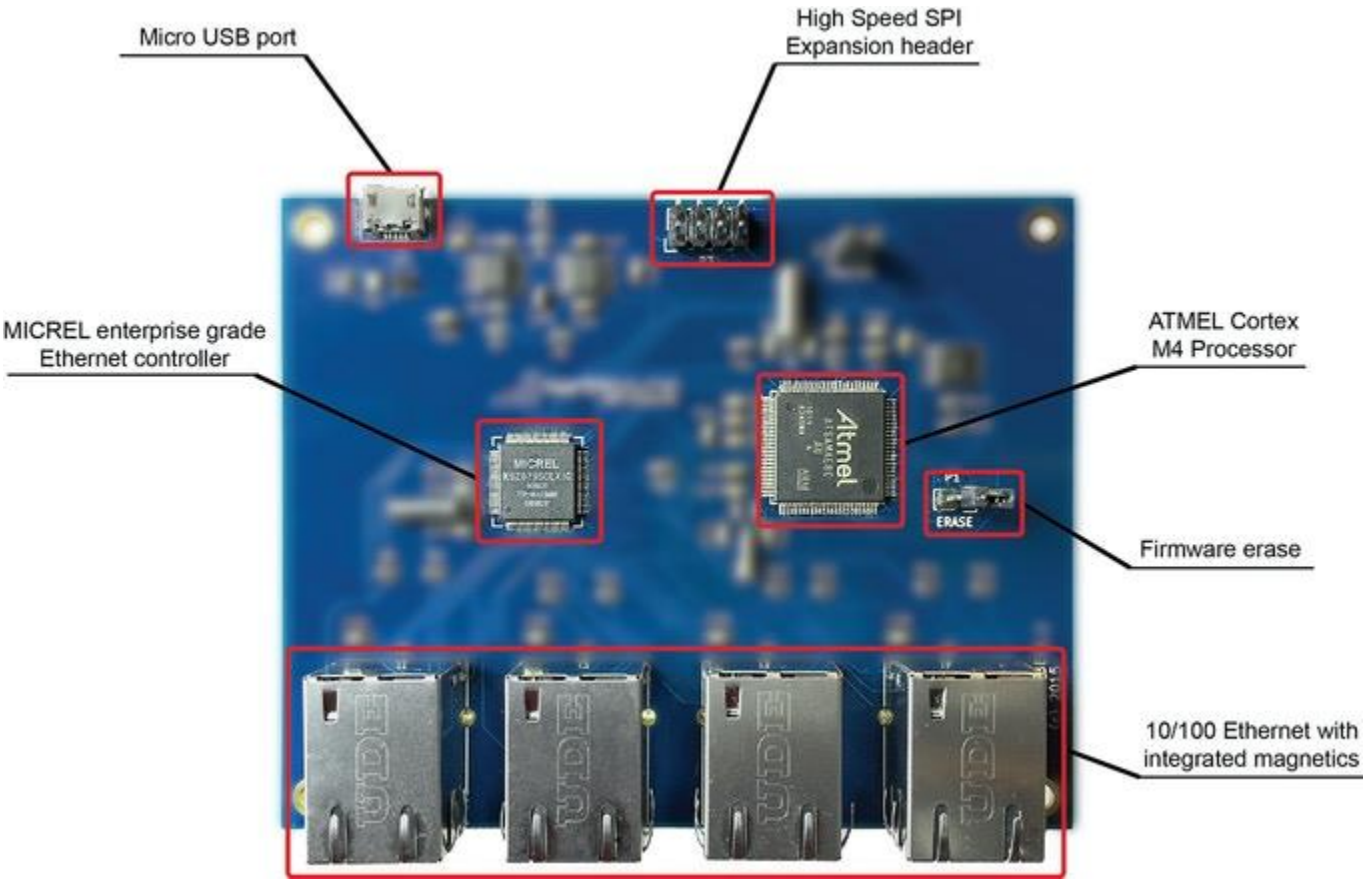
The Zodiac FX from Northbound Networks is an OpenFlow switch designed for affordability. With this switch, the power of Software Defined Networking is available to researchers, students, and anyone who wants to develop their SDN skills or build applications using real hardware.

Features 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:

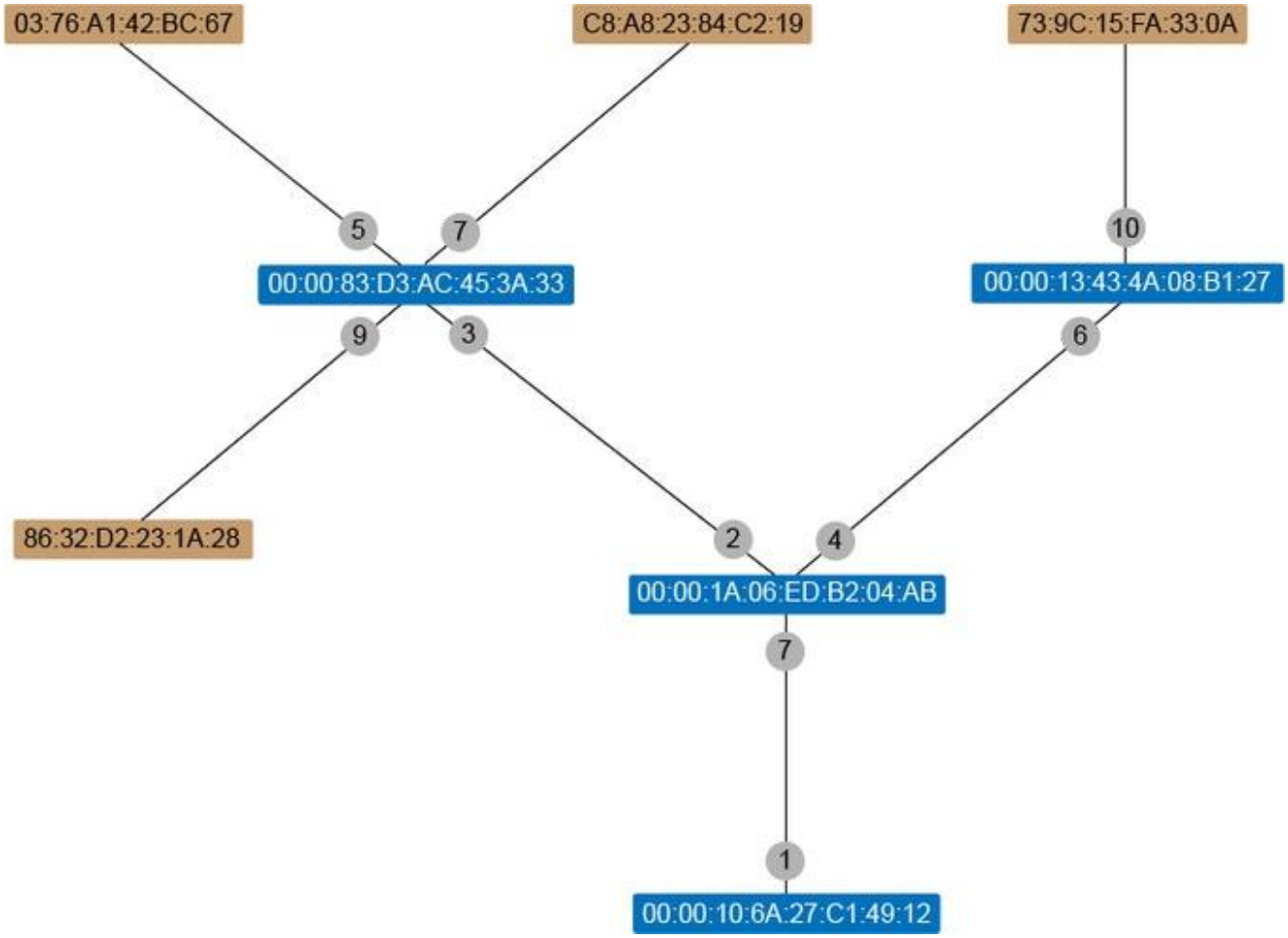
- 4 x 10/100 Fast Ethernet ports with integrated magnetics
- Command line interface accessible via USB virtual serial port
- Amtel ATSAM4E Cortex M4 processor
- Support for OpenFlow 1.0, 1.3 & 1.4
- 512 entry software flow table
- 64KB frame buffer with non-blocking store and forward
- 802.1q VLAN support 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



What is OpenFlow?

OpenFlow is the protocol that makes Software Defined Networking possible. OpenFlow was originally developed by researchers at Stanford University and is now supported by all major networking vendors. At the centre of an OpenFlow network in an SDN controller, this is where your applications are installed. OpenFlow is used to communicate to the switches on your network which then process the data going through their ports using what are called "flows". These flows are generated by your application based on the task you have programmed it to perform.



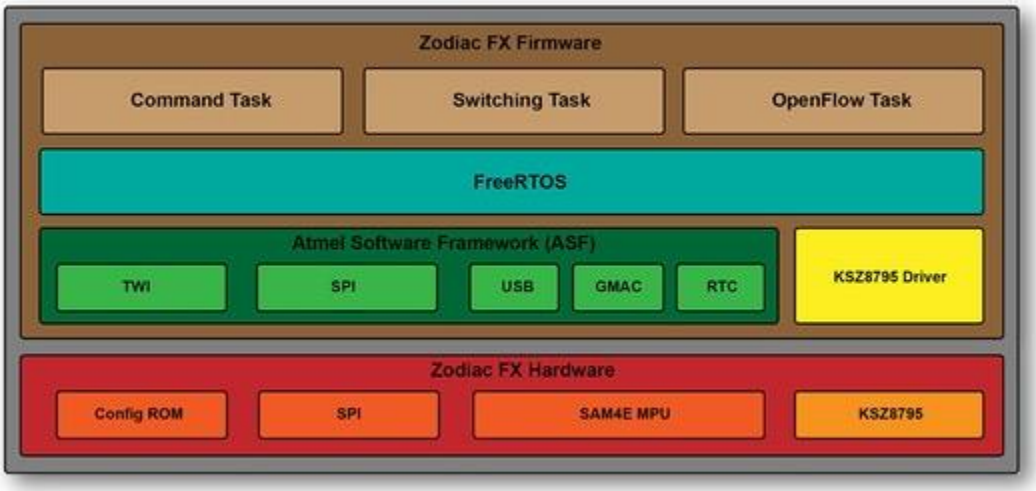
Open Source firmware

Because we are part of the SDN development community, just like you we believe it's important to share the things we create. That is why the Zodiac FX firmware is completely open source. This means that anyone can download the source from our website and use the free Atmel Studio to produce their own custom version. Even though we will continue to develop

the firmware ourselves, it doesn't stop someone from forking a different version and building an entirely new community around it.

```
/** Load MAC address */
static uint8_t gs_uc_mac_address[] =
{ ETHERNET_CONF_ETHADDR0, ETHERNET_CONF_ETHADDR1, ETHERNET_CONF_ETHADDR2, ETHERNET_CONF_ETHADDR3, ETHERNET_CONF_ETHADDR4, ETHERNET_CONF_ETHADDR5
};

/** Load IP Address */
static uint8_t gs_uc_ip_address[] =
{ ETHERNET_CONF_IPADDR0, ETHERNET_CONF_IPADDR1, ETHERNET_CONF_IPADDR2, ETHERNET_CONF_IPADDR3 };

/** The GMAC driver instance */
extern gmac_device_t gs_gmac_dev;

/** Buffer for ethernet packets */
static volatile uint8_t gs_uc_eth_buffer[GMAC_FRAME_LENTGH_MAX];

#define TASK_OPENFLOW_STACK_SIZE        ( configMINIMAL_STACK_SIZE + 30 )
#define TASK_OPENFLOW_PRIORITY          1 //(tskIDLE_PRIORITY)

//------------------------------------------------------------------------------
struct tcp_conn tcp_conn;
struct ethernet_header eth_hdr;
struct _IPheader ip_hdr;
struct _TCPheader tcp_hdr;
//------------------------------------------------------------------------------
```

The Zodiac FX firmware utilises the Atmel Software Framework (ASF) for generic device drivers such as USB, SPI, etc. On top of this it then adds a custom written driver for the KSZ8795. FreeRTOS is used to provide task and memory management for the three core processes; Command (CLI), Switching and OpenFlow.



IP addressing:

Static IP Addressing

Application owners feel challenged when they move their multi-VM, multi-tier environment from data-center to AWS & Google cloud. The networking changes –

Dynamic Host Configuration Protocol (DHCP):

you cannot keep the same IP addresses, netmask, networking interconnect on public cloud as you have in your data center. Ravello's software defined networking (SDN) overlay gives you the ability to mirror your data-center network on AWS & Google cloud without making changes. This article goes over how to mirror the static IPs that you have in your data center on top of Ravello's platform to be able to run your application on AWS & Google cloud.

Dynamic Host Configuration Protocol (DHCP) is a client/server protocol that automatically provides an Internet Protocol (IP) host with its IP address and other related configuration information such as the subnet mask and default gateway. RFCs 2131 and 2132 define DHCP as an Internet Engineering Task Force (IETF) standard based on Bootstrap Protocol (BOOTP), a protocol with which DHCP shares many implementation details. DHCP allows hosts to obtain required TCP/IP configuration information from a DHCP server.

Windows Server 2016 includes DHCP Server, which is an optional networking server role that you can deploy on your network to lease IP addresses and other information to DHCP clients. All Windows-based client operating systems include the DHCP client as part of TCP/IP, and DHCP client is enabled by default.

Virtual Local Area Network (VLAN):

In a traditional LAN, workstations are connected to each other by means of a hub or a repeater. These devices propagate any incoming data throughout the network. However, if two people attempt to send information at the same time, a collision will occur and all the transmitted data will be lost. Once the collision has occurred, it will continue to be propagated throughout the network by hubs and repeaters. The original information will therefore need to be resent after waiting for the collision to be resolved, thereby incurring a significant wastage of time and resources. To prevent collisions from traveling through all the workstations in the network, a bridge or a switch can be used. These devices will not forward collisions, but will allow broadcasts (to every user in the network) and multicasts (to a pre-specified group of users) to pass through. A router may be used to prevent broadcasts and multicasts from traveling through the network.

.***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 Page | 45 SDN-Labs 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

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.

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. Page | 46 SDN-Labs

• 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

Explain the difference between the Native and OpenFlow ports?

By default, two VLANs are configured on the Zodiac FX:

- "OpenFlow", with ID 100
- "Native", with ID 200

"OpenFlow" Port:Intended for general network traffic. Connect these ports to network hosts.

"Native" Port: intended for management traffic. Connect this port to the OpenFlow controller.

By default, ports 1-3 are set as OpenFlow ports. Port 4 is set to Native.

Why we cannot use dynamic IP between zodiac and lab-PC?

Because zodiac and controller IP address are configured in the code as static address.

What is the difference between and OpenFLow and non-OpenFow switch (apart from the support of openflow)

The architecture of the network is different; an OpenFlow switch works only when a controller is active.

Provided others examples of commercial OpenFlow switches.

- ✓ HP,
- ✓ Huawei any open-flow switch is valid.

Using the USB-MiniUSB cable connect to Zodiac FX.

We don't have any zodiac FX for this reason we can't connect to the zodiac with the computer using a USB-MiniUSB cable.

In a terminal check if the device is detected using the following the command lines a. Run cd /dev
b. Run ls and look for the ttyACMx (where x represent the number that your Linux OS give to the device.)

we don't have any zodiac FX.

Open Putty Software: run the command sudo putty and configure it using the following procedure:

Configure the communication as show in the figure:

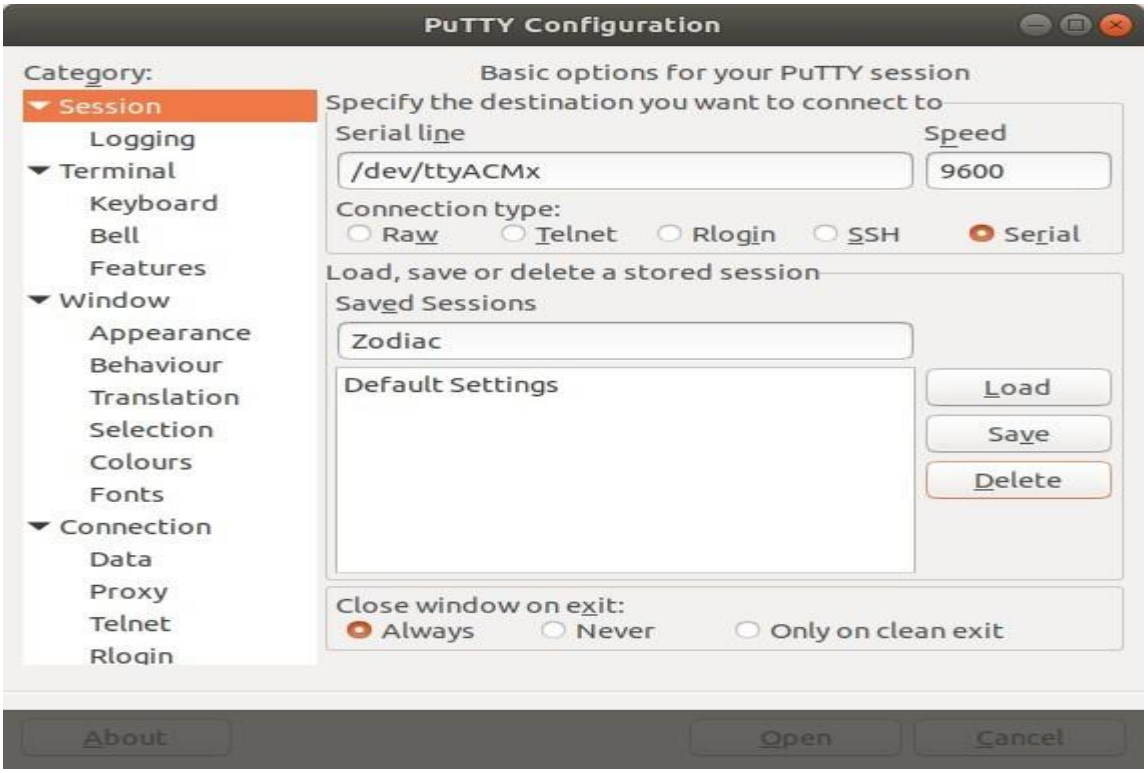
Serial Line: /dev/ttyACMx

Tick serial

Provide a name example Zodiac

Save the

session



The following error will occur because we don't have any zodiac FX connected to my PC.

Conclusion:

I have confronted a lot of problems when I doing the assignment. First I have faced that is I need a zodiac FX but I don't have any zodiac FX to connect with pc

and can not able to configure putty figure. At last I have overcome all this problem.