Data Communication Networks, HW#2

Instructor: Dr. MohammadReza Pakravan

Data Link Layer

Beginning Notes: The task of the data link layer is to convert the raw bit stream offered by physical layer into a stream of frames for use by the network layer. The link lay can present this stream with varying levels of reliability, ranging from connectionless, unacknowledged service to reliable, connection-oriented service.

In this homework, we are going to get familiar with Ethernet Data Link Layer of the standard 802.3. To this end, you will answer to some theoretical questions, investigate the Ethernet interface of your laptop, and explore the Ethernet frames by means of Wireshark. Therefore, you need an up and running Unix-based operating system and Wireshark on your computer.

Ethernet Network One of your classmates runs a 10Mbit/s Ethernet between the campus and the homes of his friends. All of these hosts are in a single broadcast domain. Fortunately, they are located just within the maximum distance of an Ethernet. The total size of an Ethernet can be about 2.5 kilometers, and an Ethernet has a minimum packet size of 512 bits. he upgrades his network to 100Mbit/s Ethernet, and notices that when only one person sends at a time, or when he sends very large packets, his network works. But when many people send very small packets, things dont work at all.

---

1. Explain how a minimum packet size can help to detect collisions in Ethernet.
2. Help him out. Compute the threshold of how big packets must be in order for things to work.
3. One solution that he came up with is to raise the minimum packet size to the answer from part (2). Suppose that Frank cannot modify the minimum packet size, move the endpoints, lay new cable, or change the software or configuration on the endpoints. You may add new devices to the network. How could Frank change the topology to fix his problems anyway?

Bit/Byte Stuffing Various framing methods are used in Data Link Layer, including byte stuffing, and bit stuffing.

---

1. Explain how bit stuffing and byte stuffing works, and name a protocol which uses each of these methods.
2. A bit string, 0111101111101111110, needs to be transmitted at the data link layer. What is the string actually transmitted after bit stuffing?
3. What is the maximum overhead in byte-stuffing algorithm?

Ethernet Protocol Answer the following questions in your report regrading Ethernet protocol.

---

1. Why did Ethernet win out over Token-Ring?

2. Why is the minimum size of an Ethernet frame 64 bytes?
3. Why is the maximum size of an Ethernet frame 1500 bytes?
4. What does your Ethernet card do to a frame if it calculates an invalid CRC?
5. Explain the purpose of MAC address and OUI.

Cyclic Redundancy Check CRC is an Error-correcting code widely used on wireless links, which are notoriously noisy and error prone. Write a MATLAB code to calculate the CRC for a given data string. You have to implement the serial Hardware implementation of CRC-32 depicted in Figure.1 of reference [1]. Note that the implementation by polynomial will not be accepted.

---

1. Explain the purpose of CRC and its basics.
2. What are the most used types of CRC and in what protocols are they used?
3. Assume this is an Ethernet frame, calculate the CRC and check it with the CRC field of the frame. (Make sure to omit the fields of the frame that are not considered in CRC calculation beforehand):
00 10 A4 7B EA 80 00 12 34 56 78 90 08 00 45 00 00 2E B3 FE 00 00 80 11 05 40 C0 A8 00 2C C0 A8 00 04 04 00 04 00 00 1A 2D E8 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 B3 31 88 1B

Investigating an Ethernet Interface Find a Linux machine to do this on, ideally with a wired Ethernet port. For best results the port should be connected to a network via a cable. (If wired Ethernet interface is accessible, you can investigate wireless-LAN).

---

1. Determine the current speed being used by your Ethernet adapter. To this end, run one of the following commands in terminal:

```
sudo dmesg | grep eth
sudo ethtool eth0
cat /sys/class/net/eth0/speed
```

2. Look at the information on your Ethernet device. Use the following command

```
ifconfig
```

   - What is your MAC address?
   - Look this up using an OUI lookup table. Does it match what you expect?
   - What is the default frame size (MTU)?
   - How many bytes have been received (RX)?
   - How many bytes have been transmitted (TX)?
   - Has your device seen any collisions?
   - Has your device dropped any packets?

3. If the collision count is low, can you explain why that is?

Exploring Ethernet Frames In order to capture and examine Ethernet frames, we use Wireshark. Wireshark is a packet trace tool which records traffic at a location on the network, as if a snapshot was taken of all the bits that passed across a particular wire. The packet trace records a timestamp for each packet, along with the bits that make up the packet, from the lower-layer headers to the higher-layer contents. Launch Wireshark and choose Ethernet interface (eth0 in Ubuntu). Open a website in your browser and capture the traffic by Wireshark. After some time, pause the capturing process. Choose a random packet and answer the following questions.

---

1. Save the selected packet from File>Export Specified Packets and choose Selected packets and upload it with your report.
2. Based on the Ethernet frame structure, what are the source and destination MAC addresses and Who owns the OUI of them.
3. What is the value of the Ethernet type field?
4. What is the purpose of an ARP packet? Select one the ARP packets, what is its destination address and why?
5. There is no CRC field in the frames you captured, search and explain the reason.

---

1 What SHOULD I prepare?

You must upload a report in Portable Document Format (.pdf) for this assignment, the MATLAB file containing your CRC procedure, and the .pcap file of your packet. The document should contain answer to each question, any necessary comments and screen-shot of what you have done in each part.
After this homework you must be familiar with elementary Ethernet data link layer and their characteristics. Let us know about any problems in questions or understanding the concepts.

References

[1] Chris Borrelli, *IEEE 802.3 Cyclic Redundancy Check*.
https://www.xilinx.com/support/documentation/application_notes/xapp209.pdf.