

BINARY EXPONENTIAL BACKOFF EXERCISE LAN1

PURPOSE

This purpose of this exercise is for you to demonstrate your understanding of how Ethernet implements a collision response algorithm.

EXERCISE PROCESS

JNWS must be installed as described in the User's Guide.

You will need to run the simulation to demonstrate that works.

You are provided with a template for your program in class ModelDataLink.EthernetUtilities.ComputeBackoff.

1. You will need a random number generator. To ensure repeatability, JNWS has only one random number generator. You can instantiate that random number generator as

```
Utility.JNWRandom randomNumberGenerator  
= Utility.JNWRandom.getInstance ( ) ;
```

This will give you access to all the methods of java.util.Random plus a Poisson random number generator. java.util.Random is described in <http://java.sun.com/j2se/1.5.0/docs/api/java/util/Random.html>. JNWRandom also has a JavaDoc. Each time you run the model you should get identical results. You can change the random number seed at the General Variables pull-down on the MainGUI. The possible seeds are -9223372036854775808 to 9223372036854775807.

2. Complete the method ModelDataLink.EthernetUtilities.ComputeBackoff () that return the time delays in ticks between successive transmission attempts. Since the delays are random within a range that increases with attempts, this is where you'll need the RNG. The return value should be a multiple of 512 ticks (which is 51.2 microseconds in the simulation). Use the given local value "slot_ticks" for 512.

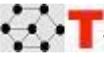
3. The description the Binary Exponential Backoff you are programming is given in Understanding Internet Protocols, Chapter 6. Input to the method is the parameter transmissionAttemptNumber. Use this parameter to determine the range of backoff values for the particular attempt. Since backoff will be done by

different interfaces at random, you cannot assume that two backoff attempts are for subsequent transmissions on the same interface; this is why each interface has its own instance of the RNG.

4. Terminology: The maximum number of retransmission attempts for Ethernet is 16. This is `EthernetState.maximumNumberOfTransmissionAttempts`. The `ComputeBackoff()` template comments mention `max_backoff_slots`. This value represents the range of possible backoff values that the function can return at any given point. For the first transmission attempt, the possible backoff slot counts are 0 and 1. `max_backoff_slots` is therefore 2 (the number of values) not 1 (the maximum value). Where the template comments mention `backoff_count`, this is the same as `transmissionAttemptNumber`. Also, don't forget that, when converting from float to int, Java (like other programming languages) truncates the floating-point value; it does not round.

EXERCISE

Complete the method `ComputeBackoff()`. Start JNWS. If the startup main class is not set to `StartJNW`, change it to that class as detailed in the JNW User's Guide. Then run JNWS following these steps. The buttons in described are at the top of the GUI, immediately under the File etc. links. They also can be accessed via the Enter link (between File and Action).

1. Use the "Open File Containing Topology" button  to read `EthernetBus.txt`.
2. Use the "Open File Containing E-Mail" button  to read the emails in `EMAIL_UDP_2.txt`.
3. Click on the run button . Both of the nodes should try to transmit the message at the same time, causing collisions.
4. Try repeating the exercise with a different random number seed. A seed can be entered on the MainGUI on the Enter Global Variables button (leftmost on the GUI). You should get slightly different answers-remember this is a randomized process.

NOTE: Statements to print the collision output required by the assignment are already in the JNWS, in class `SendEthernetFrameFromInterface()`; so the printout will happen automatically when you run the simulation.

`transmissionAttemptNumber` is not there; you can print it out yourself but that is not required. If you do print it, be sure to use `Utility.out.println()`.

DEMONSTRATION

No demonstration is provided for this exercise.

SUBMITTING

Run the simulation, displaying the appropriate data to the MainGui. (The output statements for collisions, including transmissionAttemptNumber and ethernetEventTime, are already provided in JNWS.) Copy the printed output into a text file. Submit the text file along with your EthernetUtilities.java to the grader, via Moodle.