

## Project No. 2: Data Link Layer Protocols

### PURPOSE

- Exercising the flow control and error control techniques in the data link layer protocols.

### DESCRIPTION

In this project, you will implement three data link layer protocols, *Stop and Wait + Timeout + Sequence Number (Positive Acknowledgment with Retransmission)*, *Go Back n Sliding Window* and *Selective Repeat Sliding Window*. You will also implement a routine to compute (and verify) the CRC checksum of a frame. In this project, we will use CRC-16,  $x^{16} + x^{15} + x^2 + 1$ , as the generator polynomial. To simplify the implementation, you only need to calculate the checksum for the packet data (without considering the frame header information). Your checksum routine should be able to deal with packets of different sizes. The prototype for the checksum routine should be:

```
int checksum(char *data, int length);
```

You will build the system on top of a physical layer emulator, which you can download from

<http://www.cs.fsu.edu/~xyuan/cen5515>.

The emulator simulates an unreliable duplex physical channel. Your task is to create reliable, efficient, connection oriented services on top of the unreliable physical channel.

### GRADING POLICY

To verify your program (checksum function and the protocols), your PAR and Go back N sliding window senders should be able to work with the sample receiver. Only minor partial credits will be given for incomplete efforts.

- Program compiles with no error (20)
- Correct checksum routine. (20)
- PAR (10)
- *Go back N* ( size of the sender window  $\geq 2$ ) sliding window protocol. (10)
- *Selective repeat* (size of both windows  $\geq 2$ ) sliding window protocol. (10)
- Report. (10)
- Demo. (20)

### DEADLINES AND MATERIALS TO BE HANDED IN

You will demonstrate the project on **Nov. 1**. The project report will be handed in at the demo time. The report must include a hard copy of your programs (sender and receiver), a detailed description of the implementation of the sliding window protocols, the performance comparison of the protocols, and an explanation of the performance results. Both hard and soft copies of the project should be handed in at demo time.

### MISCELLANEOUS

The cheating policy will be strictly enforced. All programs handed in will be checked by MOSS, a software plagiarism detection system.