CS 4457 - Networks Lecture 3 Notes

Nate Hunter, Andrea Zhang

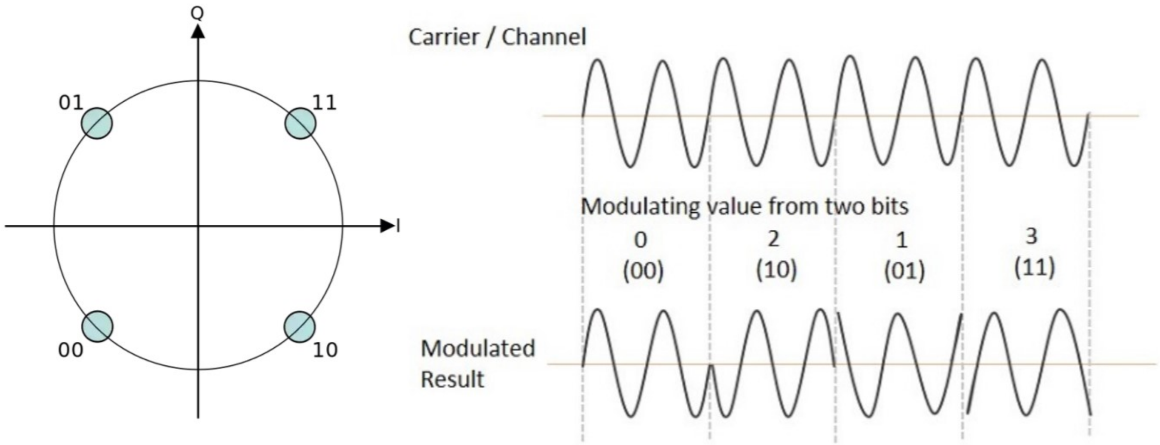
January 21, 2020

Part I
Practice Questions

1 Multiple Signals

Can you send multiple signals on the same channel? Sending multiple signals on the same channel results in disaster: rather than receiving two readable signals, both signals become garbled. Solutions to this include sending the two signals on separate carrier frequencies or dividing up one carrier frequency into alternating time slots for each signal.

2 Packet/IQ Conversion

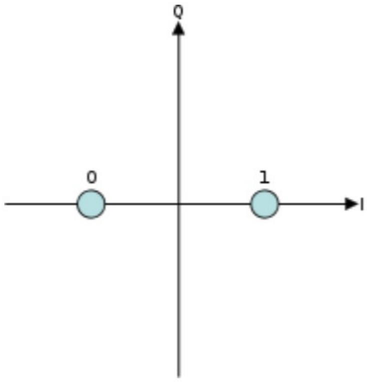


Examine the modulation scheme (left) and modulated signal (right) as shown above. What packet is being sent? The answer, of course, is 00100111 (00 10 01 11). Every two cycles, the phase of the wave specifies a two-bit code, and those two-bit codes are strung together as a packet.

What series of IQ samples would generate this signal? For this question, remember that the four phases of the wave correspond to four coordinate positions in the IQ-plane. Thus, the answer is:
[(-1, -1) , (-1, 1) , (1, -1) , (1, 1)].

3 Baud Rate

Looking back at the image of the modulated signal, suppose that each unit of time (two sinusoid periods) is 1 millisecond. What is the date rate (or baud rate)? Well, two bits are sent in each millisecond unit of time, so the data rate is 2 bits per millisecond or 2000 bits per second.



Now suppose that BPSK (binary phase shift keying) is the modulation scheme, as shown above. What is the new data rate? Before, we employed QPSK (quadrature phase shift keying), which transmitted 2 bits for each unit of time. Now, with BPSK, we only transmit 1 bit for each millisecond unit of time. Thus, the data rate is 1000 bits per second (1 kbit/s).

Part II

Chapter 5: Link Layer

4 Introduction, Services

4.1 Terminology

Node: Any device that runs a link-layer protocol, like hosts and routers.

Link: Communication channels that connect adjacent nodes along communication path, like wired links, wireless links and LANs.

Frame: A frame consists of a data field, in which the network-layer datagram is inserted, and a number of header fields. The structure of the frame is specified by the link-layer protocol. It's a layer-2 packet.

Data-link layer A layer that has responsibility of transferring datagram from one node to physically adjacent node over a link.

4.2 Context

Table 1 summarizes how the link layer is similar to real life transportation.

4.3 Services

Basic Services: To move a datagram from one node to an adjacent node over a single communication link

Possible Services:

Link Layer	Transportation Analogy
datagram transferred by different link protocols over different links: e.g., – Ethernet on first link, frame relay on intermediate links – 802.11 on last link	trip from C-Ville to New York – limo: C-ville to CHO – jet: CHO to LGA – train: LGA to Mahanttan
each link protocol provides different services. e.g., may or may not provide rdt over link*rdt: reliable data transfer	tourist = datagram transport segment = communication link transportation mode = link layer protocol travel agent = routing algorithm

Table 1: Link Layer Context Analogy

- **framing** Almost all link-layer protocols encapsulate each network-layer datagram within a link-layer frame before transmission over the link. Encapsulate datagram into frame, adding header, trailer.

- **link access** A medium access control (MAC) protocol specifies the rules by which a frame is transmitted onto the link. For point-to-point links that have a single sender at one end of the link and a single receiver at the other end of the link, the MAC protocol is simple (or nonexistent)—the sender can send a frame whenever the link is idle. "MAC" addresses used in frame headers to identify source, dest. They're different from IP address. They only exist in the link layer.

- **reliable delivery between adjacent nodes** A link-layer reliable delivery service is often used for links that are prone to high error rates, such as a wireless link, with the goal of correcting an error locally—on the link where the error occurs—rather than forcing an end-toend retransmission of the data by a transport- or application-layer protocol. Note it's seldom used on low bit-error link (fiber, some twisted pair).

- **flow control** Pacing between adjacent sending and receiving nodes since it's impossible to process data simultaneously.

- **error detection** Bit errors are introduced by signal attenuation, noise. Receiver detects presence of errors: signals sender for retransmission or drops frame.

- **error correction** Receiver identifies and corrects bit error(s) without resorting to retransmission.

- **half-duplex and full-duplex** With half duplex, nodes at both ends of link can transmit, but not at same time

4.4 Where is the link layer implemented?

- The link layer is in each and every host.

- **NETWORK ADAPTER** The link layer is implemented in a **network adapter**, also sometimes known as a **network interface card (NIC)**. At the heart of the network adapter is the link-layer controller, usually a single, special-purpose chip that implements many of the link-layer services. Thus, much of a link-layer controller's functionality is implemented in hardware. 4.4 shows a network adapter attaching to a host's bus. The link layer is a combination of hardware and software. It has a MAC address of your machine associates with it.

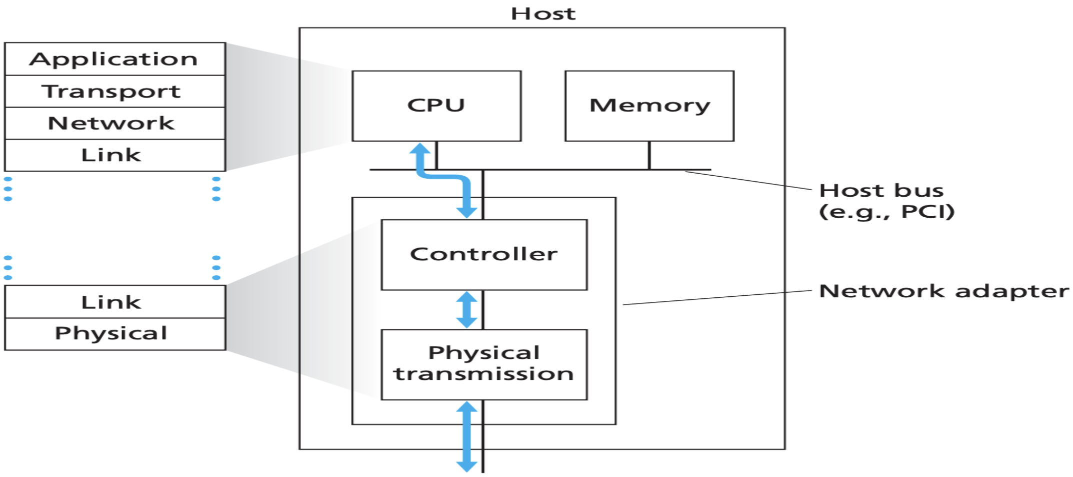


Figure 1: Network adapter: its relationship to other host components and to protocol stack functionality

- **Adaptors communicating** As 4.4 shows, the sending side encapsulates datagram in frame and adds error checking bits, rdt and flow control, etc. The receiving side looks for errors, rdt, flow control, etc, and extracts datagram, passes to upper layer at receiving side.

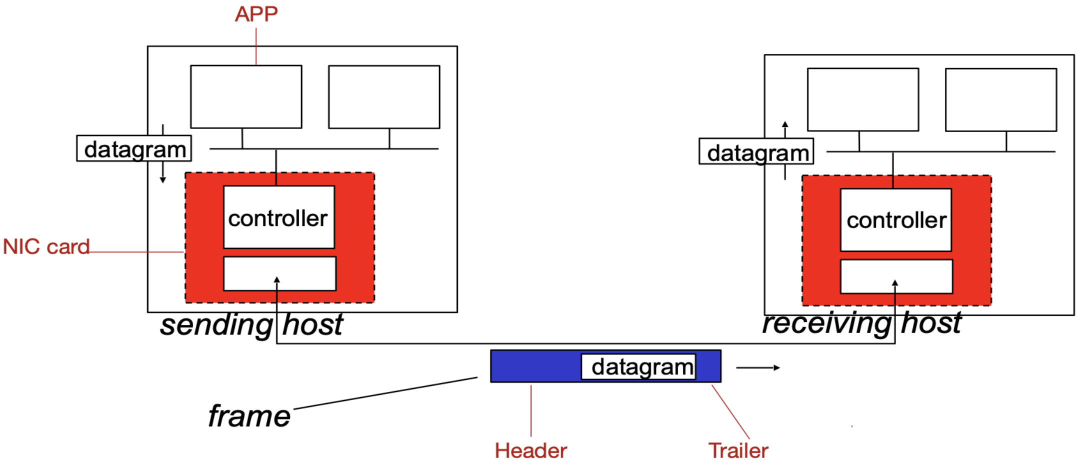


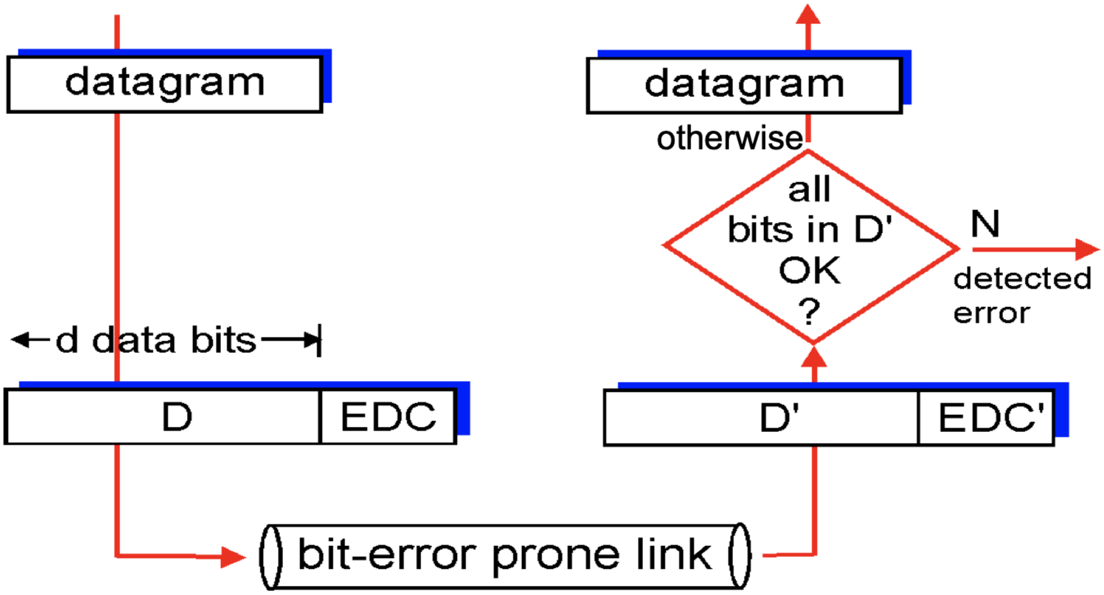
Figure 2: Adaptors communicating

5 Error detection, Correction

5.1 Error detection

- Terminology

1. EDC= Error Detection and Correction bits (redundancy)
2. D = Data protected by error checking, may include header fields



heightheight

Figure 3: Error Detection

- As we can tell in 5.1, with EDC, as a consequence of getting correct information, we receive less data each time.
- Error detection not 100% reliable since we're sending the EDC through the unreliable link as well. However, a larger EDC field yields better detection and correction.

5.2 Parity Checking

5.2.1 Single Parity Check

Suppose that the information to be sent, D, has d bits. In an even parity scheme, the sender simply includes one additional bit and chooses its value such that the total number of 1s in the d + 1 bits (the original information plus a parity bit) is even. For odd parity schemes, the parity bit value is chosen such that there is an odd number of 1s. Same for even/odd parity with 0s. 5.2.2 is an example of data with a single parity bit.

Single Parity Check can detect single bit errors.

5.2.2 Two-dimensional bit parity

The d bits in D are divided into i rows and j columns. A parity value is computed for each row and for each column. The resulting i + j + 1 parity bits comprise the link-layer frame's error-detection bits.

Suppose now that a single bit error occurs in the original d bits of information. With this two-dimensional parity scheme, the parity of both the column and the row containing the flipped bit will be in error. The receiver can thus not only detect but identify the bit that was corrupted and correct that error. 5.2.2 is a comparison between the original data with two-dimensional bit parity and the corrupted data.

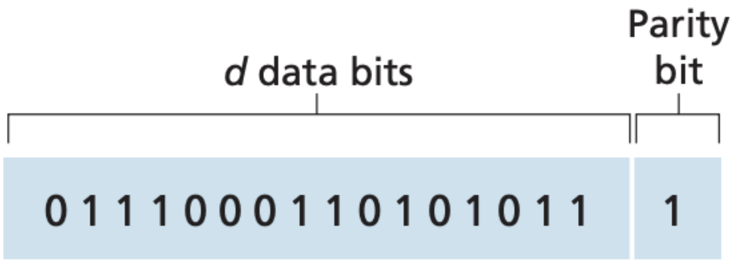


Figure 4: Single Parity Check(uncorrupted)

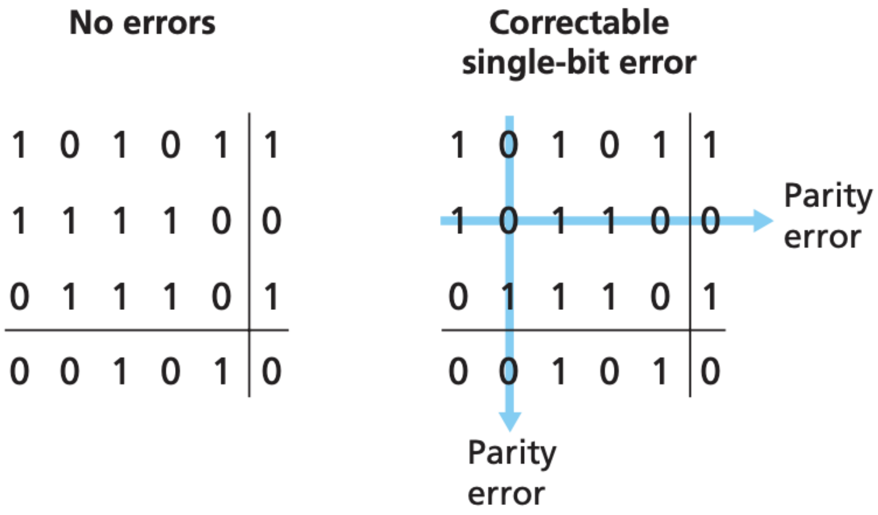
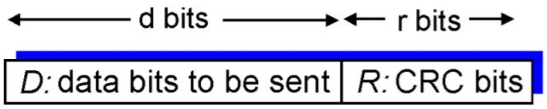


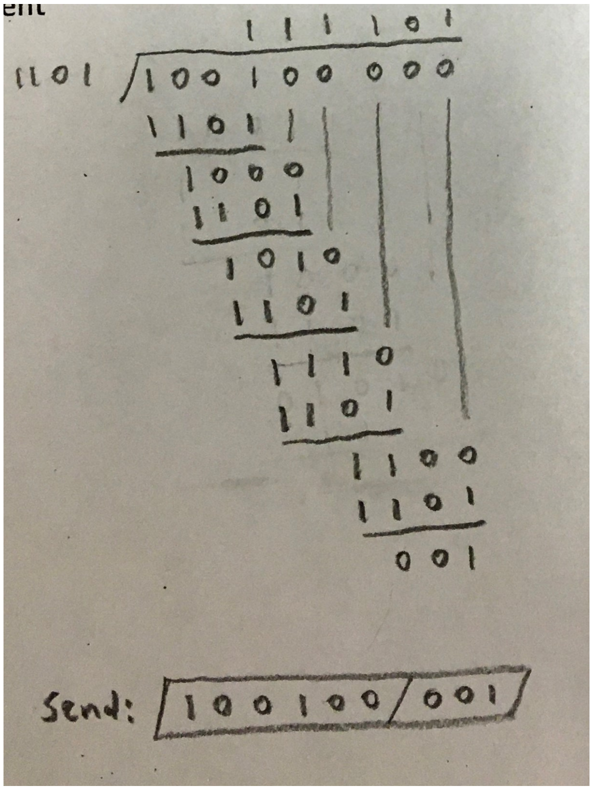
Figure 5: Two-dimensional bit parity

6 Cyclic Redundancy Check

Cyclic redundancy check (CRC) is a form of error detection (not correction) which can be more powerful than a simple parity check. Consequently, it is used by many network protocols, including Ethernet and WiFi. It involves sending a remainder along with the data that is the result of dividing the data (D, a binary number) by a generator number (G), using modulo-2 arithmetic. If the CRC code (R) does not match the expected remainder, then there is an error.



As an example, suppose that the datagram (D) we wish to send is 100100, and we use a generator (G) of 1101. What CRC code should we send at the end of the packet?



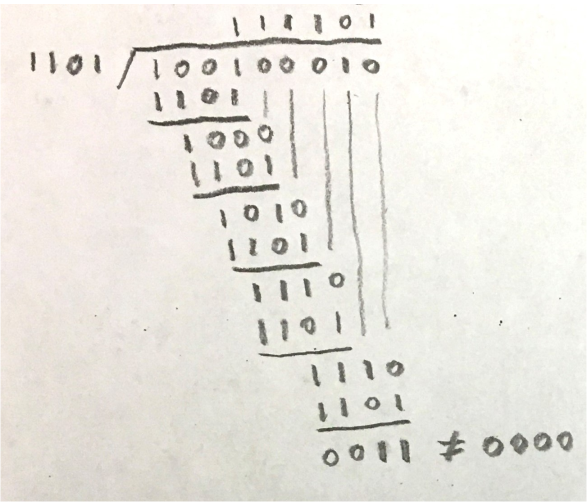
ent

```
              1 1 1 1 0 1
1101 / 1 0 0 1 0 0 0 0 0
       1 1 0 1
       -------
         1 0 0 0
         1 1 0 1
         -------
           1 0 1 0
           1 1 0 1
           -------
             1 1 1 0
             1 1 0 1
             -------
                 1 1 0 0
                 1 1 0 1
                 -------
                   0 0 1

Send: 1 0 0 1 0 0 / 0 0 1
```

Above is the long-division process that gives us the result (001). First, since the generator is 4 bits, we know that the remainder must be 3 bits (it is always one less bit). Thus, we tag 3 zeros onto the end of the datagram and divide by the generator. With modulo-2 arithmetic, however, it's a bit different. First, we "subtract" the generator each time the leftmost bit in the dividend is a 1 (we do nothing if it is a 0). Second, "subtraction" is just a bitwise XOR operation (no carries). Thus, at the end of this process we arrive at the remainder, 001. This is tagged onto the end of the data before transmission.

Now suppose that we receive the packet [100 100 010], knowing that the generator is still 1101. Is the packet corrupted? The solution is shown below.



```
          111101
1101 / 100100010
       1101
        1000
        1101
         1010
         1101
          1110
          1101
           1110
           1101
           0011 ≠ 0000
```

To determine whether the packet is corrupted, we simply divide the entire packet (data + CRC) by the generator. If the remainder is not equal to zero (i.e. the generator does not divide the packet evenly), then the packet is corrupted. In this example, the division results in a nonzero remainder, so the packet must be corrupted.

Summary: To determine the remainder (CRC) to send, calculate $R = rem(\frac{D*2^r}{G})$. To detect an error, determine whether or not $rem(\frac{D*2^r}{G}) = R$, or alternatively, whether or not $rem(\frac{D*2^r+R}{G}) = 0$.

7 Multiple Access Protocols

Collisions occur when multiple signals arrive at a node simultaneously. They must be resolved with multiple access protocols. The various strategies of these protocols are covered in the next lecture.