IOWA

CS3640

The finals: Technical Interview

Prof. Supreeth Shastri
Computer Science
The University of Iowa

Structure of the Interview

1-on-1

conducted by your instructor over zoom

25%

your interview determines quarter of your final grade

78 slots

schedule your interview at a time that works best for you

15 mins

in which you will be asked 4 questions, all carrying equal weights

Q banks

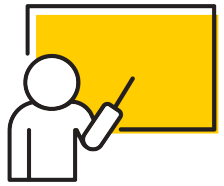
Each question comes from a distinct category (or bank); more in the next slide

Question Categories and Selection

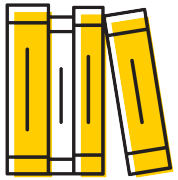
Category	Example questions and topics	Weight
Networking **Principles**	*End-to-end argument; Packet switching; Routing and forwarding*	25%
Internet **Protocols**	*HTTP headers and extensions; TCP RDT state diagrams*	25%
Numerical **Problems**	*Construct Dijkstra's LS table; Calculate networking delays*	25%
Network **Programs**	*Explain how traceroute works; Analyze Wireshark output*	25%

Each category has a bank of 8-10 question; for each student, I will generate a sequence of four random numbers that will determine the specific questions picked from each bank.

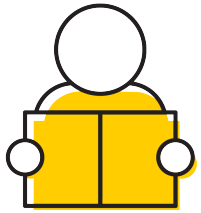
How to Prepare



*Revisit the **lectures and slides**:*
https://shastri.info/teaching/cs3640



*Read the **textbook**:*
Kurose-Ross chapters 1-6



*Practice the **material**:*
assignments, quizzes, textbook exercises

Keep a pen and paper handy to work out numerical problems and as aids to visualization

Ground rules and policies

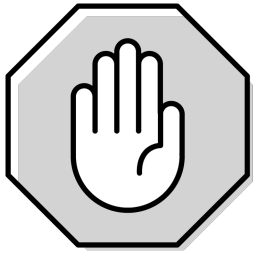
Time management

is completely your responsibility. I will not double guess your strategy (i.e., split time 4-ways, or skip harder ones, etc)

On camera

All interviews will be recorded to ensure academic integrity, and to sort out any grading discrepancies.

Please bring your student ID card.



Accommodations

If you need any special accommodations for the interview, please discuss with me at least a week before your slot.

Code of conduct

During the interview, you cannot use books, notes, Internet resources, or seek help from others.

After the interview, you should not share interview details with anyone.

Last set of assignments will be out!

Programming Project-2: **Implement Ping**

An important tool for network programmers and users

- implemented using ICMP and raw sockets
- we will provide a sample code with missing parts

Late submissions will not be accepted

Written Assignment-4: **Numerical Problems**

Serves as a reference for numerical problems to expect in interviews

- this will be the last of the written assignments
- written assignments will still contribute to 25% of final grade

ICMP: Internet Control Message Protocol

Network Working Group
Request for Comments: 792

J. Postel
ISI
September 1981

Updates: RFCs 777, 760
Updates: IENs 109, 128

INTERNET CONTROL MESSAGE PROTOCOL

DARPA INTERNET PROGRAM
PROTOCOL SPECIFICATION

Introduction

The Internet Protocol (IP) [1] is used for host-to-host datagram service in a system of interconnected networks called the Catenet [2]. The network connecting devices are called Gateways. These gateways communicate between themselves for control purposes via a Gateway to Gateway Protocol (GGP) [3,4]. Occasionally a gateway or destination host will communicate with a source host, for example, to report an error in datagram processing. For such purposes this protocol, the Internet Control Message Protocol (ICMP), is used. ICMP, uses the basic support of IP as if it were a higher level protocol, however, ICMP is actually an integral part of IP, and must be implemented by every IP module.

ICMP Message Format

	Bits 0–7	Bits 8–15	Bits 16–23	Bits 24–31
ICMP Header (8 bytes)	Type of message	Code	Checksum	
	Header Data			
ICMP Payload (*optional*)	Payload Data			

For echo request and reply

- Type == 8 (for request) or 0 (for reply)
- Code == 0 (for both)
- Header Data == 16-bit ID and 16-bit sequence number
- Payload Data == NULL

How to Send and Receive ICMP Messages

- Socket API has a special type: SOCK_RAW that allows direct access to IP datagrams

Type	Code	Description
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

IOWA

CS3640

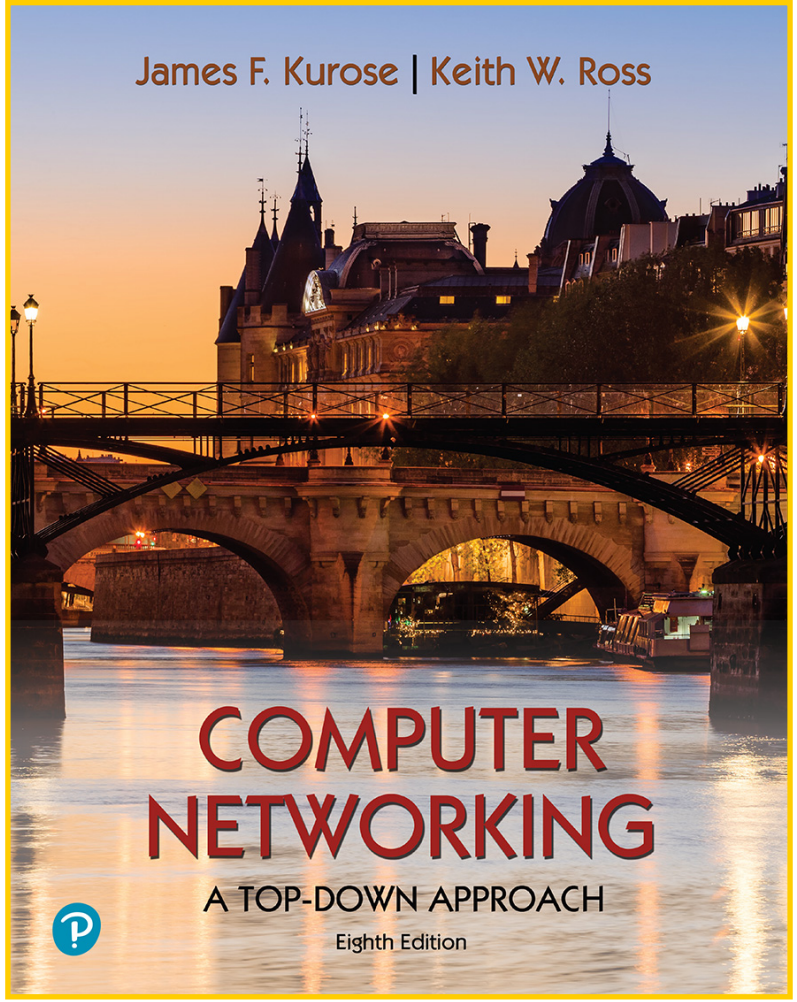
Link Layer (1): Multiple Access Links

Prof. Supreeth Shastri
Computer Science
The University of Iowa

Lecture goals

a technical overview of the data link layer

- *Link layer services*
- *Network Interface Controller (NIC)*
- *Multiple Access Channels*



Chapters 6.1, 6.3

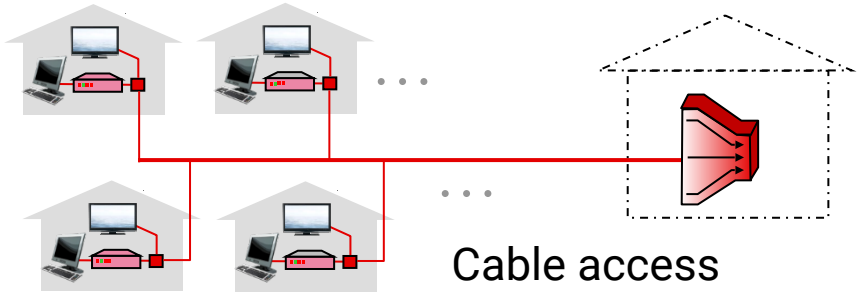
IOWA

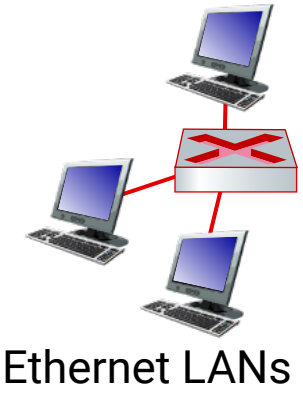
Link-layer Overview

Goals

- Transfer IP datagrams from one node to its *physically adjacent* node over a link
- Serve as an interface to the physical layer networking devices

Link layer technologies





Link-layer Services

Framing

- encapsulate network datagram into link-layer frame, adding header and trailer

Link sharing

- protocols that govern how multiple nodes share a broadcast medium
- identify source and destination nodes via MAC address (a link-layer exclusive ID)

Error detection and correction

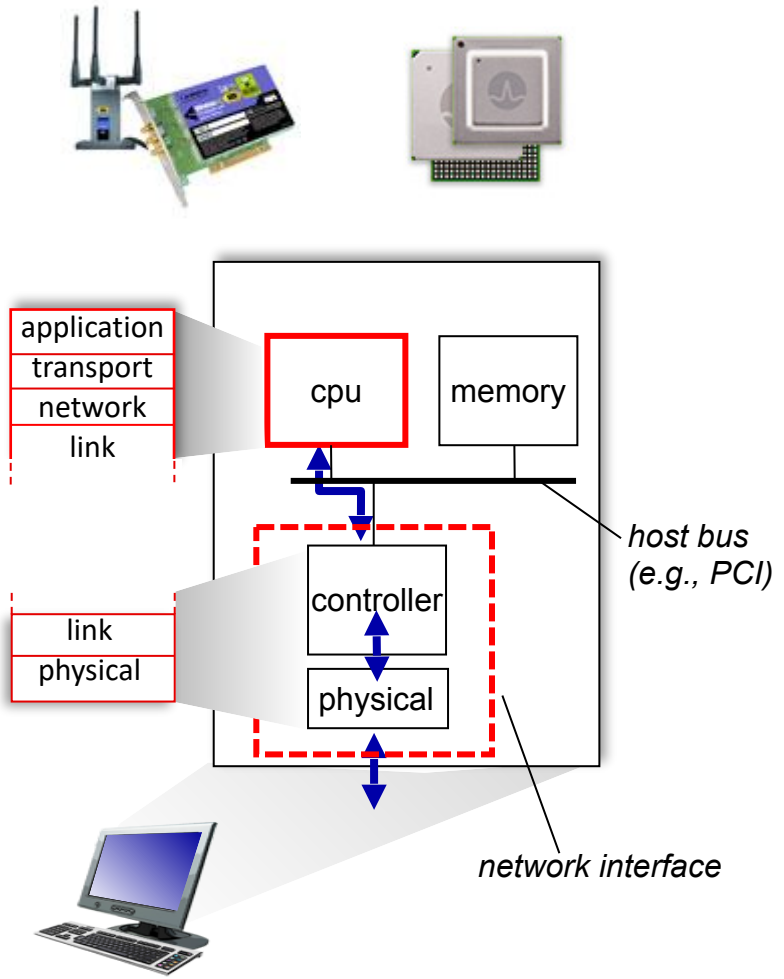
- mechanisms to detect errors, drop frames, and signal retransmission
- techniques to correct errors without the need for retransmission

Other services

- flow control, reliable frame transfer, directionality (half or full duplex)

Link-layer Implementation

- Link layer exists in each-and-every host
- Typically, implemented as a combination of hardware, software, and firmware
- Example: network interface card (NIC)
  - NIC implements physical layer and parts of link-layer
  - NIC attaches into host's system buses
  - Some link-layer functionalities are implemented in TCP/IP software stack



Multiple Access Protocols

sharing a single broadcast channel amongst multiple nodes

1. Channel Partitioning Protocols

divide channel into small pieces (e.g., time slots, frequency), and allocate each piece to one node

2. Random Access Protocols

do not divide the channel, and allow nodes to transmit at any time, but detect/ recover from collisions

3. Taking-turns Protocols

nodes take turn to send frames; this dynamism achieves balance between the first two class of protocols

Spot Quiz (ICON)