Important networking problem: multi-user communication

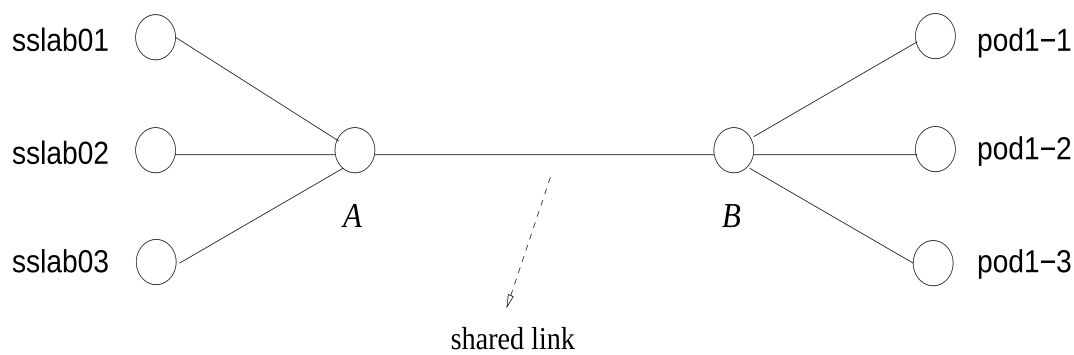
→ A and B talking to each: single-user communication

→ not too big a deal

→ challenging problem: many users talking to each other over shared link

→ e.g., A and B are routers/switches

Example scenario:



→ in LWSN 148/158: A and B are Ethernet switches

→ Ethernet switches are multi-access links (not point-to-point)

→ could replace A, B with IP routers with non-Ethernet NICs

Issue: link between A and B carries multiple bit streams belonging to different users

→ how to share?

First solution:

Suppose copper wires have "bandwidth" 100 MHz–102 MHz.

Assign:

→ frequency 100 MHz to sslab01—pod1-1

→ frequency 101 MHz to sslab02—pod1-2

→ frequency 102 MHz to sslab03—pod1-3

→ also called carrier frequency

Perform amplitude modulation (AM) on each frequency.

→ 3 parallel lanes

→ bandwidth (bps) of each?

Potential issues/complications?

Router B hears the **combined** signal (from sslab01, sslab02, sslab03) arriving on its interface from router A.

B needs to split the combined signal apart into its 3 component signals and send

$\rightarrow$ signal from sslab01 onto its first link (to pod1-1)

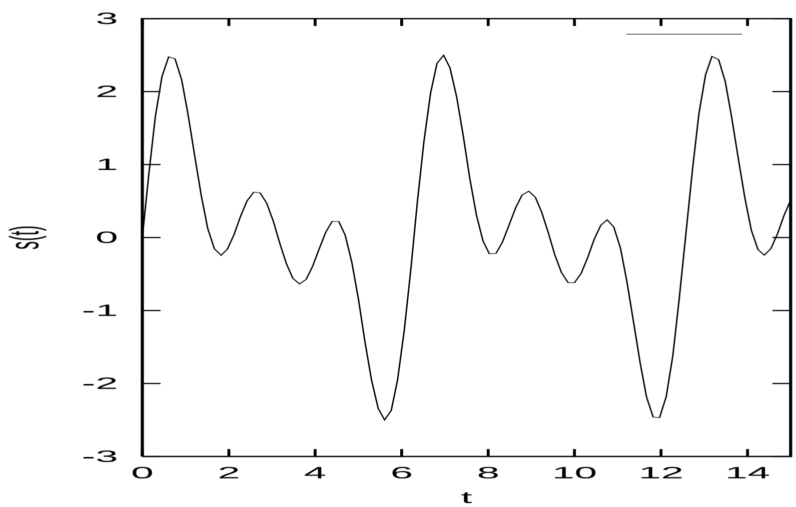
$\rightarrow$ signal from sslab02 onto its second link (to pod1-2)

$\rightarrow$ signal from sslab01 onto its third link (to pod1-3)

Technical issue: is this doable?

Illustration of issue:

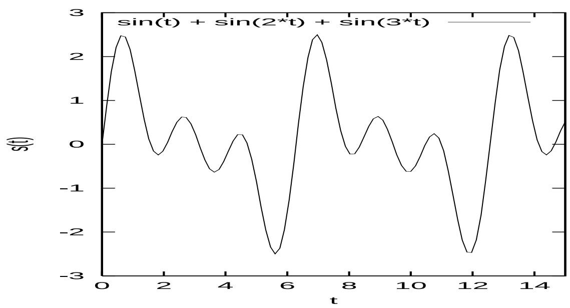
Suppose B receives the following combined signal from A.

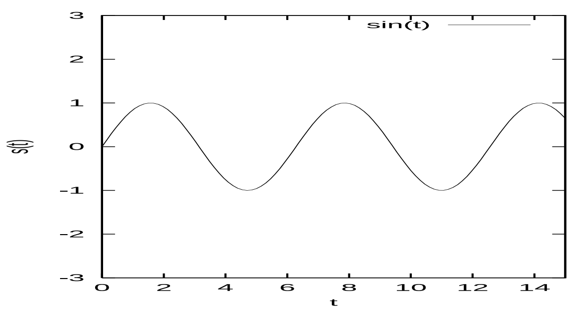


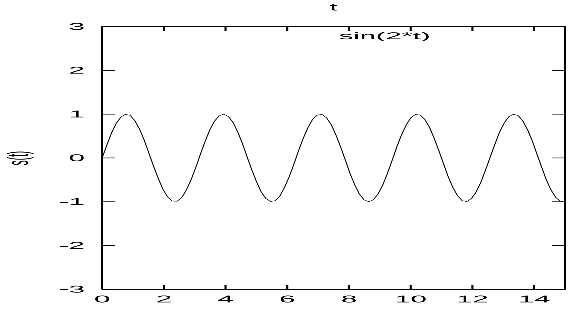
→ what were the three signals sent from sslab01, sslab02, sslab03?

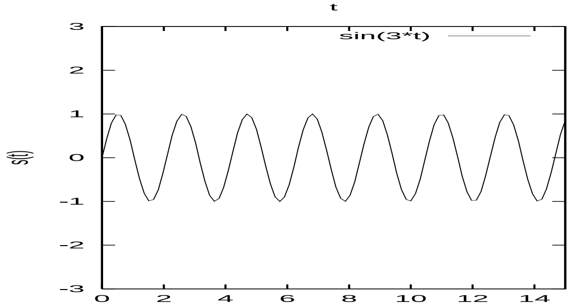
→ need to know to decode bits bits sent on three parallel streams

Answer: combined signal is sum of three signals below









When combined signals can be split apart is at the crux of modern high-speed wired/wireless communication.

Practical issue:

→ given bandwidth 100 MHz—102 MHz how many carrier frequencies can we squeeze in?

→ e.g., is 100.0, 100.1, 100.2, . . ., 101.8, 101.9, 102.0 MHz possible?

→ would support 31 simultaneous users!

→ called frequency division multiplexing (FDM)

We will look into it and see how it's done in real networks.

→ not difficult

→ but not easy to learn from books (too much info to wade through)

→ not easy to learn from web (too much junk info)

→ not easy to learn on the job (those who know are too busy)

Second solution to multi-user communication problem:

→ share a carrier frequency by splitting up time

Ex.: sslab01, sslab02, sslab03 sharing link between A and B using single carrier frequency (say 100 MHz)

→ divide time into blocks

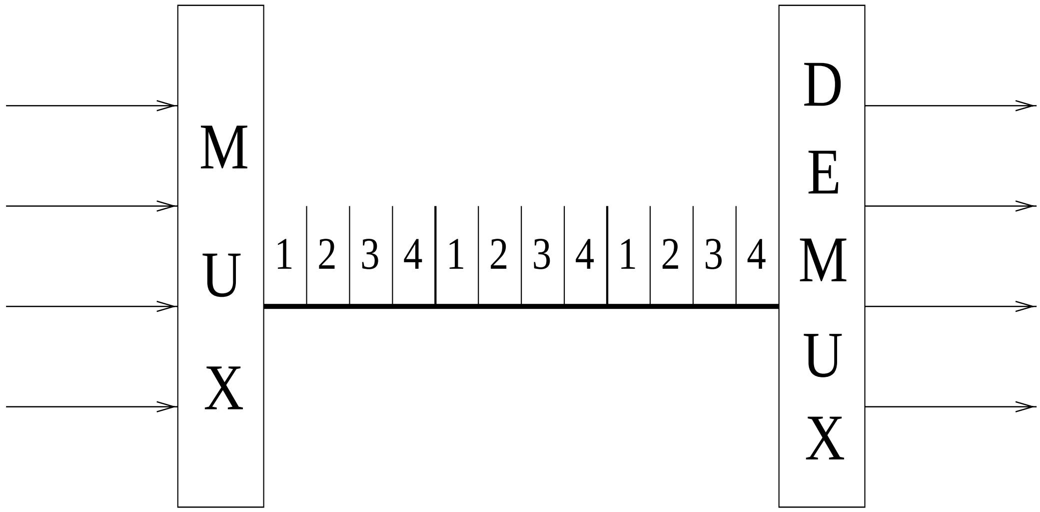
→ reserve blocks to 3 users: 1, 2, 3, 1, 2, 3, . . .

→ e.g., sslab01 gets to use time slots #1, #4, #7, etc.

→ sslab02 gets to use time slots . . .

→ called time division multiplexing (TDM)

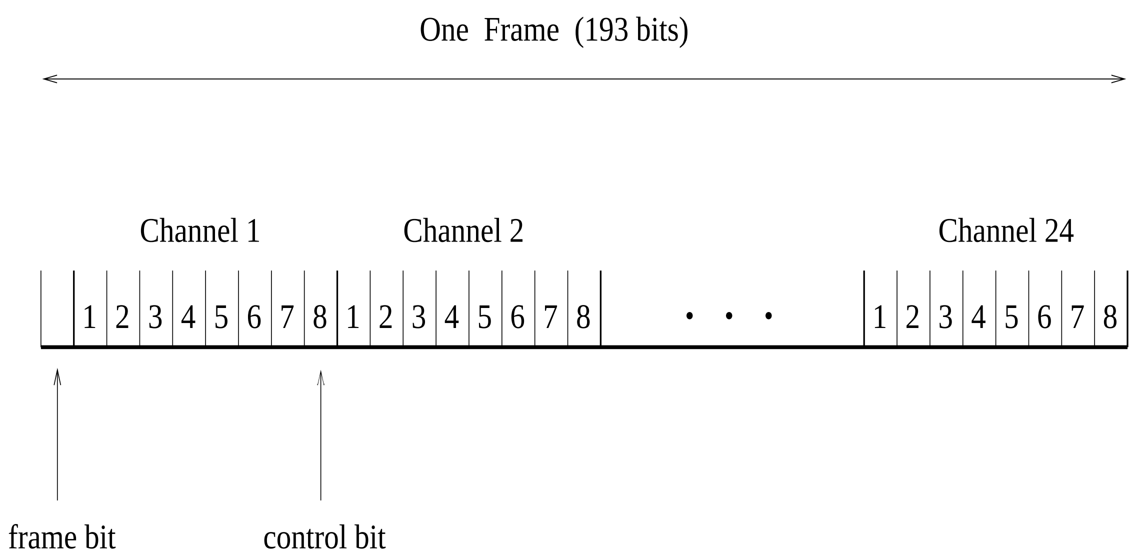
Illustration of 4-user TDM system:



→ router A: MUX (multiplexer, or combiner)

→ router B: DEMUX (demultiplexer, or splitter)

Real-world example: T1 carrier (1.544 Mbps)



- multi-user: 24 simultaneous users
- time slot: 8-bit block (each user gets to send 8 bits at a time)
- squeeze 8000 frames (or packets) into 1 second time interval

  $\rightarrow$ frame duration: 125 μsec
- bandwidth (bps): $8000 \times 193 = 1.544$ Mbps

T1 line is still in use today.

$\rightarrow$ T3 line: 44.736 Mbps

What are potential problems with TDM way of enabling multi-user communication?

Do they also apply to FDM?

FDM + TDM hybrid

→ use multiple carrier frequencies (e.g., 100 MHz, 101 MHz, 102 MHz

→ but share each carrier frequency among multiple users

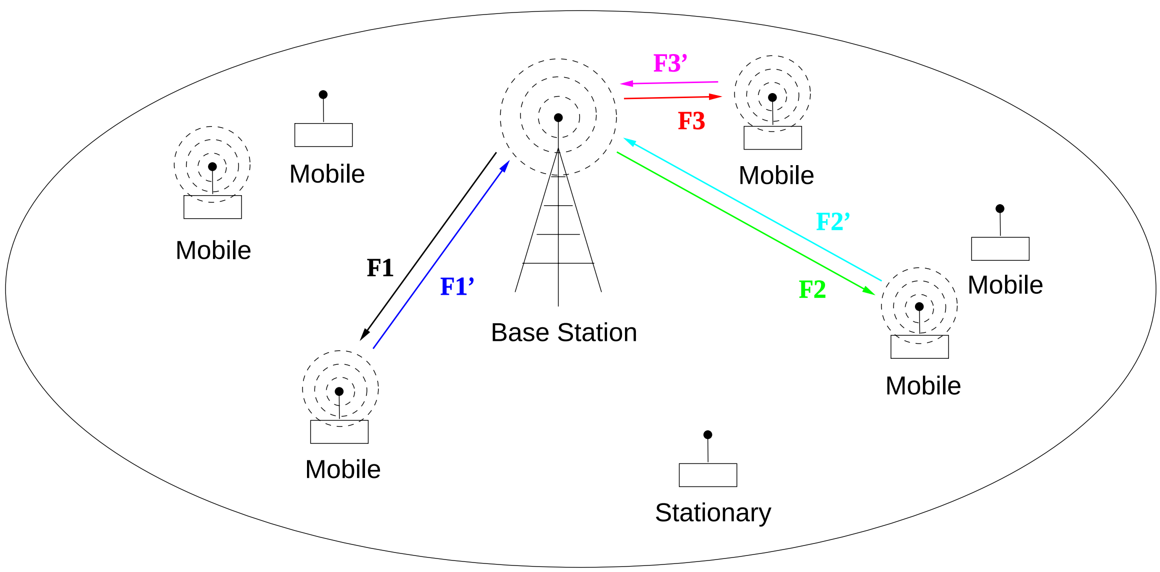
→ e.g., AT&T Wireless, T-Mobile use hybrid FDM + TDM technology

→ also called GSM in telephony world

→ what do Verizon Wireless and Sprint Nextel use?

Real-world example: FDM + TDM hybrid

→ U.S. IS-136 (GSM)



FDD & TDMA

- uplink: 890–915 MHz
- downlink: 935–960 MHz

  → 25 MHz bandwidth
- 125 channels 200 kHz wide each ($= 25000 \div 200$)

  → FDM portion

- 8 time slots within each channel

  $\rightarrow$ TDM portion

- up to a total of 1000 possible users $(= 125 \times 8)$

  $\rightarrow$ 124×8 realized

- codec: 13.4 kb/s

  $\rightarrow$ voice audio is also compressed