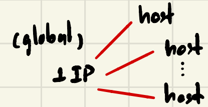
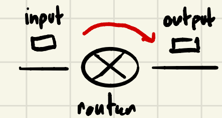
Chapter 4 → Network Layer

- **NAT (Network Address Translation)** → สามารถทำให้ IP 1 IP ใช้กับ host ได้มากกว่า 1 host.



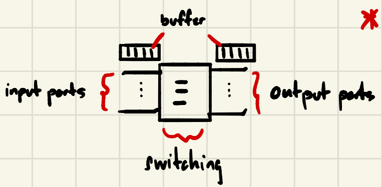
- **middleboxes** → อุปกรณ์คอมพิวเตอร์ - Firewall กรอง traffic

- **NETWORK**
  - → **Data plane** - Hardware, fowarding, Per-router function, พิจารณา packet ที่มาจาก input และพิจารณาจาก ip address ของ host ปลายทาง
  - → **Control plane** - Software, Routing.



- **Router** → ประกอบไปด้วย input ports, switching, output ports, buffer management, scheduling (การปล่อย data ไหนออกก่อน.)

[IPv4 → 32 bit, IPv6 → 128 bit.]



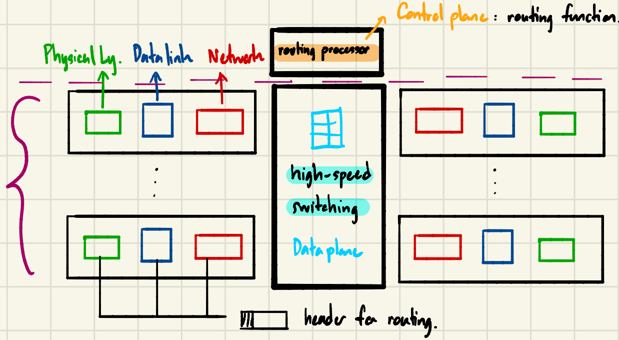
* การทำงานส่วนใดส่วนหนึ่งอาจจะทำงานเร็วกว่าส่วนอื่นๆ ก็เป็นได้ จึงมี buffer คอยเก็บ data เพื่อไม่ให้ data หาย *

- **SDN (Software Define Network)** → ช่วยแยก **data plane** และ **control plane** ออกจากกัน
  - ↳ (data plane) อยู่ใน router และมี **fowarding table** → เพื่อบอกว่า data นี้ควรออก output ช่องไหน

- **Services of Network Layer**
  - Sender : **encapsulates segments** → datagram ใส่หัว (address) ให้กับ data
  - Reciever : **deliver segments** to transport ถอดหัวและส่ง data ให้กับ transport
  - Router : ส่ง data จาก source → destination (end to end) (โดยผ่านการ routing (การจัดเส้นทาง.))
  - Guarantee delivery (w/ less than x ms delay)
  - In-order of flow data, guarantee min. bandwidth to flow

- **Functions of Network layer.**
  - forwarding : ย้าย packet จาก $input_{router}$ → $output_{router}$ { **data plane** : forwarding table (→ router.)
  - routing. : พิจารณาเส้นทางของ packet ว่าควรส่งไปทางไหน (แผนการเดินทาง.) { **control plane** : routing algorithm
    - routing algorithm (router)
    - ↳ SDN (remote controlled) → มี Controll Agent
  - (การเดินทาง)

• Router Components



} management control plane (SW) (ms)

↑ faster.

} forwarding data plane (HW) (ns)

INPUT FUNCTIONS

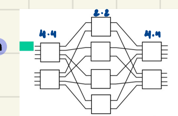
- Physical Layer : NRZ, RZ, manchester...
- Data link layer : stop and wait, go back N, Selective ARQ.

} Data Com.

- Network layer
  - using header fields value (macth plus action)
    - ↳ ใช้ forwarding table เพื่อหา output port.
  - Queuing : เกิดขึ้นเมื่อ datagrams มาถึงเร็วกว่า อัตราการส่งต่อข้อมูลผ่านของ switch (fowarding rate) → ปัญหา
    1. buffer overflow
    2. Head of the line blocking (HOL)
       ↳ ตัวหัวไปพร้อมกันไม่ได้ต้องรอ
  - Decentralized switching : มี fowarding table อยู่ใน header ของ input.

SWITCHING FUNCTIONS

- switching rate → N·R → line rate
  - ↳ N input
- 3 major types 1. memory / 2. bus / 3. interconnection.
  - ↳ backbone
  - ↳ crosspoint
    - ↳ จน. crosspoint = $N_{input} \times N_{output}$
  - multistage switch



8x8 multistage switch
built from smaller-sized switches

จน. crosspoint = 2(4×4) + 4(2×2) + 2(4×4)

  - parallelism : ในทางเข้าแบ่ง datagrams เป็นชิ้นย่อยๆ ก่อน
    ในทางออกรวม datagrams กลับมาเป็นก้อน

OUTPUT FUNCTIONS

└ Network layer
  - Queuing : เกิดต่อเมื่อ switch ทำงานเร็วกว่า output port. → ถ้า buffer เต็ม → Drop policy (Data loss)
  - Scheduling : เลือก datagrams ก่อนส่งออก (Priority)

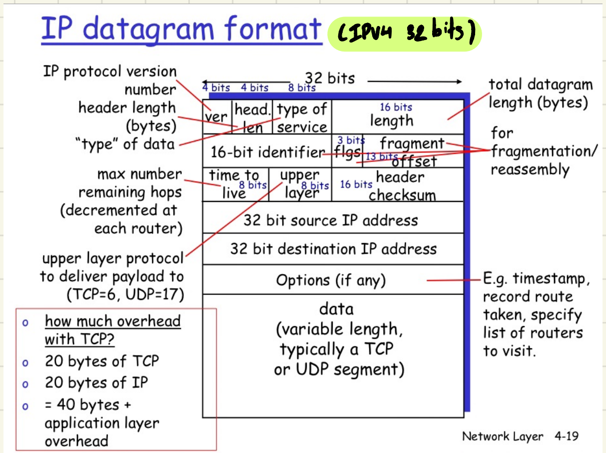
1 flow: average buffering = RTT × $Link_{BW}$

N flow: average buffering = $\frac{RTT \times LinkBW}{\sqrt{N}}$

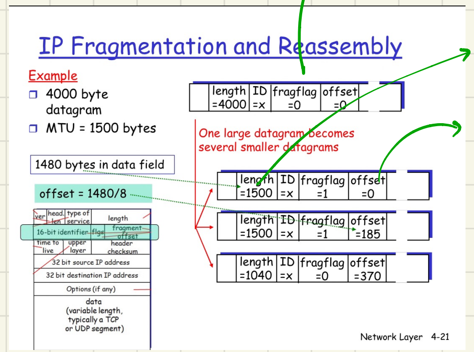
1.40 นาทีไหน.

- Internet in network layer.
  - 3 ส่วน
    - IP Protocol - บอก field ของ datagram, องค์ประกอบต่างๆใน packet.
    - Routing Protocol - จัดเส้นทางให้ datagram (Fowarding table)
    - ICMP Protocol - บอกว่าเกิด error หรือไม่, router signaling.

- IP datagram format.



IP datagram format (IPv4 32 bits)
IP protocol version number
header length (bytes)
"type" of data
max number remaining hops (decremented at each router)
upper layer protocol to deliver payload to (TCP=6, UDP=17)
32 bits
4 bits 4 bits 8 bits
16 bits
ver
head. len
type of service
length
16-bit identifier
flgs
3 bits
fragment offset
13 bits
time to live
8 bits
upper layer
8 bits
header checksum
16 bits
32 bit source IP address
32 bit destination IP address
Options (if any)
data (variable length, typically a TCP or UDP segment)
total datagram length (bytes)
for fragmentation/ reassembly
E.g. timestamp, record route taken, specify list of routers to visit.
how much overhead with TCP?
20 bytes of TCP
20 bytes of IP
= 40 bytes + application layer overhead
Network Layer 4-19



IP Fragmentation and Reassembly
Example
4000 byte datagram
MTU = 1500 bytes
1480 bytes in data field
offset = 1480/8
length =4000 | ID =x | fragflag =0 | offset =0
One large datagram becomes several smaller datagrams
length =1500 | ID =x | fragflag =1 | offset =0
length =1500 | ID =x | fragflag =1 | offset =185
length =1040 | ID =x | fragflag =0 | offset =370
Network Layer 4-21

- fragflag - 0 ไม่แบ่ง < initial state / final state | 1 แบ่ง
- length. - data length + header length → 20-60 (options); 20 (no options)
- offset - 1st offset = 0, 2…n ofset = $\frac{\text{data length}}{8}$

Sol'n
1st frame: L = 1500 = 20 + 1480 data, os = 0
2nd frame: L = 1500 = 20 + 1480 data, os = 185
3rd frame: L = 1040 = 20 + 1020 data, os = 370

Conclu.
all data = 1480 + 1480 + 1020 = 3980 bytes
header = 20 no options.
final offset = 370

- head. length - มีขนาด 20 bits (no options), มีขนาด > 20 bits ≤ 60 bits (options)
- length = head. length + data length
- 16 bit identifier / flags / fragment offset
  - fragmentation → แยก datagram ให้เป็นชิ้นเล็กๆ
  - reassemble → รวมรวมกลับ
- Time To Live (TTL) - ระบุจน. hop ที่สามารถผ่านไปได้
- Upper Layer - 6 (TCP), 17 (UDP)
- Checksum - check error
- Options - มีขนาดสูงสุด 40 bytes

IP Fragmentation, IP reassembly
- MTU (Maximum Transfer Unit) - ขนาดสูงที่สุดของข้อมูลที่สามารถส่งได้ใน 1 frame (ไม่รวม header, tailer.)
- fragmentation - เกิดขึ้นที่ router., reassembly - เกิดขึ้นที่ host des.

- **IPv4 Addressing** → 32 bits, identifying host or router interfaces ↷ การเชื่อมต่อระหว่าง host/router และ physical link
  - host: มีเพียง 1 interface
  - router: มี > 1 interfaces.
  - class-full addressing - 5 class →
    - class A. byte 1 : network, byte 2-4 : host (0-127)
    - class B. byte 1-2 : network, byte 3-4 : host (128-191)
    - class C. byte 1-3 : network, byte 4 : host (192-223)
    - } host, router interface
    - class D — all byte : multicasting
    - class E — future

* หา จน. network | จน. host *

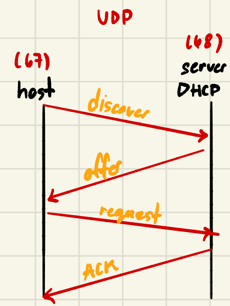
คิด byte (ตัวที่)

[(Last ip range - first ip range) + 1] × [(Last ip range - first ip range) + 1]

- **Subnet.** → ถ้า host อยู่ใน subnet เดียวกัน host จะสามารถส่งข้อมูลได้ถึงกันโดยไม่ผ่าน router.

- **CIDR (Classless Inter Domain Routing)** → subnet mask = /x บอกจน. bit ที่เป็น network

- **DHCP (Dynamic Host Configuration Protocol)** → แจก IP ให้กับ host แบบ dynamic
  - หลักการทำงาน
    - → host broadcast (DHCP discover) = [host ส่ง msg ไปหา server]
    - → DHCP server responds (DHCP offer) = [server ตอบกลับ]
    - → host req. IP address (DHCP request) = [host ยืนยัน ip จาก server]
    - → DHCP server send address (DHCP ACK) = [host ready to use ip]



- **NAT (Network Address Translation)** (Implement) → ช่วยให้ host หลาย host ใช้ ip มากกว่า 1 ip จาก global ip ได้.

  (host → global network)
  - Data local network → internet : datagram อยู่ในรูปแบบ private + port $\xrightarrow{NAT}$ global + port

  (global network → host)
  - Data internet → local network : /————/ global + port $\xrightarrow{NAT}$ private port.

- **Address Mapping** → Map จาก logical address (IP) ไป physical address (MAC) โดยใช้ protocol : ARP (Address Resolution Protocol)

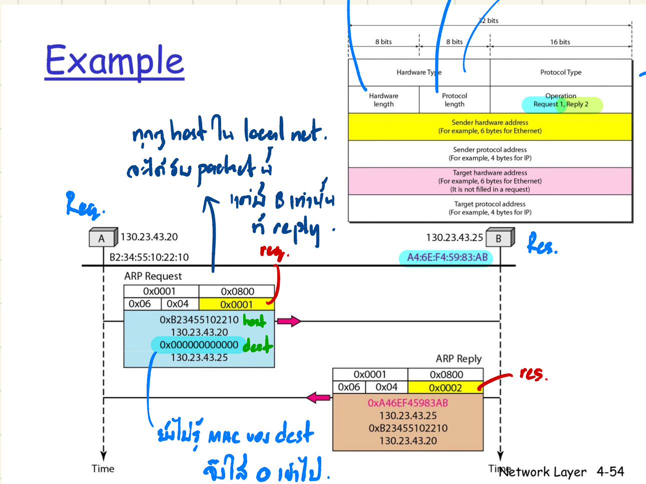
  ↳ ARP Packet + Example

Ethernet : 6

IPv4 ใช้ : 4

ถ้าเป็น 1 : Ethernet , 6 : IEE 802

IPv4 : 0800_{16}



Example
32 bits
8 bits
8 bits
16 bits
Hardware Type
Protocol Type
Hardware length
Protocol length
Operation
Request 1, Reply 2
Sender hardware address
(For example, 6 bytes for Ethernet)
Sender protocol address
(For example, 4 bytes for IP)
Target hardware address
(For example, 6 bytes for Ethernet)
(It is not filled in a request)
Target protocol address
(For example, 4 bytes for IP)
Req.
ทุก host ใน local net. จะได้รับ packet นี้
แต่มี B เท่านั้น ที่ reply.
req.
A
130.23.43.20
B2:34:55:10:22:10
130.23.43.25
B
A4:6E:F4:59:83:AB
Res.
ARP Request
0x0001
0x0800
0x06
0x04
0x0001
0xB23455102210 host
130.23.43.20
0x000000000000 dest
130.23.43.25
ARP Reply
0x0001
0x0800
0x06
0x04
0x0002
res.
0xA46EF45983AB
130.23.43.25
0xB23455102210
130.23.43.20
ยังไม่รู้ MAC ของ dest จึงใส่ 0 เข้าไป.
Time
Time
Network Layer 4-54

- **ICMP (Internet Control Message Protocol)** → unreliable
  - ↳ ข้อจำกัด! 1. lack of error control (error reported)
    2. host อื่น host is alive/not alive?
  - ↳ message 2 ประเภท
    - → Error-reporting → 5 ชนิด
      - 3 destination unreachable
      - 4. Source quench. no congestion
      - 11 Time exceeded
      - 12 Parameter problems
      - 5 Redirection
        ↳ บอกว่า Routing ไหนเร็วกว่า
    - → Query message → 4 ชนิด
      - 0 echo reply
      - 8 echo request
      - 9 route adver.
      - 10 route discovery

Time exceeded code → 0 : TTL expired
↳ 1 : Fragment Reassembly

Parameter problems code → 0 : Bad headers
↳ 1 : option part is missing.

Redirection code → 0 : network redirection
↳ 1 : host redirection

- **Ping** → โปรแกรมที่ใช้ตรวจสอบว่า host เปิดหรือปิด? และสามารถส่งข้อมูลได้หรือไม่ ?

- **Traceroute** → โปรแกรมใช้ตรวจสอบเส้นทางว่าใช้เส้นไหนบ้างใน network
  ↳ ใน router มี data อะไรบ้าง.

- IPv6
  - ขนาด Header 40 bytes
  - ไม่มีการแยกย่อย datagram (fragmentation)
  - Tunneling
- Components / field
  - traffic class → congestion control, Non-congestion control.
  - flow label → QoS management.
  - payload range → ความยาวข้อมูลที่บรรจุ.
  - next header → UDP=17, TCP=6
  - hop limit → TTL (in ipv4)

* ไม่มีการแยกย่อย *
- Checksum
- Options

Chapter 6 Data link

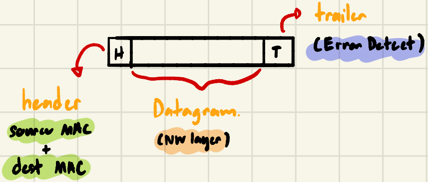
→ implement in Network Adapter.

Introduction

- node → hosts (PC), router
- link → เส้นที่อยู่ระหว่าง node ได้แก่ wired, wireless, LANs.
- data (หน่วย) ของ datalink → frame (encapsulated datagram)
- หน้าที่หลัก → ส่ง datagram จาก node-node
  - └→ เพิ่ม header และ trailer ให้กับ datagram = frame
- hop-hop delivery
  - └→ Algorithm in flow control.
    - Go back N
    - Selective
    - Repeat
    - } Data Com.
- MAC Address เรียกได้อีกแบบ Physical Address.

- error detection (ตรวจจับ.) เกิดจาก noise, etc. (trailer)
- error correction → แก้ไขข้อผิดพลาด.

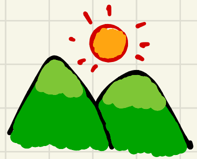
* ส่วนประกอบ.



} ทั้งก้อนเรียกว่า 'frame'

interface communication

- sending → encapsulate (add H, T)
- recieving
  - → look for errors
  - └→ extracts datagram to network layer.



Error detection and Correction. → trailer field

(พื้นฐานยังมีการตรวจสอบละเอียด)

- Parity bit checking
  - single bit
    - odd check: ถ้า data bit (1) เป็น จน.คู่ ∴ parity bit = 1
      - ————— คี่ ∴ parity bit = 0
    - even check: ————— คู่ ∴ parity bit = 0
      - ————— คี่ ∴ parity bit = 1
  - two dimensional → check both row + column.

- CRC → มีค.แม่นยำสูง
  - Analysis formula: $\frac{x^g \cdot D(x)}{G(x)} = Q(x) \oplus \frac{R(x)}{G(x)}$
    - highest expo in D(x)
    - Dataword (initial code)
    - Quotient.
    - remainder.
    - Divisor.
  - Codeword C(x): $x^g \cdot D(x) \oplus R(x)$
    - └ ชุดข้อมูลที่ถูกต้อง.

- check sum → เช็คที่ trailer ของ recieve. (16 bits)
  - ถ้าตรงกับที่ให้ไป: no errors, pass
  - ถ้าไม่ตรง: errors!

Multiple Access Protocol.

- 2 types
  - point to point → host to host.
  - broadcast → shared
- collision → การชนกันระหว่าง signals 2 สัญญาณขึ้นไป.
- calculate with multiple access channel (MAC) [กำหนดให้ ช่องนี้ส่งด้วย rate R bps]
  - ส่ง 1 node → ส่งเร็วในการส่ง "R bps"
  - ส่ง m node → ส่งเร็วในการส่ง "$\frac{R}{m}$ bps"

Type of Multiple Access Channel

- 3 Groups

Channel partitioning → การแบ่งช่องทางการสื่อสาร TDMA, FDMA, CDMA.

1. TDMA - ใช้ช่วงเวลาในการแบ่งในการส่งข้อมูล.
2. FDMA - ใช้ความถี่ในการแบ่งในการใช้ช่องทางติดต่อสื่อสาร.
3. CDMA - ใช้ code ในการเข้ารหัส.

Random access protocol

(ทำให้ไม่รู้ว่าจะส่งข้อมูลเวลาไหนดี.

- ★ Pure Aloha → no synchronization, unslotted
  - easy to get collision.
- ★ Slotted Aloha → frame มีขนาดเท่ากันทุกๆ frame
  - ถ้าเกิดการ 'collision' จะมีโอกาส p ในการส่งใหม่
- ★ CSMA (Carrier Sense Multiple Access)
  - ถ้าช่องว่าง(ปกติ cidle) → ส่งเลย. (เพราะรู้ว่าเปลี่ยนการส่งเสร็จ, เลย)
  - เกิดการ collision, busy → ยกเลิกและเลื่อนการส่งข้อมูล
- ★ CSMA/CD algo
  1. NIC ได้รับ datagram จาก NW. และ สร้าง frame ขึ้นมา (เพิ่ม header, trail)
  2. CH ว่าง → ส่ง, CH ไม่ว่าง → รอจนว่าง ค่อยส่งอีกที
  3. หลังจากการหยุดส่ง NIC จะทำการ 'binary exponential backoff'
     - มีการวน m ครั้ง และครั้งที่ m จะมีการสุ่มเลข K
     - จาก $0 - 2^{m-1}$ และทำการรอด้วย 'K·512 bit times'

back to step 2

(เลื่อนเวลาให้ node ส่งข้อมูลแบบไม่ชนกัน.

- node ทุกช่องมีอัตราเท่าเทียมกัน.
- เมื่อเกิด 'collision' ก็ต้องมีวิธี 'recover from collision' (delay retransmission)
- ตัวอย่างของ R.A.P. → Aloha, Slotted Aloha, CSMA, CSMA/CD, CSMA/CA (no collision, collision)

Taking turns → polling → loop checking devices Master จะเป็นตัวถาม Slaves ว่า Slaves ต้องการส่งข้อมูลหรือไม่ yes → ส่งมา no → ผ่าน.

- 2 types 1. central sites ปกติ, 2. token pass ลูปวงกลม.

LANs

- IP Address → 32 bit address (dec no.), use for interface. (network layer)
- MAC Address → 48 bit address (hex no.), hop to hop, subnet เดียวกัน (datalink layer) → MAC ไม่มีวันเหมือนกัน จัดการโดย IEEE
  - ↳ 24 bit แรก - fixed, 24 bit หลัง - random.
- switch ไม่มี interface ที่เป็น MAC Address.
- ใน ARP table จะมีตาราง IP, MAC, TTL → บอกว่าจะเก็บค่า MAC ของ IP นี้ไปถึงตอนไหน?
  - ↳ แรกเริ่มยังไม่รู้ว่า dest. MAC คือเครื่องไหน ∴ ใส่ boardcast เข้าไปแทน.
  - ทุก node ได้รับ "ARP query" แต่ใน boardcast นั้นมี target ip address แนบอยู่ / node ที่มี ip นี้ก็จะตอบกลับไปหาตัว request.

Ethernet

- การทำงานของ token ring, FDDI, ATM ซับซ้อน, ราคาสูง
- มีการปรับปรุง version + data rate ที่ดีขึ้น
- Hardware เช่น Adapter, switch ราคาถูกลง

} สาเหตุที่ยังใช้ ethernet อยู่.

- Ethernet frame Structure

8 bytes	6 bytes	6 bytes	type (2 bytes)	46 bytes (no option) - 1500 bytes	4 bytes
preamble	dest. address	source address		data (payload)	CRC

preamble - มีเพื่อ synchronize กับ reciever 7 bytes แรก → 10101010..., bytes สุดท้าย 10101011

dest / source address - บอก MAC ของ host ที่ S/D, ถ้า target ip ไม่ตรง host จะ discard frame ทิ้ง.

type - บอกประเภทของ protocol เช่น ipv4, ipv6, arp ...etc.

CRC - error detection.

Switch

→ 2 แบบ
① Switch layer 2 - ทำงานที่ physical + data link
② Switch layer 3 - 〃 physical + data l. + network

- Store, selectively foward
  - L ตารางสอบ
  - L ส่งไปหา port ที่เหมาะสม.
- มีกระบวนการ self-learning
  - ↳ switch จะมีข้อมูล (MAC, Port, TTL) ก็ต่อเมื่อ host ส่งข้อมูลมายัง switch.
- filtering / fowarding.
  - ถ้าไม่มีข้อมูลในตาราง → broadcast ไปทุก port. (flooding)
  - ถ้ามีข้อมูล → ส่งไปที่ port นั้นเลย.
  - (ถ้า host ได้รับ frame พร้อมกัน = collision → drop frame)

VLANS

→ ค.สามารถของ switch เพื่อให้มี subnet ให้มีมากกว่า 1 network

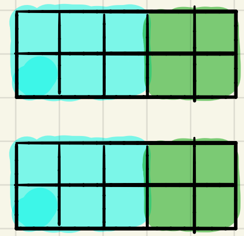
efficiency (ประสิทธิภาพ) → ลดปัญหา boardcast traffic
- → switch มีการจัด data เร็วขึ้น.
- → แบ่งแยก boardcast โดย boardcast domain.

security (ค.ปลอดภัย) → VLAN มองอุปกรณ์ข้าม VLAN ไม่ได้.

ต่อ. Interconnection in VLAN

1.) Two cables → เป็นวิธีการเชื่อมจาก port VLAN ไปยัง port VLAN อีกอัน.
แต่เปลือง! เพราะ ' n VLANs ใช้ n ports '

VLAN$_1$ 6 ports



— VLAN$_2$ 4 ports ∴ 10 ports = 10 cables.
เปลืองทรัพยากร.

→ 10 ports = 1 cable.

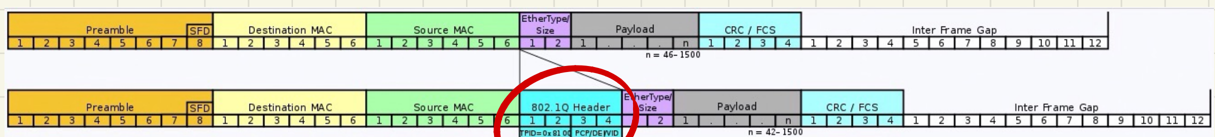
2.) Trunked → สามารถมีหลาย VLANs ภายใน cable เดียวได้
' ลดทรัพยากรลง, ประหยัด '

จะรู้ได้ไงว่า frame นี้มาจาก VLAN ไหน??

Ans: ในการส่ง frame ใน trunk link นั้น จะใช้ VLAN data frame (IEEE 802.1Q)
ซึ่ง switch ผู้ส่งจะป้อน tagged VLAN เข้าไปใน frame ส่วนผู้รับจะถอดออกตรวจสอบ.

(IEEE 802) traditional Ethernet frame. ←

(IEEE 802.1Q) VLAN data frame frame. ↙



Insertion of 802.1Q tag in an Ethernet frame

tagged bytes (4 bytes)

2 bytes แรก → TPID (Tag Protocol Identifier)
↳ มีค่าเป็น 0x8100 ' ระบุว่าเป็น IEEE 802.1Q '

2 bytes ถัดมา → Tag Control Information → 12 bits = VLAN ID
→ 3 bits = IP Type of Services
→ 1 bit = CFI

[Access port vs Trunk port]

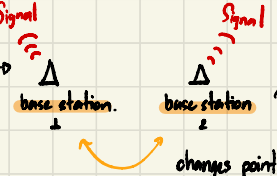
Access port
- port เชื่อมต่อระหว่าง Client-switch
- มี VLAN traffic เพียงเส้นเดียว
- มี mac address เพียงเส้นเดียว

Trunk port
- port เชื่อมต่อระหว่าง switch-switch, switch-router.
- มี VLAN traffic หลายเส้น
- มี mac address หลายเส้น.

Chapter 7 - Wireless

• WIRELESS VS MOBILITY

- Wireless → ไม่มีการเคลื่อนที่, เคลื่อนย้าย.
- Mobility → มีการเคลื่อนย้าย

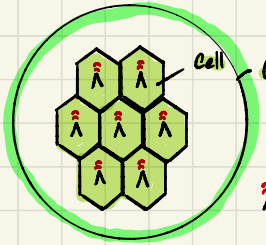


real reflection, blocked

สถานีกระจายสัญญาณ = ideal กระจายเป็นวงกลม

changes point of attachment = " Handoff, Handover "

• CELLULAR NETWORK (เครือข่ายโทรศัพท์เคลื่อนที่)



Cluster 7 เพราะมี cell 7 cells

→ base station.

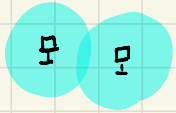
- เซลล์: cell ใช้ ก.ถี่ต่ำก้น
- ถ้า cell มี user จำนวนมาก ก็สามารถยืม ก.ถี่บน cell ที่อยู่ใกล้ๆมาใช้

• WIRELESS NETWORK ARCHITECTURE

① Infrastructure mode

* Base station เชื่อมต่อกับ wireless host เข้ากับ wired network
* `handoff' → changes of base station.

② Ad Hoc mode → ถ้า infra. mode พัง Ad Hoc ก็ใช้ได้.



* no base station
* work between node-node which in `Area coverage' ของเขตสัญญาณ
* node แต่ละตัวทำหน้าที่เสมือน router.

* มีการสลับสายวนเมื่อต้องการใช้ (on-demand routing)

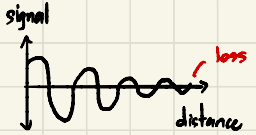
• ELEMENTS OF A WIRELESS NETWORK

① Wirless host or mobile host
↳ IoT devices connected by BS. or Access point (Base Station.)

② Base Station
↳ เชื่อมต่อเครือข่ายแบบมีสาย (wire network)
↳ relay - ส่งข้อมูลระหว่าง wired network - wirless hosts ใน พื้นที่ (Cell, Area)

③ Wireless link (ง่าย)
↳ ใช้ multiple access protocol เชื่อมต่อ
↳ transmission rates, frequency, distance ต่ำก้น

WIRELESS LINK AND NETWORK CHARACTERISTICS



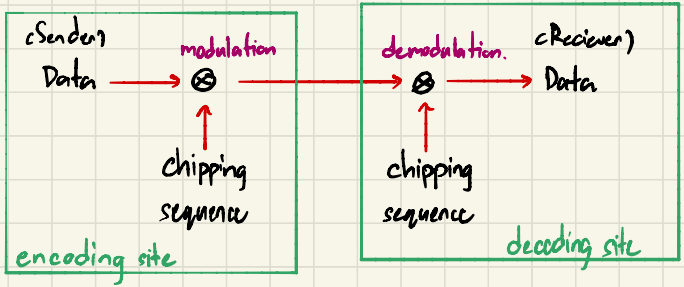
- ★ decrease signal strength: สัญญาณลดลงเมื่อผ่านวัตถุต่างๆ
- ★ interference from other sources: รบกวนจากแหล่งอื่น, สัญญาณต่ำกว่า, ช่วง ก.ถี่ที่ใช้มีการส่ง data เยอะไป
- ★ multipath propagation: การเดินทางของสัญญาณที่เดินทางต่างกัน → ทำให้ปลายทางได้รับข้อมูลไม่ตรง, ช้า, loss เกิด delay)

  → Multipath
  - Direct Line of Signal Path (ไปตรงๆ) best case
  - Diffraction (เลี้ยวเบน)
  - Scattering (กระจาย)
  - Reflection (หักเห)
- ★ SNR : Signal to Noise Ratio → SNR = Signal / noise 'SNR มีเยอะ ดีกว่า'
  - SNR เยอะ BER น้อย.
  - SNR น้อย BER เยอะ

  Bit Error Rate
- ★ Multiple senders, recievers → อาจจะทำให้เกิด interference (Hidden terminal problem)

CODE DIVISION MULTIPLE ACCESS (CDMA)

หลักการ
1. User มี code ของตัวเอง (chipping sequence) เพื่อ encode
2. user 'มี ก.ถี่ เท่ากัน'
3. ค่าเกือกการตีกันของสัญญาณ แสดงว่า code เหมือนกัน.



• IEEE 802.11 WLAN

- ★ โครงสร้าง → มี AP (Access Point) เป็น base station.
  - ↳ ถ้าเป็น Ad hoc mode จะไม่มี AP มาเกี่ยวข้อง
- ★ ช่องทางการติดต่อสื่อสาร (channel) → AP admin เลือกช่วงค.ถี่ในการสื่อสาร
  - ↳ สามารถเกิดการรบกวนกันได้
- ★ Arriving host → ต้องขอเข้าร่วม AP จึงจะสามารถใช้งานได้
  - ↳ beacon frame → SSID (ชื่อ AP)
    - ↳ MAC Address
  - → เชื่อมต่อ AP 3 ขั้นตอน.
    1. Wifi Network Discovery
    2. Authentication
       - run DHCP to get IP Address in AP's subnet.
    3. Association.
  - → Passive / Active Scanning.

└ Passive scanning. (AP ส่งสัญญาณมา host)

↳ host รอ beacon frame จาก AP

host เลือก AP โดยการ request ไปหา AP

AP response กลับไปหา host

done!

└ Active scanning. (host หาสัญญาณ AP ในพื้นที่)

↳ host ส่ง probe request frame ไปให้ AP ที่อยู่ใน Area

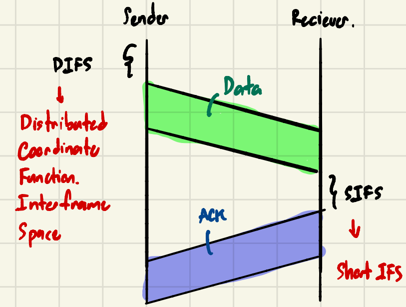
probe response กลับมาจาก APs

host เลือก AP โดยการ request ไปหา AP

AP response กลับไปหา host

done!

• CSMA IN WLAN



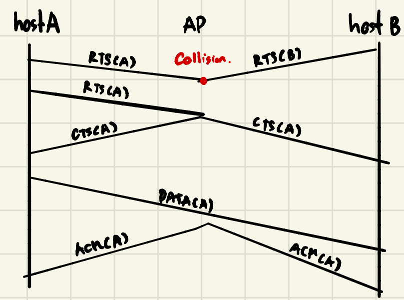
Sender

1. ถ้า channel ว่าง → รอระยะเวลา DIFS
   - → ถ้าไม่ว่างรอจนกว่าจะว่าง
2. เมื่อผ่าน DIFS แล้ว → idle → ส่ง data เลย
   - ↳ busy → sender จะสุ่ม back off time และจะนำเลขที่สุ่มได้มานับถอยหลัง
3. backoff time = 0 (+ว่าง), sender ส่ง frame รอ ACK
   - ↳ ถ้าไม่มี ACK ∴ เพิ่ม back off time → ขั้นตอน 2.

Reciever.

1. รอ SIFS หมด ค่อยส่ง ACK.

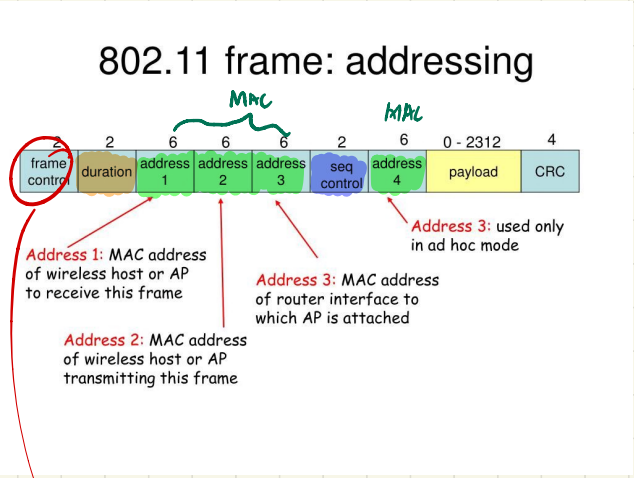
• Collision Avoidance : RTS - CTS exchange



RTS → Request to send

CTS - clear to send

• WLAN ADDRESSING



address 1 : Reciever host / AP

1 — 1 2 : Sender host / AP

1 — 1 3 : Router which AP is attached

1 — 1 4 : Used: Ad Hoc mode.

duration : RTS time, CTS time

seq. control : Duplicated frame

FIGURE 5.1 Control frame: Frame Control fields

B0										B15
Protocol version	Type	Subtype	To DS	From DS	More frag	Retry	Pwr mgt	More data	Protected frame	Order
Protocol version	Control	Subtype	0	0	0	0	Pwr mgt	0	0	0
Bits : 2	2	4	1	1	1	1	1	1	1	1

type : Control, Mangement, data

Subtype : Association, Disassociation, re-association, probe, beacon, authen, de-authen, CTS, RTS, Ack

Retry : Retransmission.

Power mgn : save mode

More data : buffered data in save mode