1) Computer System

[com → semiconductor / diff → microprocessor, microcontroller]

① **Mechanical Computer**
1) Abacus (ลูกคิด) → China, Europe, Russia
2) Machical calculator → ใช้เพื่อบวก, ลบ (1642 Blaise Pascal)

② **Vacuum Tube** (หลอดสุญญากาศ) ≈ 5 วัตต์
1) พัฒนาจากหลอดไส้ (หลอดไฟ) incandescent
2) มี guardgate ควบคุม electron ได้
↳ พัฒนาไปเป็น 1) หลอดขยายเสียง Amplifier
2) computer เครื่องแรก [eniac]
3) ขดลวดร้อน → ปล่อย electron จาก cathode → anode → ไฟติด
↳ diode (LED เปล่งแสง)
↳ กลืน ค.ร้อนในช่วงที่สายตาคนมองเห็น

③ **หลอด fluorescence** (หลอดนีออน) ← สารเรืองแสง
1) ข้างในเป็นสุญญากาศ → พอเปิด switch
- I ไหลจาก ΔV สูง → ΔV ต่ำ
- e^- ไหลจาก ΔV ต่ำ → ΔV สูง

ตอน e^- วิ่ง = ไฟติด!
2) ข้างในมีปรอทนิดหน่อย
↳ ทำให้เรืองแสงได้ดี
3) ขอบๆ เคลือบฟอสฟอรัส → ถ้า electron เคลื่อนที่ผ่านจะเรืองแสง **
4) เอาไปวางตรงที่ electron เยอะๆ → ฟอสฟอรัสเรืองแสง → ไฟติด!

* สารเรืองแสงในหลอด มี ค.ไว ต่อกระแสไฟฟ้า จึงทำให้หลอดไฟสว่าง

④ **Conductor** (ตัวนำไฟฟ้า) ex. เหล็ก อลูมิเนียม

⑤ **Semi conductor** (สารกึ่งตัวนำ)

⑥ **Register, Insulator** (ฉนวน) ex. แก้ว พลาสติก (กันไฟฟ้าไหลผ่าน)

⑦ **Silicon Transister**
1) เป็น semi conductor ที่สร้างจาก
- Silicon^{14} VVV
- Germanium^{32} ใกล้กับเป็น conductor

↳ มีไฟป้อนเข้ามา → ป้อนมาก → e^- ไหลผ่านได้น้อย

* Silicon 1 แผ่น ≈ 12 inches

* Silicon สร้างจาก sand, quarts, rock crystal, amethyst, agate, flint (หินเหล็กไฟ), jasper (หินอ่อน), opal

* วิธีสร้าง ทราย เผาโดยการ + oxygen / อัดไฟ → set เป็น pure silicon → ตัดเป็น Wafer → ฉายรังสี
1 แผ่น = chip 1 ตัว

1) Computer System

⑧ **Integrated Circuit** (IC , วงจรรวม)

↳ ใน chip 1 ตัว มี transistor มาต่อกันหลายตัว

↳ transistor ยิ่งเล็ก = สค.ซ้อน (e⁻ เคลื่อนที่สั้นๆ) = ประหยัดพลังงาน

* 2N 2222 A331 → NPN Transistor

* IC 555 (NE555N) → สร้างสัญญาณนาฬิกาได้

⑨ **Electronic Devices**

1) Passive Devices (อุปกรณ์ทั่วไป)

1) Capacitor (F) ตัวเก็บประจุ เก็บพลังงานเหมือนแบต (กรองค.ถี่ ให้ค.ถี่สูงผ่านๆ)

↳ Capacitor แบบ Electrovize

↳ ด้านในเป็นเจล ไม่ทนค.ซ้อน

↳ ตัวการ์ตูน คนพลัง?

2) Resistor (Ω) ตัวต้านทาน มีแถบสีระบุค่า

↳ 220 V → แดง แดง น้ำตาล ทอง

→ แดง ดำ น้ำตาล ทอง

3) Inductor (H) ขดลวด ขดลวดสร้างสนามแม่เหล็ก (กรองค.ถี่ ให้ค.ถี่ต่ำผ่านๆ)

4) Transformer หม้อแปลง

↳ ค.ถี่สูง , แกน Carbon

2) Active Devices (สารกึ่งตัวนำ)

1) Diode กันแรงดันไหลย้อนไม่ให้กลับขั้ว (Anode / Cathode)

2) Transistor

3) Light Emitting Diode (LED)

4) IC , Chip

3) Component

1) Print Circuit Board (PCB)

⑩ **Processors** (CPU)

1) Microprocessor ตัวประมวลผลในคอม ใช้ประมวลผลทั่วไป (CPU ex. core i7)

1) คำสั่งทั่วไป +, -, x, ÷, jump (ข้อมูลใน mem), copy

2) ไม่มี I/O port / ไม่มี mem

3) Multicore ใช้คำสั่งหลากหลาย (core ใหญ่, น้อย)

↳ ex. x86 , x64 , i7 , 80286

2) Graphic Processor Unit การ์ดจอ

1) ชุดคำสั่งพิเศษ → Vector, Graphic, Video

2) มีสายต่อ mem 2 เส้น

3) Multicores (core เล็กมาก, แต่เยอะ) → แบ่งกันประมวลผล

4) ก.คำนวณ Memory คำนวณเป็น Vector

↳ ex. Geforce , Radeon , Quadro , FirePro

1) Computer System

Sound Card

3) Digital Signal Processors (DSP) ex. เครื่องเล่นเปิดไฟฟ้า, router wifi, MP3
- ↳ ประมวลผล streaming signal
  - ↳ ส่งสัญญาณเสียง
  - ↳ ส่งสัญญาณภาพ ไม่ใช่การส่งข้อ, เป็นการเข้ารหัส < Encoding / Decoding
- ↳ เกี่ยวข้องกับด้านเสียง
- ↳ มีขนาดเล็กมาก
- Analog → Digital ช่องรูเล็ก หูฟัง

ex. family chips → C6000 (Texas Instrument), SHARC (Analog Devices), EMU10K (Creative)

4) Microcontroller ex. Arduino [มี mem, I/O, timer, DAC, ADC]
1) เน้น input / output
2) เน้น timers, ไม่เน้นค.เร็ว, ระบบควบคุม
   - ↳ Microwave
   - ↳ เครื่องซักผ้า
3) คล้าย Microprocessor แต่ช้า
4) ขนาดเล็ก, ใช้งานเล็กๆ, มี mem
5) ใช้สำหรับส่งผ่านข้อมูล, คำนวณน้อยๆ

* CPU SPEC

*** ของบริษัทไร ? - ? ***

1) Z80 → ปี 1980 → เครื่องเล่นเกม
2) 8088 → ปี 1980 → Intel
3) 80286 → Intel 16 bit
4) 80386 → Intel 16 / 32 bit
5) Pentium → รุ่นแรกที่เป็น 64, สุดท้ายที่เป็น 32

} Microprocessor

6) 8051 → เครื่องจักร
7) 68HC11 → ยานยนต์ (ระบบคุยกันผ่าน Bus)
8) PIC 16FXX → Electronic Devices ขนาดเล็กมาก
9) Arduino → Library เยอะ

} Microcontroller

2) Mem , I/O Addressing

ex. เวลา cpu ต้องการเขียนข้อมูลลงใน mem
- ใช้ Address bus ส่งตำแหน่ง
- ใช้ Data bus ส่ง Data

Clock Signal * สัญญาณนาฬิกา มี f, f เป็นตัวกำหนดจังหวะการทำงานของคอม, เชื่อมกับอุปกรณ์ทุกชิ้นในคอม

Control Signal * ส่งไปควบคุมสัญญาณทุกชิ้น ex. ควบคุม mem ในการอ่านเขียน, ควบคุม Port ต่างๆ
- คอมที่ไม่ซับซ้อน controller ฝังใน cpu , ซับซ้อนจะอยู่ใน chipset

Address Bus * สัมพันธ์กับจำนวน bit , สายพันเป็นกลุ่มๆ (จน.เส้นขึ้นกับจน. bit) * input devices ทุกชิ้นมี แบบ Address
* Address ไม่ซ้ำกัน
* แสดงผลเป็น 16
↳ 220H → Hex
↳ 0x 220 Hex
} เขียนได้ 2 แบบ

Data Bus * สัมพันธ์กับขนาดของข้อมูล , เชื่อมกับอุปกรณ์ทุกชิ้น

printer port ส่งข้อมูลแบบขนาน (เชื่อมกับ printer dotted matrix) * pararel
↳ mouse สมัยก่อน port อนุกรม

① Error Signal
1) Digital ทนสัญญาณรบกวนได้ดี (มีแค่ 0, 1)
2) Parity bit ตรวจสอบค.ผิดพลาด
↳ หลักการ นับเลข 1 แล้วเติม 0 หรือ 1 ที่ bit สุดท้าย
↳ ถ้า check แบบ Even → เติมให้เป็นจน.คู่ ex. 0100 แสดงมีจน.คู่
↳ ถ้า check แบบ Odd → เติมให้เป็นจน.คี่
- check Even เติม 1 → 01001
- check Odd คี่แล้ว → 01000
3) Error correction

ASCII → American Standard Code for Information Interchange

② Address Bus, Data Bus and control signal
1) Data Bus 1) 2 ทิศทาง
2) บอกขนาดข้อมูล
3) ใช้ส่งข้อมูล
- ช่วยปรับปริมาณข้อมูล
- ข้อมูลละเอียดขนาดไหน

ex. แฟลชไดร์ฟ 4 GB เก็บข้อมูลขนาด 8 bit → Data 8 เส้น
4000 ล้าน byte = 2^{32} → Address 32 เส้น
1 byte = 1 char

2) Mem , I/O Addressing

2) Address Bus
1) 1 ทิศทาง
2) บอกพท. เก็บข้อมูล
3) 2^n (Address)
4) บอกตำแหน่งข้อมูล

ข้อมูลกี่ชุด
มีตำแหน่ง
ติดต่ออุปกรณ์ได้อีกจน. เท่าไหร่

3) Control Bus → ส่งสัญญาณควบคุมไปอุปกรณ์ต่าง ๆ
↳ ก.อ่านเขียน
↳ สัญญาณนาฬิกา
↳ เปิด, ปิด

4) Decoder Circuit

③ LED seven segment Common Anode and cathode

3) Mux, Latch, Buffer

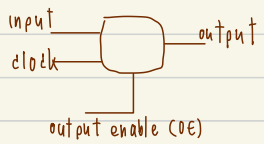
① **Mux, Demux**

② **Latch**

↳ มีก.พัก / เก็บ Data ไว้

↳ ถ้าสัญญาณ clock Active มึงจะเก็บ, พัก

↳ OE ควบคุม on/off ของ output



↳ ข้างใน Latch เป็น SR flip-flop

③ **Buffer**

↳ ไม่เก็บ, ไม่พักข้อมูล

↳ ข้อมูลจะส่งผ่าน, ขยายสัญญาณ

↳ บาง Buffer มี OE

4) Counter and Dac & ADC Part - 1

① Combinational Logic circuit vs Sequential logic circuit

1) Combinational Logic circuit
   1) ไม่มีหน่วยค.จำ * สำคัญ
   2) อาจมีสัญญาณ clock เข้ามาเกี่ยวข้องตอนปิดเปิดบ้าง
2) Sequential Logic circuit
   1) มี mem
   2) มีก.ขดจำ bit / เก็บ state ก่อนหน้า
   3) output ที่ได้อาจเปลี่ยนแปลงตาม state ก่อนหน้า
   4) เกี่ยวกับสัญญาณนาฬิกาเพื่อใช้ในจังหวะปิด-เปิด ก.ทำงานเสมอ

② Rising / Falling edge trigger

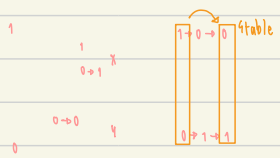
1) Rising
   ↳ ก.ทำงานที่ขอบขาขึ้น
   1) จังหวะที่ Logic เปลี่ยนจาก 0 → 1 = Active
2) Falling
   ↳ ก.ทำงานที่ขอบขาลง
   1) จังหวะที่ Logic เปลี่ยนจาก 1 → 0 = Active

* ใช้สะดุ้น ก.ทำงานของวงจร

ชุดสัญญาณ 1) Rising edge ไม่มีวงกลม (Positive)
2) Falling " มี (Negative)

1) check Rising
   1) ต้องการให้เกิดภายหน่วยเวลา

③ SR Flip - Flop



B		NOR
0	0	1
0	1	0
1	0	0
1	1	0

4) Counter and Dac & ADC Part - 1

④ JK Flip-flop

↳ Jack Killby

↳ พัฒนาจาก SR Flip-flop

↳ เพิ่มขา clock มา

↳ ใช้ในวงจรหารความถี่ → Toggle ช่วยให้ค่าที่สลับ

↳ * ใช้ทำนาฬิกาได้ด้วย

* No change = 0 0 , No clk

* Toggle 1 → 0

0 → 1

สภาวะสลับไปมา

⑤ D Flip-flop and T Flip-flop

1) D from JK

1) ไม่สนใจสภาวะ Toggle * สน Set / Reset (ในที่นี้เรียกว่า `Store`)

เพิ่ม Not ได้

2) เก็บค่า

↳ ถ้า เรียงกัน 8 ตัว = 8 bit = 1 char (ASCII code)

2) T from D and JK

1) โฟกัสที่ Toggle

2) เรียก clk → T

⑥ 4-bit latch circuit

⑦ 4-bit FIFO

1) ใช้กับข้อมูลอนุกรม (Serial data)

5) Counter and Dac & ADC Part - II

↳ คนได้ยิน 20 Hz - 20 kHz

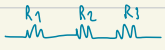
① DAC (Digital → Analog)

↳ ex. 4 bit, 12 V } แบ่งสมมติ → 2^4 = 16 Levels (0-15) (D_0 D_1 D_2 D_3)

12 V 16 Level → $\frac{12}{16}$ = Level ล : 0.8

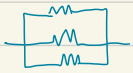
{ ถ้าม Level ที่ 1 มีกี่ Volt
เอา 0.8 × 1 = 7.2 V
Level ที่ 0 = 0 V

↳ ก.ต่อ R → อนุกรม



$\Sigma R = R_1 + R_2 + R_3$

→ ขนาน



$\Sigma R = \frac{1}{\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}}$

↳ คำนวณ R2R

$$V_{out} = -V_{ref}\left[b_3\frac{1}{2} + b_2\frac{1}{4} + b_1\frac{1}{8} + b_0\frac{1}{16}\right]$$

✱ เวลาคำนวณเช็คแบบทุก bit เป็น 1 ก่อน
มันควรเป็น x Volt ตามโจทย์ ถ้าไม่ตรงอย่าลืม ± ส่วนต่างตอนคำนวณเสร็จ

↳ Opamp
- ↳ Operational amplifiers
- ↳ ตัวต้านทานตรง input สูงมาก = ไม่มีกระแสไหลผ่าน
- ↳ มี input 2 อัน { ↳ ⊕ ↳ ⊖ } ขยายสัญญาณโดยเอา 2 ตัวนี้มา ⊖ กัน

② ADC

1) Flash ADC
- ↳ เอาตัวต้านทานมาต่อขนุกรมเพื่อสร้างแรงดันอ้างอิง
- ↳ ใช้ opamp เปรียบเทียบแรงดัน
- ↳ ก.เปรียบเทียบ ถ้า ⊕ มากกว่า = 1
  ⊖ มากกว่า = 0

2) Successive Approximation Register

6) Memory Unit