Chapter 3 Exercise - Instruction Set Architecture (ISA)

1. สถาปัตยกรรมชุดคำสั่งคืออะไร มีสิ่งใดที่ต้องกล่าวถึงบ้าง และเป็นประโยชน์ในการพัฒนาซอฟต์แวร์ และ ฮาร์ดแวร์หรือไม่ อย่างไร
   - คือ สถาปัตยกรรม(Architecture)ของภาษาเครื่อง(Machine language/code)ที่ใช้สำหรับหน่วยประมวลผลกลาง(CPU)
   - สิ่งที่ต้องกล่าวถึง
     - Register Transfer Language (RTL)
     - Register and Flag
     - Memory Organization
     - Instruction format
     - Interrupt and Exception
   - เป็นมาตรฐานที่ใช้สื่อระหว่างการพัฒนาฮาร์ดแวร์และซอฟต์แวร์

2. opcode และ operand คืออะไร
   - opcode คือ รหัสคำสั่ง
   - operand คือ ข้อมูลสำหรับการประมวลผล

3. หากระบบคอมพิวเตอร์มีการอ้างอิงหน่วยความจำโดยใช้เลขที่(Address)อยู่ทั้งสิ้น 32 บิต จะสามารถอ้างอิงข้อมูลแบบ word address ได้ทั้งสิ้นกี่ word

   Answer: 2 word, because 8 bits = 1 byte, 1 word = 2 bytes = 16 bits

4. ระบบ big endian และ little endian มีข้อแตกต่างกันอย่างไร และแต่ละแบบมีข้อดีข้อเสียอย่างไร

Big endian
   - ข้อมูลที่มี นัยสำคัญ(Significance)ต่ำ จะอยู่ที่ เลขที่อยู่(Address) สูง, S-Low, Addr-High
   - การอ่านเลขที่ต้องมีการ เปรียบเทียบ ค่าทำได้รวดเร็ว เพราะเห็นเครื่องหมาย หรือนัยสำคัญสูงก่อน

Little endian
   - ข้อมูลที่มี นัยสำคัญ(Significance)ต่ำ จะอยู่ที่ เลขที่อยู่(Address) ต่ำ, S-Low, Addr-Low
   - การประมวลผลบางแบบ (เช่นการบวกลบทางคณิตศาสตร์) ทำได้รวดเร็ว เพราะ สามารถคำนวณเลขจากตำแหน่งที่มีนัยสำคัญต่ำไปก่อนที่จะอ่านข้อมูลได้ครบก่อน

5. การที่หน่วยประมวลผลมีระบบการอ้างอิงหน่วยความจำ(Addressing Mode)หลากหลายรูปแบบ เป็นประโยชน์หรือไม่ อย่างไร
   - เพิ่มความยืดหยุ่นในการเขียนโปรแกรม
   - ประสิทธิภาพในการประมวลผล
   - ลดขนาดของ code

- ความยืดหยุ่นในการจัดการข้อมูล

6. หากสถาปัตยกรรมชุดคำสั่งที่กำหนดให้ มีเพียงระบบการอ้างอิงหน่วยความจำแบบ register indirect และ immediate เท่านั้น เราจะสามารถใช้งานสถาปัตยกรรมชุดคำสั่งดังกล่าวในการพัฒนาโปรแกรมทั่วไปได้หรือไม่ อย่างไร จงอธิบายพร้อมให้เหตุผลประกอบ

ได้ เพราะ หากทำการคำนวณค่าของเลขที่อยู่ที่ต้องการใส่ในเรจิสเตอร์ก่อน จะสามารถใช้การอ้างอิงเลขที่อยู่แบบ register indirect แทนได้ทุกแบบ แม้การมีวิธีการอ้างอิงเลขที่อยู่เพียงไม่กี่แบบ แล้วต้องอาศัยคอมไพเลอร์ และ การคำนวณที่มากขึ้น และผลที่ได้ในทางหนึ่งคือ คำสั่งมีความซับซ้อนน้อยลง และสามารถประมวลผลแบบขนานได้มากขึ้น

7. การที่ instruction format ทุกรูปแบบมีความยาวคำสั่งเท่ากันหมด มีข้อดีข้อเสียอย่างไร

สามารถอ่านคำสั่งได้ภายใน 1 รอบการประมวลผล ทำให้อ่านแต่ละคำสั่งในเวลาคงที่

8. ภาษาแอสเซมบลีและสถาปัตยกรรมชุดคำสั่งมีข้อเหมือนหรือข้อแตกต่างกันหรือไม่อย่างไร

ข้อเหมือน
- ภาษาแอสเซมบลีและสถาปัตยกรรมชุดคำสั่ง มีความสัมพันธ์กันโดยตรง มีความสอดคล้องกันอย่างชัดเจน
- การควบคุมฮาร์ดแวร์ได้โดยตรง

ข้อแตกต่าง
- ความหมายและบทบาท
  - ISA: ข้อกำหนดหรือชุดคำสั่ง
  - Assembly: programming language
- ระดับการทำงาน
  - ISA: ระดับของการออกแบบ hardware
  - Assembly: ระดับของผู้พัฒนาโปรแกรม
- การเปลี่ยนแปลงและความยืดหยุ่น
  - ISA: เปลี่ยนแปลงได้ยากและต้องการการออกแบบใหม่ในระดับฮาร์ดแวร์ หากมีการปรับปรุงหรือเปลี่ยนแปลง
  - ภาษาแอสเซมบลี: สามารถเขียนหรือปรับแต่งได้โดยผู้พัฒนาโปรแกรม และมีการแปลงรหัสเพื่อให้เข้ากับสถาปัตยกรรมที่กำหนดโดย ISA

9. จากมุมมองของคอมไพเลอร์ สถาปัตยกรรมชุดคำสั่งที่ดี ควรมีคุณสมบัติอย่างไร

1. Compatible
2. Generality / Completeness
3. Orthogonal
4. Ease of Compilation
5. Ease of Implementation

10. จงแสดงการจัดเรียงข้อมูลตามลำดับการจองตัวแปรดังกล่าวในภาษาซี ลงในหน่วยความจำที่มีการบังคับ restricted alignment
(กำหนดให้ char มีขนาด 1 ไบต์short มีขนาด 2 ไบต์ int, long และ foat มีขนาด 4 ไบต์ long long และdouble มีขนาด 8 ไบต์)

```
char a;   // 1 byte
short b;  // 2 bytes
int c;    // 4 bytes
char d;   // 1 byte
long e;   // 4 bytes
double f; // 8 bytes
```

Answer:

Address (byte)	Data Type	Variable
0	char	a
1	padding	-
2	short	b
4	int	c
8	char	d
9, 10, 11	padding	-
12	long	e
16	double	f

11. เหตุใดในสถาปัตยกรรมชุดคำสั่งสมัยใหม่ จึงไม่นิยมทำเป็นแบบ accumulator จงให้เหตุผลประกอบการอธิบาย

เป็นสถาปัตยกรรมในยุคแรกที่ทรานซิสเตอร์ยังมีราคาแพง ทำให้การสร้างหน่วยประมวลผลจำเป็นต้องออกแบบให้มีหน่วยความจำภายในเฉพาะเท่าที่จำเป็น ทำให้ accumulator เป็นคอขวดของการประมวลผล

12. ในการเรียกใช้งานโปรแกรมย่อย จงเปรียบเทียบข้อดีข้อเสียระหว่างการจำค่าโดยผู้เรียก(Caller Save) และ การจำค่าโดยผู้ถูกเรียก(Callee Save) กรณีใดน่าจะทำงานได้อย่างมีสมรรถนะสูงกว่า จงอธิบายพร้อมยกตัวอย่างประกอบ

กรณีผู้ถูกเรียกเป็นผู้บันทึกค่า(Callee Save) สมรรถนะที่ได้จะดี เพราะ ผู้ถูกเรียก(Callee)จะเลือกบันทึกเพียงค่าเรจิสเตอร์ที่ตนเองใช้ และเว้นค่าอื่นที่ไม่เกี่ยวข้องไว้

13. จากตัวอย่างโปรแกรมที่กำหนดให้ ให้ลองแปลด้วย “gcc -C -S test.c” แล้วลองศึกษาดูว่า โปรแกรมดังกล่าว มีการส่งผ่านค่า parameters และ การคืนค่าอย่างไร (หากเป็นไปได้ให้ลองทำในระบบปฏิบัติการและหน่วยประมวลผลกลางที่ใช้สถาปัตยกรรมชุดคำสั่งแตกต่าง

กันไป เพื่อจะได้เห็นความหลากหลาย)
```c
int min(int a, int b) {
    int minv;
    minv=a;
    if (min>b) {
        minv=b;
    }
    return minv;
}

void test() {
    int x=min(3,2);
}
```

14. หากสถาปัตยกรรมชุดคำสั่งเลือกใช้เรจิสเตอร์ (เช่น $r13) ในการบันทึกค่า returnaddress โปรแกรมย่อยจะต้องมีการจัดการอย่างไร เพื่อให้สามารถเรียกใช้งานโปรแกรมย่อยอื่นหรือทำ recursive call ได้อีก จงอธิบายพร้อมยกตัวอย่างประกอบ

    ใช้ Stack เพื่อจัดการกับ return address
```assembly
foo:
    ; recursive_function
    push $r13 ; บันทึกค่า 'return address' ลง stack ที่ '$r13'

    ; ถ้าเงื่อนไขหยุดการ recursive call ไม่เป็นจริง
    cmp $0, $r0         ; เปรียบเทียบค่า r0 กับ 0
    je base_case        ; jump if equal ถ้าเป็นจริงไปยัง
base_case/terminate_case

    ; perform recursive
    call foo

base_case:
    ; terminate_case
    ; ทำงานใน base case ที่นี่

    ; กู้คืนค่า return address
    pop $r13
    ; คืนค่ากลับไปยัง caller
    ret
```

15. จงอธิบายข้อแตกต่างระหว่าง vector table และ handler table พร้อมยกตัวอย่างประกอบการอธิบาย (โดยเฉพาะกรณีที่ handler ไม่มีการทำงานมีแต่เพียงการทำ IRET)

vector table: ตารางเก็บที่อยู่
handler table: ตารางเก็บ code

กรณีที่ handler ไม่มีการทำงานมีแต่เพียงการทำ IRET(การจบ exception) ในกรณีนี้
หากเกิดข้อขัดจังหวะ (เช่น interrupt 0) ระบบจะไปที่ Vector Table เพื่อหาที่อยู่ของ handler
Handler ที่อยู่ใน Handler Table อาจมีเพียงคำสั่ง IRET เท่านั้น ซึ่งหมายความว่าเมื่อ handler ถูกเรียกใช้งาน จะไม่มีการทำงานใด ๆ นอกจากการคืนค่ากลับไปยังสถานะที่ถูกขัดจังหวะไว้
นี่จะทำให้ CPU สามารถกลับไปดำเนินการต่อได้ทันที โดยไม่มีการดำเนินการเพิ่มเติมในกรณีที่ต้องการให้ข้อขัดจังหวะนั้นเพียงแค่คืนค่าเท่านั้น