

1. Explain the internal structures of a computer using IAS computer (in Figure 1.6)

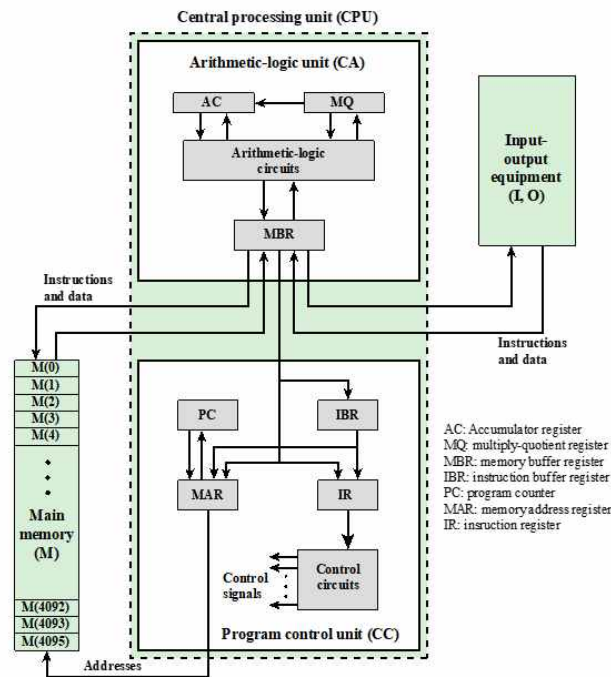


Figure 1.6 IAS Structure

✓ CPU (Central Processing unit) :

중앙처리장치라 불리며 컴퓨터의 동작을 제어하고 데이터 처리 기능들을 수행한다. 특히 하나의 칩으로 된 중앙처리장치를 마이크로프로세서(Microprocessor)라 한다.

- AC (Accumulator unit : 40bits) & MQ (Multiply-Quotient Register : 40bits) :
계산에 사용될 데이터와 ALU의 계산 결과를 일시적으로 저장한다. 예를 들면, 두 개의 40-비트 수를 곱하면 80-비트 수가 되는데, 그들중의 상위 40비트는 누산기에 저장되고 하위 40비트는 MQ에 저장된다.
- MBR (Memory Buffer Register) :
기억장치에 저장될 단어를 가지고 있거나, 기억장치로부터 읽혀질 단어를 저장하는 데 사용된다.
- PC (Program counter) :
다음에 실행될 명령어가 저장된 주기억장치의 주소를 저장하는 레지스터
- IBR (Instruction Buffer Register) :
기억장치로부터 읽혀진 단어의 우측에 위치한 명령어를 일시적으로 저장하는 데 사용된다.
- MAR (Memory Address Register) :
MBR쪽으로 읽어들이 메모리 주소를 저장하는 레지스터
- IR (Instruction Register) :
실행될 명령어의 8-비트 연산 코드를 저장한다.

✓ Main memory :

주기억장치를 뜻하며 실행중인 프로그램과 실행에 필요한 데이터를 저장하는 장치이다.

✓ I/O (Input Output equipment) :

입력장치는 문자, 그림, 소리, 동영상 등과 같은 외부의 데이터를 컴퓨터로 읽어들이는 일을 한다. 대표적인 입력장치로 키보드, 마우스, 스캐너, 조이스틱 등이 있다.

2. Indicate the width (in bits) of each connection (data/address path) between components (registers, AL circuits, control circuits, memory, I/O,...) in IAS Computer. (in Figure 1.6)

IAS의 기억장치는 4096개의 저장 장소들로 구성되며, 각 저장 장소에는 단어(word)라고 불리는 숫자들이 각각 40bits로 저장된다. (left instruction 20bits + right instruction 20bits) 각 숫자는 한 개의 부호 비트와 39bits 데이터로 표현된다. 하나의 명령어는 8bits의 opcode와 12bits의 address로 구성되므로 먼저 실행되는 데이터는 IR에 opcode를 저장해야 하므로 폭은 8bits 뒤에 실행되는 데이터는 MAR에 address를 저장해야 하므로 폭이 12bits이다.

3. Consider a 32bit microprocessor with a 16 bit external data bus, driven by 4GHz clock. It's bus cycle equals four input clock. What is the maximum data transfer rate(bytes/sec) across the bus? To increase its performance, would it be better to make its external data bus 32 bits or to double the input clock frequency? Explain your answer.

$$Clock = \frac{1}{f} = \frac{1}{4GHz} = 250ps$$

4. Suppose that 30% of the computing time is consumed by arithmetic operations in a system. If the arithmetic module of a system is speedup by a factor of 3, what is the maximum speedup of this system?

< Amdahl's law >

어떤 시스템을 개선하여 전체 작업 중 f%부분에서 n배의 성능이 향상되었을 때 전체 시스템에서의 최대 성능 향상

$$\rightarrow T = \frac{1}{(1-f) + \frac{f}{n}}$$

(* 여기서 $(1-f)$ 는 프로그램을 수행하는데 무조건 걸리는 시간, 즉 개선되지 않는 시간이다)

(* f의 증가가 n의 증가보다 큰 영향을 미친다)

Sol) $f = 0.3, n = 3$

$$\gg T = \frac{1}{(1-0.3) + \frac{0.3}{3}} = 1.25$$

☞ 결과에 따라 최대 성능은 1.25배 향상되었다

5. Generally, a computer operates by repetitively performing an instruction cycle, which consists of two sub cycles: a fetch cycle and an execute cycle. Describe in English the tasks accomplished during the fetch cycle and those accomplished during the execute cycle.

1 - 1) Fetch :

- ①. The CPU sends the contents of the PC to the MAR and sends a read command on the address bus.
- ②. In response to the read command (with address equal to PC), the memory returns the data stored at the memory location indicated by PC on the data bus.
- ③. The CPU copies the data from the data bus into its MBR.
- ④. A fraction of a second later, the CPU copies the data from the MBR to the IR for instruction decoding.
- ⑤. The PC is incremented so that it points to the next instruction. This step prepares the CPU for the next cycle.

1 - 2) Decode :

The decoding process allows the CPU to determine what instruction is to be performed so that the CPU can tell how many operands it needs to fetch in order to perform the instruction. The opcode fetched from the memory is decoded for the next steps and moved to the appropriate registers. The decoding is done by the CPU's Control Unit.

2) Execute :

The function of the instruction is performed. If the instruction involves arithmetic or logic, the ALU is utilized. This is the only stage of the instruction cycle that is useful from the perspective of the end user. Everything else is overhead required to make the execute step happen.

※ IF(Instruction Fetch) > ID(Instruction Decode) > OF(Operand Fetch)
> EX(Execute) > OS(Operand Store) > IC(Interrupt Check)

6. Explain the purpose of I/O interrupt. How the instruction cycle in the problem 5 is modified to accommodate interrupts?

입출력장치가 CPU보다 느리므로 발생함. 입출력 준비가 완료되면 CPU가 Execute Cycle, 즉 Instruction Cycle이 모두 끝나고 Interrupt Check 과정에서 핸들러에 의해 Interrupt 수행 후 다시 프로그램을 수행한다. Instruction Cycle이 끝나고 Interrupt를 수행하는 이유는 현재 명령어를 모두 수행하였기 때문에 따로 데이터를 저장할 필요가 없기 때문이다.

7. Explain the formula of $CPI = p+k \times m$ using the instruction cycle.

$$T_{exe.time} = I_c \times CPI \times \tau (= \frac{1}{f}) = I_c \times (p + k \times m) \times \frac{1}{f}$$
$$(T_{exe.time} = \text{sec}, I_c = \text{instr}, CPI = \text{cycles/instr} * \text{sec/cycles})$$

여기서 p 는 CPU가 처리하는데 걸리는 시간, k 는 메모리를 가져가는 시간, m 은 메모리에 가는 횟수를 뜻하며 결국 $(p + k \times m)$ 은 명령어를 실행하는 동안에 그 동작의 일부는 프로세서에 의해 이루어지며, 일부 시간은 단어가 기억장치로 전송되는 데 소모된다는 것을 고려하는 명령주기를 나타낸다.(전송에 걸리는 시간이 기억장치 사이클 시간에 따라 결정된다) 결국 요구되는 시간이 적으려면(명령주기가 짧으려면) 즉 속도가 빠르기 위해선 CPI의 값이 작아야 한다.

이 결과 소요시간을 줄이기 위해 나타난 것이 CISC, RISC 이다.

(* CPI (Cycle Per Instruction) : CPU가 명령을 실행하는데 필요로 하는 시스템 클록의 사이클 수)

8. Compare a hardwired program in the customized hardware and a software program in the general purpose hardware.

● Hardwired program : 논리 회로대로 설계하여 회로도가 복잡하고 구현하는 비용이 많이 요구되며 제어 메모리를 필요로 하지 않는다. 속도가 빠르지만 명령어를 교체하기 위해서는 하드웨어를 교체해야 한다. 여러 부품들을 모아서 필요한 기능을 수행하도록 연결하는 과정을 프로그래밍의 한 형식으로 볼 수도 있고 이 결과로 나타나는 프로그램은 하드웨어 형태를 가지며, 이것을 hardwired program이라고 부른다. > 'Special Purpose'

● Software program : 설계가 간결하고 하드웨어가 최소화 되어있다. ROM내의 프로그램에 의존하므로 구조적이다. 개발하는 중에 수정이 쉽고, 개발 후 기능 변경을 하는 방법도 간단하다. 새로운 프로그램마다 별도의 하드웨어를 재구성할 필요 없이 새로운 제어 신호들을 제공해주기만 하면 된다. > 'General Purpose'

9. Explain why contemporary systems rely on point-to-point interconnection rather than shared bus.

데이터율이 더 높아질수록, 시간 방식으로는 동기화와 중재 기능을 수행하는 것이 점점 더 어려워지고 멀티코어가 출현하고, 다수의 프로세서들과 큰 기억장치가 하나의 칩에 들어가게 되면서, 같은 칩에 전통적인 공유버스를 사용한다면 프로세서와 보조를 맞추기 위하여 버스 데이터율을 증가시키거나 버스 지연을 줄이는 데 어려움이 커지는 것으로 밝혀졌다. 공유 버스와 비교하면, 점대점 상호연결은 지연이 적고, 데이터율이 높으며, 선형성도 더 좋다.

10. A benchmark program is run on a 1GHz, and then on a 2GHz processor.
The executed program has the following instruction mix.

Instruction Type	Instruction count	CPI
Integer arithmetic	10 million	1
Data transfer	6 million	2
Floating point	3 million	2
Control transfer	1 million	2

Determine the effective CPI, program execution time and MIPS rate for both case.

각각의 Instruction Mix(%)는
Integer arithmetic 50
Data transfer 30
Floating point 15
Control transfer 5
이다.

$$CPI = \frac{\sum_{i=1}^n (CPI \times I_i)}{I_c}$$

따라서 $CPI = 1 \times 0.5 + 2 \times 0.3 + 2 \times 0.15 + 2 \times 0.05 = 0.5 + 0.6 + 0.3 + 0.1 = 1.5$ 가 된다.

따라서 MIPS와 Execution time은

$$MIPS = \frac{f}{CPI \times 10^6}$$

$$Executiontime(T) = I_c(Instruction.count) \times CPI \times \tau (= \frac{1}{f})(clocktime)$$

① 1GHz Processor

$$MIPS = \frac{1 \times 10^9}{1.5 \times 10^6} \approx 666.67 MIPS$$

$$Executiontime = \frac{1.5 \times 2 \times 10^7}{1 \times 10^9} = 0.03 \text{sec} = 30ms$$

② 2GHz Processor

$$MIPS = \frac{2 \times 10^9}{1.5 \times 10^6} \approx 1333.33 MIPS$$

$$Executiontime = \frac{1.5 \times 2 \times 10^7}{2 \times 10^9} = 0.015 \text{sec} = 15ms$$