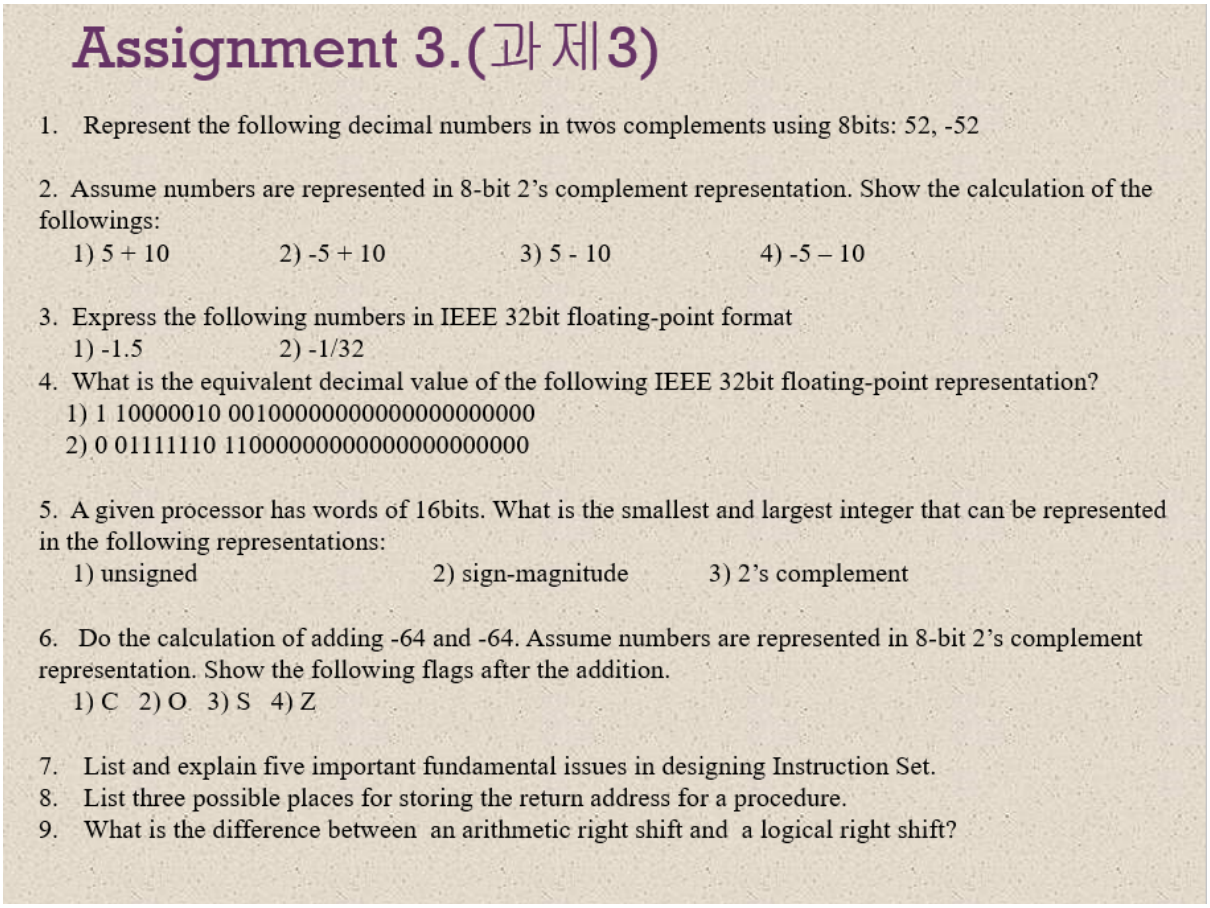
Report #3

32202546 안지성



Assignment 3.(과제3)

1. Represent the following decimal numbers in twos complements using 8bits: 52, -52

2. Assume numbers are represented in 8-bit 2's complement representation. Show the calculation of the followings:
   1) 5 + 10 2) -5 + 10 3) 5 - 10 4) -5 – 10

3. Express the following numbers in IEEE 32bit floating-point format
   1) -1.5 2) -1/32
4. What is the equivalent decimal value of the following IEEE 32bit floating-point representation?
   1) 1 10000010 00100000000000000000000
   2) 0 01111110 11000000000000000000000

5. A given processor has words of 16bits. What is the smallest and largest integer that can be represented in the following representations:
   1) unsigned 2) sign-magnitude 3) 2's complement

6. Do the calculation of adding -64 and -64. Assume numbers are represented in 8-bit 2's complement representation. Show the following flags after the addition.
   1) C 2) O 3) S 4) Z

7. List and explain five important fundamental issues in designing Instruction Set.
8. List three possible places for storing the return address for a procedure.
9. What is the difference between an arithmetic right shift and a logical right shift?

1. 52를 8bit형식으로 2진수로 나타내면 00110100이다.
   -52를 2의 보수로 구하기 위해서는 1의 보수를 먼저 구해야 하는데 1의보수는 00110100이고 2의 보수는 1의 보수에 1을 더하면 되므로 11001100이다.

2. 5는 2진수를 8bit형식으로 표현하면 00000101이고 10은 00001010이다.
   -5는 00000101에 2의 보수를 취해서 나타내면 11111011이고, -10도 00001010에 2의 보수를 취해서 나타내면 11110110이다.
   1) 00000101과 00001010을 더하면 00001111 즉, 15가 나온다.
   2) 11111011과 00001010을 더하면 00000101 즉, 5가 나온다.
   3) 00000101과 11110110을 더하면 11111011이고 음수인데 2의 보수를 취해서 양수로는 00000101이 나오므로 -5라는걸 알 수 있다.
   4) 11111011과 11110110을 더하면 11110001이고 음수인데 2의 보수를 취해서 양수로는 00001111이 나오므로 -15라는걸 알 수 있다.

3. 1) -1.5를 2진수로 표현할 때, -1은 부호빼고 1이므로 가만히 두고 0.5는 2를 곱하면 1.0이 나오므로 즉, -1.5는 2진수로 -1.1이다.
   -1.5는 -1.1*2^0으로 표현 가능하고 이는 부호비트-> 1, 지수부 비트-> 01111111(127), 가수부 비트-> 10000000000000000000000 이므로
   1 01111111 10000000000000000000000로 총 32bit이다.

   2) -1/32를 2진수로 표현할 때, -1/32=-0.03125이고, 이를 2진수로 표현하면 -0.00001이 나오므로 즉, -1.0*2^(-5)이다.*
   -1.02^(-5)는 부호비트-> 1, 지수부 비트-> 01111010(127-5=122), 가수부 비트-> 00000000000000000000000 이므로
   1 01111010 00000000000000000000000로 총 32bit이다.

4. 1) 1 10000010 00100000000000000000000는 부호-> (-), 지수부-> 130-127=3, 가수부-> 001이므로 즉, -1.001*2^3이다.
   2) 0 01111110 11000000000000000000000는 부호-> (+), 지수부-> 126-127=-1, 가수부-> 11이므로 즉, 1.11*2^-1이다.

5. 1) 16비트이고 unsigned는 부호가 없으므로 16bit를 모두 데이터 bit로 사용하여 최솟값: 0, 최댓값: 2^16-1 이다.

   2) sign-magnitude는 16비트에서 부호비트가 1bit 쓰이고 나머지 15bit가 데이터bit로 쓰이는데 0이 양수와 음수 모두 있으므로 최솟값: -2^15+1, 최댓값: 2^15-1 이다.

   3) 2's complement는 16비트에서 부호비트가 1bit 쓰이고 나머지 15bit가 데이터bit로 쓰이는데 0이 양수에만 포함되므로 최솟값: -2^15, 최댓값: 2^15-1 이다.

6. 8bit로 64는 01000000이고 2의 보수로 -64를 표현하면 11000000이다. (-64) + (-64)를 구하면 10000000이 나오는데 이렇게 구하기 직전에 1|10000000으로 Carry가 1이 있었지만 이는 답에는 영향주지 않으므로 떼어냈다.

   1) C(Carry)는 C=1이다.

   2) O(Overflow)는 8bit인데 (-64)+(-64)= -128로 8bit를 넘기지 않았으므로 O=0이다.

   3) S(Sign)는 10000000으로 부호비트가 1(음수)이므로 S=1이다.

   4) Z(Zero)는 결과가 10000000 즉, -128이므로 0이 아니라 Z=0이다.

7. Instruction Set(명령어 세트)는 CPU를 위해 정의되어 있는 명령어들의 집합이다.
   명령어 세트의 설계를 위한 중요한 근본적인 이슈들은 다음과 같이 5가지가 있다.

   1) Operation repertoire: CPU가 수행할 연산들의 수와 종류 및 복잡도이다.

   2) Data type: 연산을 수행할 데이터들의 형태, 데이터의 길이(비트 수), 수의 표현 방식 등이 있다.

   3) Instruction format: 명령어의 길이, 오퍼랜드 필드들의 수와 길이 등이 있다.

4) Registers: 명령어 처리에 필요한 레지스터의 개수를 표현한다.

5) Addressing Mode: 오퍼랜드의 주소를 지정하는 방식이다.

8. procedure의 주소를 반환하여 저장하는 장소 3가지는 Return Pointer, Stack Pointer, Frame Pointer이다.
   Return Pointer는 함수를 호출하면 돌아올 값의 주소를 저장한다.
   Stack Pointer는 데이터를 삽입해야하는 위치를 의미하는 스택의 최상위 요소를 나타낸다.
   Frame Pointer는 함수들의 프레임을 구별하기 위해서 사용한다.

9. logical right shift는 부호비트가 보존되지 않고, 모든 비트를 우측으로 한칸씩 이동하며 최상위 비트는 무조건 0으로 채우고 최하위 비트는 버린다.
   반면 arithmetic right shift는 부호비트인 최상위비트를 유지하는 기술이므로 2진수 연산에서 보수로 표현된 음수의 경우는 모든비트를 우측으로 한칸씩 이동하고
   최상위 비트에 부호를 유지하기 위해 1을 채운다. 그리고 양수인 경우는 모든비트를 우측으로 한칸씩 이동하고 최상위 비트에 양수이므로 0을 채워준다.
   즉, 오른쪽 논리와 산술 쉬프트의 차이점은 부호비트를 유지하는지 안하는지에 있다.
   논리는 부호비트가 보존되지 않고, 산술은 보존한다.