

주제/단락	내용
지수 조정 단계	부동소수점 연산의 첫 단계로, 두 수의 지수가 일치하도록 조정하며 더 큰 수를 기준으로 작은 수를 오른쪽으로 시프트한다.
가수 연산 단계	지수 조정 후, 가수들 간의 덧셈 혹은 뺄셈 연산을 수행하는 단계이다.
정규화 단계	가수 연산이 끝난 결과를 표준 형식에 맞게 변환하는 과정이다.
10진수 부동소수점 산술 예시	$135 \times 10^{-5}$ 과 $246 \times 10^{-3}$ 의 덧셈을 예로 들 수 있다.
부동소수점 덧셈 예제	$(0.110100 \times 2^3) + (0.111100 \times 2^5)$ 의 덧셈 연산 예제이다.
부동소수점 덧셈 풀이	지수 조정( $0.001101 \times 2^5$ ), 더하기( $1.001001 \times 2^5$ ), 정규화( $0.1001001 \times 2^6$ ) 순서로 진행된다.
부동소수점 산술 파이프라이닝	연산 과정을 독립적인 단계로 분리하여 처리 속도를 높이는 기법이다.
파이프라이닝 속도 향상	파이프라인의 단계 수만큼 연산 속도가 향상되는 효과가 있다.
파이프라이닝 적용 분야	대규모 부동소수점 계산을 처리하는 대부분의 슈퍼컴퓨터에서 채택하여 사용한다.
수 배열 덧셈 예시	파이프라이닝의 예로, 배열 C의 각 원소는 배열 A와 B의 해당 원소의 합, 즉 $C(I)=A(I)+B(I)$ 로 계산된다.
부동소수점 연산 오차 예시	프로그래밍 언어에서 $1.2 + 0.3$ 이 정확히 1.5가 되지 않는 문제이다.
연산 오차 원인	IEEE 754 표준에서 유한한 비트로 수를 표현하므로 1.2와 0.3을 완벽하게 나타내지 못해 발생하는 정밀도 한계 때문이다.
1.2 이진 표현	1.2는 이진수로 약 $1.199999809265136710$ 으로 표현된다.
0.3 이진 표현	0.3은 이진수로 약 $0.299999713897705110$ 으로 표현된다.
1.2+0.3 이진 표현 결과	1.2와 0.3의 이진 표현 덧셈 결과는 약 $1.499999523162841810$ 이다.
1.5 이진 표현	1.5는 이진수로 정확히 1.1로 표현된다.
오차 해결 방안	엡실론(epsilon)과 같은 작은 오차 허용 범위를 설정하여 $(a + b - 1.5) < \text{epsilon}$ 과 같은 조건으로 비교한다.
2진수 부동소수점 곱셈	가수 곱셈, 지수 덧셈, 결과 정규화의 세 과정으로 이루어진다.
곱셈의 가수 처리	두 피연산자의 가수를 정수 곱셈 방식으로 곱한다.
곱셈의 지수 처리	두 지수를 더하며, 바이어스된 지수 사용 시 바이어스 하나를 제거한다.
곱셈의 정규화	가수 곱셈과 지수 덧셈 후 결과를 표준 형식으로 맞춘다.

주제/단락	내용
2진수 부동소수점 나눗셈	가수 나눗셈, 지수 뺄셈, 결과 정규화의 세 과정으로 이루어진다.
나눗셈의 가수 처리	피제수의 가수를 제수의 가수로 정수 나눗셈 방식으로 나눈다.
나눗셈의 지수 처리	피제수의 지수에서 제수의 지수를 빼며, 바이어스된 지수 사용 시 바이어스 하나를 추가한다.
나눗셈의 정규화	가수 나눗셈과 지수 뺄셈 후 결과를 표준 형식으로 맞춘다.
부동소수점 곱셈 예제	$(0.1011 \times 2^3) \times (0.1001 \times 2^5)$ 의 곱셈 연산 예제이다.
부동소수점 곱셈 풀이	가수 곱하기( $1011 \times 1001$ ), 지수 더하기( $3+5$ ), 정규화( $0.1100011 \times 2^7$ ) 순서로 진행된다.
지수 오버플로우	양의 지수 값이 표현 가능한 최대 지수 값을 초과하는 상태로, $+\infty$ 또는 $-\infty$ 로 처리된다.
지수 언더플로우	음의 지수 값이 표현 가능한 최대 음수 지수 값을 초과하는 상태로, 0으로 처리된다.
가수 언더플로우	정규화 과정에서 가수의 유효 비트가 오른쪽으로 벗어나는 상태로, 반올림이나 절삭으로 처리한다.
가수 오버플로우	같은 부호의 두 가수를 더할 때 올림수가 발생하는 상태로, 오른쪽 시프트와 지수 증가를 통해 재정규화한다.
2진수 반올림 규칙 1	반올림 할 위치의 다음 비트가 0이면 버린다. (예: $0.101 \rightarrow 0.1$ )
2진수 반올림 규칙 2	반올림 할 위치의 다음 비트가 1이고 그 뒤에 1이 하나라도 있으면 올린다. (예: $0.11001 \rightarrow 1.0$ )
2진수 반올림 규칙 3	반올림 할 위치의 다음 비트가 1이고 그 뒤가 모두 0이면, 앞자리가 짝수가 되도록 올리거나 버린다. (Ties-to-even)
Ties-to-even 예시 (버림)	0.01000은 반올림할 자리 숫자가 0(짝수)이므로 버려서 0.0이 된다.
Ties-to-even 예시 (올림)	0.11000은 반올림할 자리 숫자가 1(홀수)이므로 올려서 1.0이 된다.