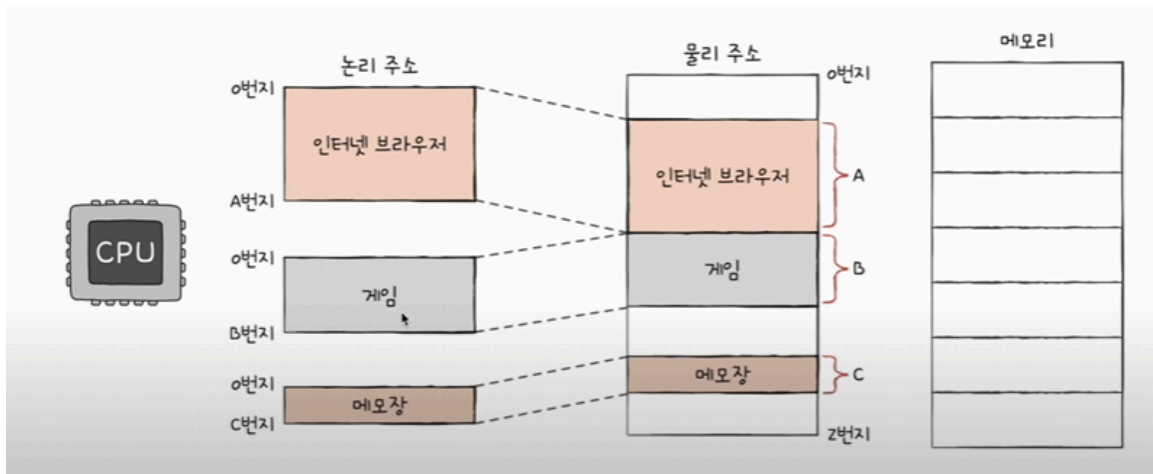


16강. 메모리의 주소공간 - 물리 주소와 논리 주소

물리 주소와 논리 주소



프로그램이 실행될 때, 메모리에는 새로운 데이터가 적재되고 실행이 끝난 프로그램은 삭제됩니다. 같은 프로그램이라도 실행할 때마다 적재되는 위치는 달라질 수 있습니다. 그렇기 때문에 프로그램 입장에서 바라보는 주소와 메모리 하드웨어 입장에서 바라보는 주소는 서로 다릅니다.

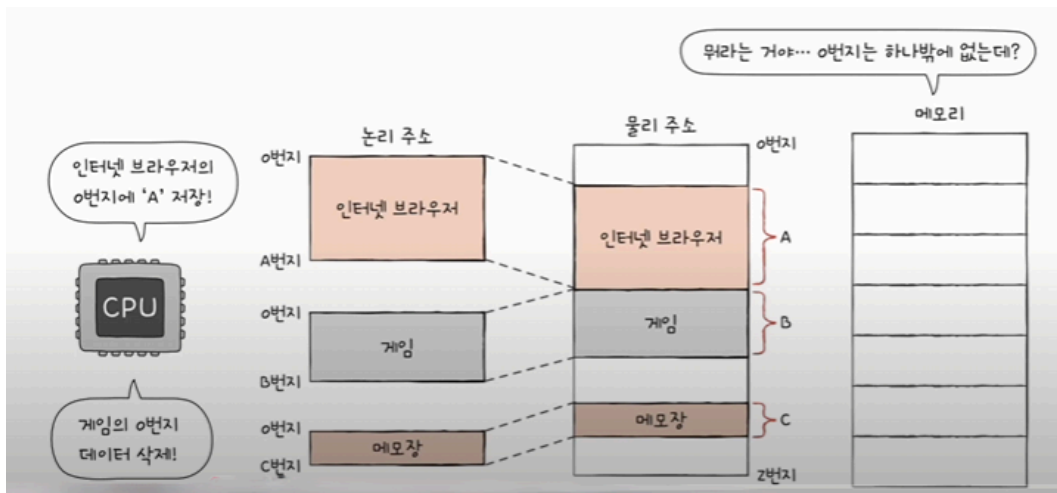
물리 주소

물리주소는 메모리 하드웨어 관점에서의 주소입니다. 말 그대로 실제 메모리 칩 위에 데이터가 저장된 위치를 가르킵니다.

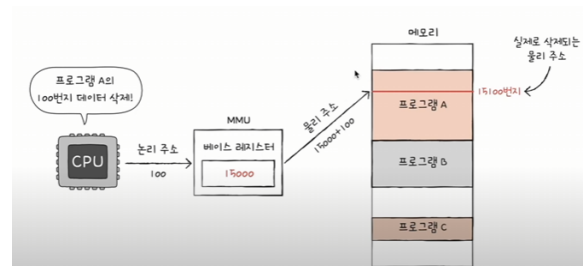
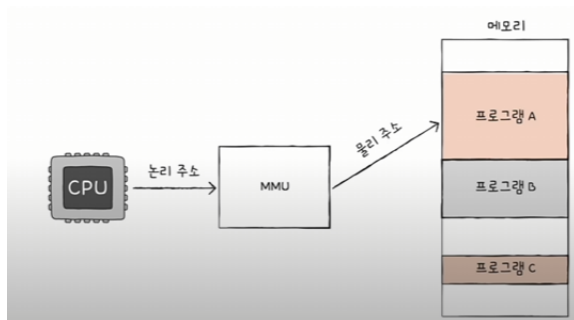
논리 주소

논리 주소는 CPU와 실행 중인 프로그램이 바라보는 주소입니다. 각 프로그램은 자신만의 독립된 주소 공간을 가지며, 보통 0번지부터 시작합니다. 따라서 같은 프로그램이라도 매번 다른 물리 주소에 배치될 수 있지만, 프로그램은 항상 동일한 논리 주소를 사용하는 것처럼 동작합니다. → 이 부분 다시 살펴보기

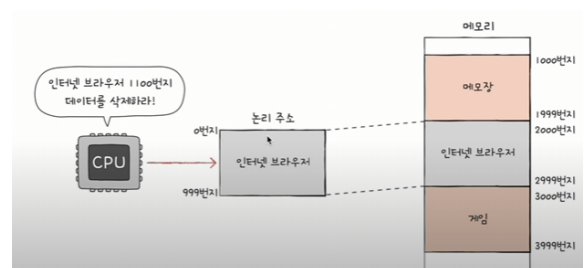
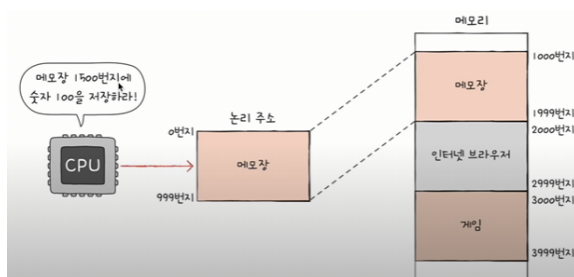
물리 주소와 논리 주소의 변환



물리주소와 논리주소, 이 두 주소 체계 사이의 변환은 MMU(메모리 관리 장치)라는 하드웨어에 의해 수행됩니다. MMU는 프로그램이 사용하는 논리 주소를 받아 베이스 레지스터에 저장된 시작 주소와 더하여 실제 물리 주소를 계산합니다. 이렇게 함으로써 CPU는 논리 주소만 사용해도 되고, 메모리는 하드웨어 차원에서 물리 주소로 접근할 수 있게 됩니다.



메모리 보호



메모리에는 보호 장치가 필요합니다. 예를 들어, 하나의 프로그램이 다른 프로그램의 메모리 영역을 침범하면 시스템 안전성이 깨지기 때문입니다. 이를 막기 위해 베이스 레지스터와 한계 레지스터가 사용됩니다.

베이스 레지스터와 한계 레지스터

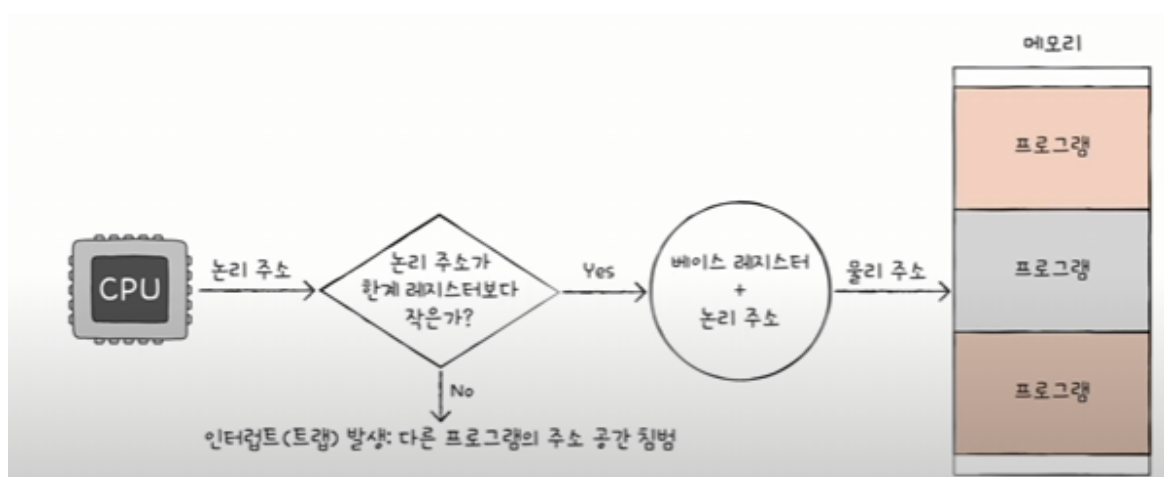
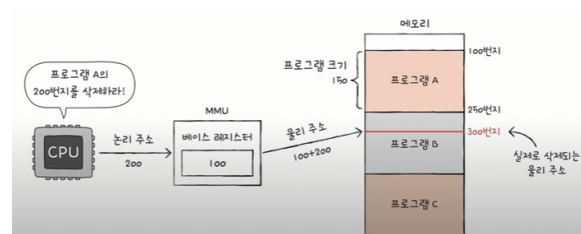
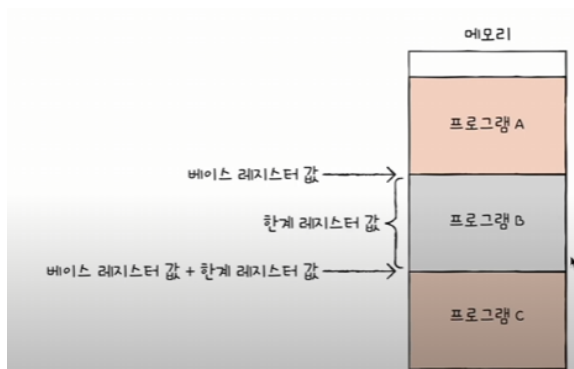
베이스 레지스터는 현재 실행 중인 프로그램이 차지하는 메모리 공간의 시작 주소를 저장합니다.

한계 레지스터는 프로그램이 사용할 수 있는 최대 크기를 기록합니다.

CPU가 어떤 논리 주소에 접근하려고 할 때, 이 값이 한계 레지스터에 저장된 범위를 초과하면 잘못된 접근으로 간주되어 차단됩니다.

예시

베이스 레지스터가 100이고 한계 레지스터가 150이라면, 프로그램의 물리적 실행 범위는 100번지부터 249번지까지입니다. CPU가 그 이상의 주소에 접근하려고 하면 즉시 차단됩니다. 이런 방식으로 프로그램마다 독립적인 실행 공간을 확보하고, 서로 간섭하지 못하도록 보호할 수 있습니다.



결국 논리 주소와 물리 주소의 구분, 그리고 MMU를 통한 변환과 레지스터 기반 보호 기법은 현대 운영체제와 하드웨어에서 프로그램을 안전하고 효율적으로 실행하기 위한 핵심 장

치입니다.