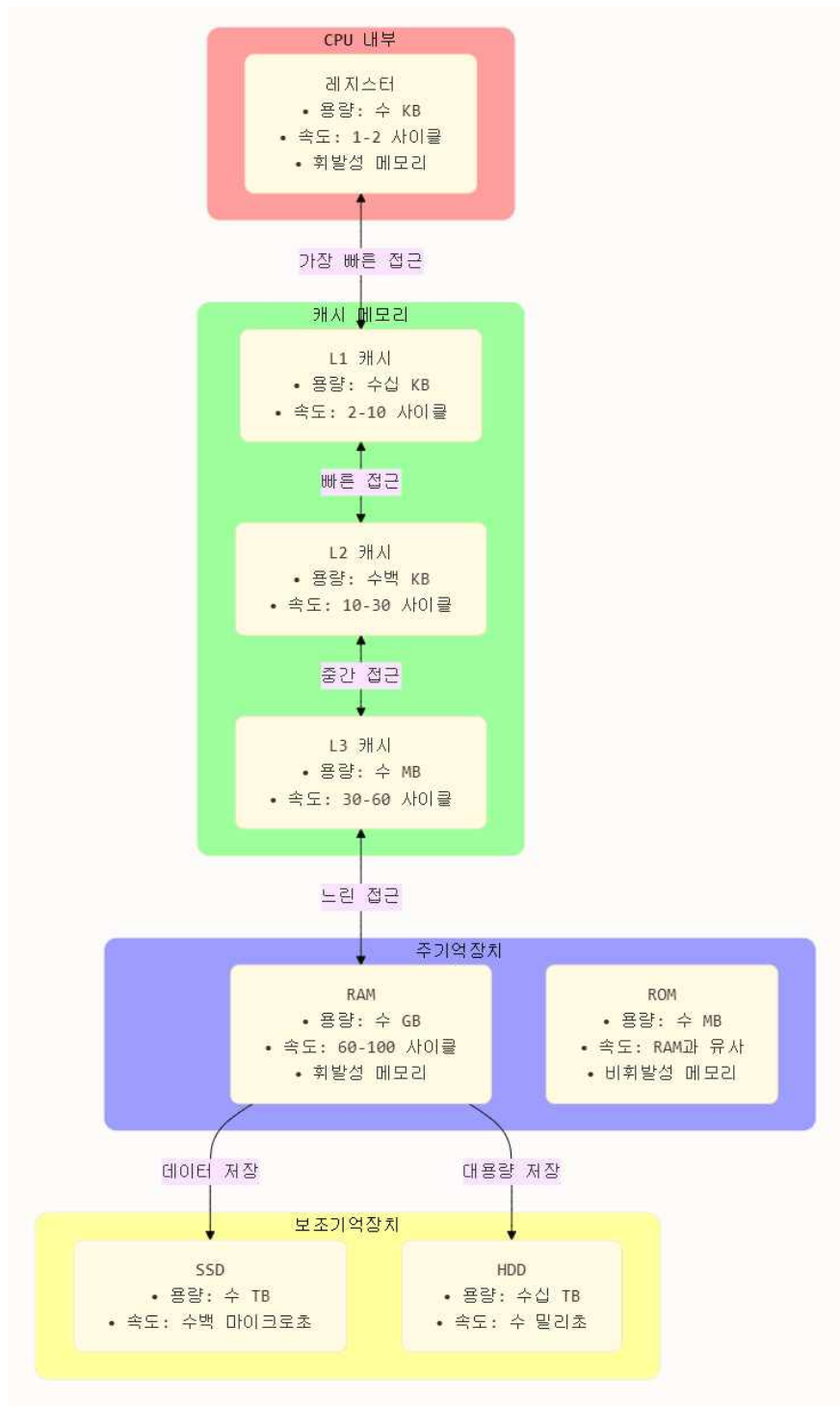


메모리 계층구조 개요

컴퓨터 시스템은 **처리 속도와 용량**에 따라 계층적으로 구성됩니다. 각 계층의 특징과 역할을 이해하는 것이 시스템의 **전체적인 동작을 파악하는 데 매우 중요합니다.**



CPU 내부 구성요소

레지스터 (Register)

- 컴퓨터에서 가장 빠른 메모리입니다.
- 프로세서 내에 위치한 고속 메모리로, 극히 소량의 데이터나 처리 중인 중간 결과를 저장합니다

종류:

전용 레지스터와 범용 레지스터

데이터 레지스터, 주소 레지스터, 상태 레지스터 등으로 구분됩니다

캐시 계층 구조

캐시 메모리 (Cache Memory)

- CPU와 주기억장치 간의 속도 차이를 보완하기 위한 중간 버퍼 역할을 합니다
- L1, L2, L3 캐시로 구분되며, 숫자가 작을수록:
 - 용량이 작음
 - 속도가 빠름
 - CPU에 더 가까움
- SRAM으로 구성되어 있으며, 가장 자주 사용되는 데이터를 저장하여 CPU의 메모리 접근 시간을 줄입니다.

주기억장치 (Main Memory)

RAM (Random Access Memory)

- DRAM으로 구성되며, 전원이 공급되는 동안만 데이터를 유지하는 휘발성 메모리입니다
- 어느 위치에 저장된 데이터든지 동일한 시간으로 접근 가능합니다
- CPU의 직접적인 데이터 처리를 위한 작업 공간으로 사용됩니다

ROM (Read Only Memory)

- 저장된 데이터를 읽기만 가능하고 수정할 수 없는 비휘발성 메모리입니다 m.blog.naver.com
- BIOS나 POST와 같은 기본 시스템 소프트웨어를 저장하는 데 주로 사용됩니다.

보조기억장치 (Storage Device)

SSD (Solid State Drive)

- 순수 전자식으로 작동하여 기계적 지연이 없습니다 m.blog.naver.com

- HDD보다 데이터 읽기/쓰기 속도가 월등히 빠릅니다
- 플래시 메모리 기반으로 작동합니다

HDD (Hard Disk Drive)

- 기계식 구성 요소가 있어 탐색 시간과 반응 시간이 있습니다 m.blog.naver.com
- 대용량 저장이 가능하며, 가격 대비 성능이 좋습니다
- 순차접근이 가능하며 비휘발성 특성을 가집니다

처리 속도 순위

1. 레지스터 (수 나노초 수준)
2. L1 캐시 (수 나노초 수준)
3. L2 캐시 (10-30 사이클)
4. L3 캐시 (30-60 사이클)
5. RAM (60-100 사이클)
6. SSD (수백 마이크로초)
7. HDD (수 밀리초)

CPU 내부/외부 장치 구분

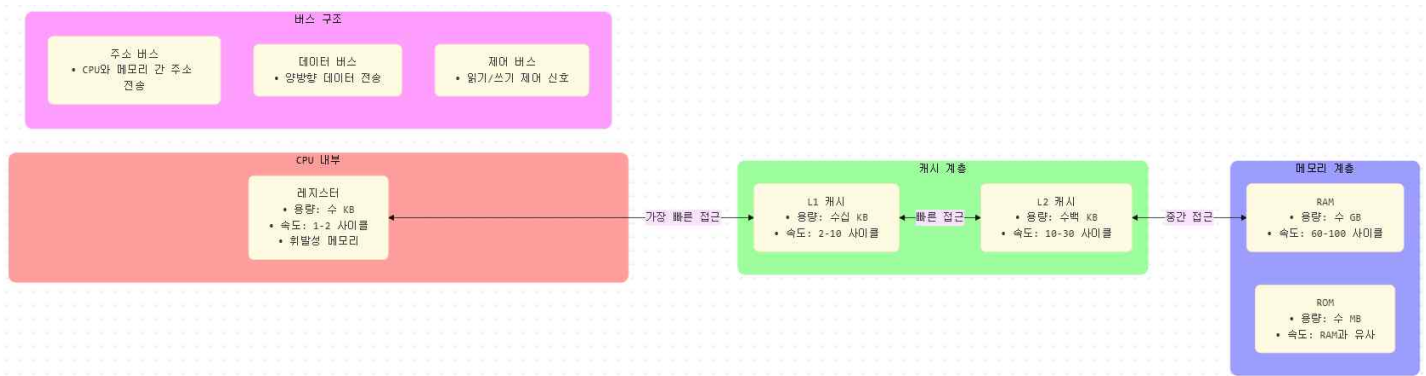
CPU 내부 장치:

- 레지스터
- L1 캐시
- 제어 유닛
- 연산장치

CPU 외부 장치:

- L2/L3 캐시
- RAM
- 입출력 컨트롤러
- 보조기억장치
- 입출력 장치

이러한 계층적 구조는 컴퓨터가 효율적으로 데이터를 처리할 수 있도록 해주며, 각 메모리 장치는 용도에 맞게 최적화되어 있습니다. 특히 레지스터와 캐시는 CPU의 성능을 결정하는 핵심 요소입니다.



버스 구조의 세부 구성

주소 버스 (Address Bus)

- CPU가 주기억장치나 입출력 장치에 데이터를 요청할 때 사용할 주소를 전달하는 데 사용됩니다.
- 단방향으로 작동하며, CPU에서 외부로만 신호가 흐릅니다.
- 버스의 비트 수는 시스템이 접근할 수 있는 최대 메모리 주소 범위를 결정합니다.

데이터 버스 (Data Bus)

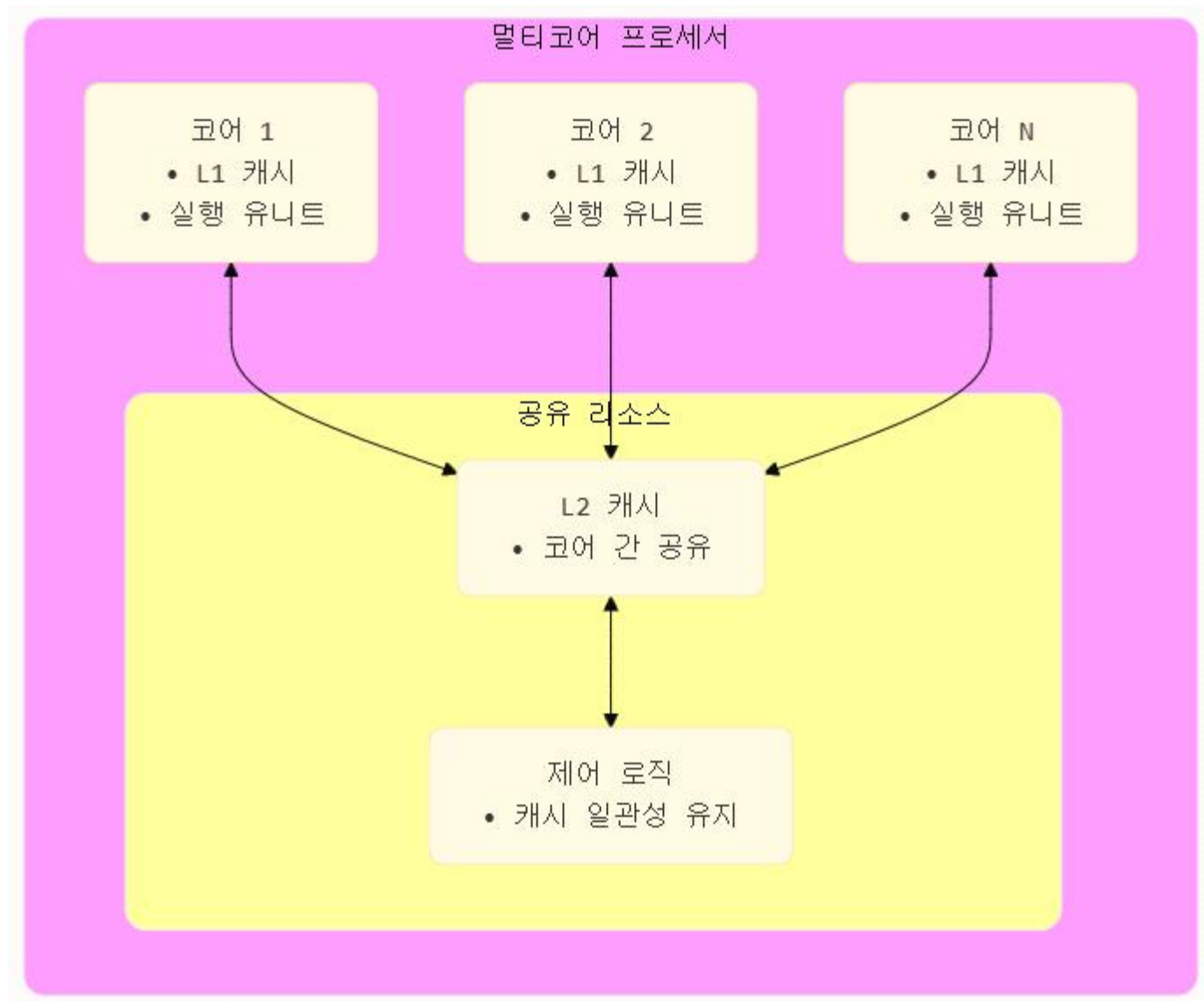
- 양방향으로 작동하여 CPU와 다른 장치 간에 데이터를 전송합니다.
- 데이터 버스의 비트 수는 한 번에 전송 가능한 데이터 크기를 결정합니다.
- 일반적으로 32비트 또는 64비트로 구성됩니다.

제어 버스 (Control Bus)

- 시스템의 모든 장치 간에 제어 신호를 전송합니다.
- 읽기/쓰기 신호, 인터럽트 요청 신호, 타이밍 신호 등을 포함합니다.
- 시스템의 동작을 제어하는 중요한 역할을 수행합니다.

병렬 처리 시스템

- 병렬 처리 시스템의 핵심인 멀티코어 구조를 살펴보겠습니다.



멀티코어 시스템의 특징

- 여러 개의 프로세서 코어가 하나의 패키지에 통합되어 있습니다.
- 각 코어는 독립적으로 프로그램을 실행할 수 있어 처리 능력이 크게 향상됩니다.
- 공유 캐시를 통해 코어 간 데이터 교환이 효율적으로 이루어집니다.
- 캐시 일관성 유지 로직으로 데이터 일관성을 보장합니다.

Quiz

문제 1: 다음 SQL 쿼리의 결과로 올바른 것은 무엇인가?

```
SELECT COUNT(*) FROM employees WHERE department_id =10;
```

- ㄱ) 부서 ID가 10인 직원의 수를 반환한다.
- ㄴ) 부서 ID가 10인 직원의 평균 급여를 반환한다.
- ㄷ) 부서 ID가 10인 직원의 이름을 반환한다.
- ㄹ) 부서 ID가 10인 직원의 최대 급여를 반환한다.

정답: ㄱ)

정답 이유:이 쿼리는 employees테이블에서 department_id가 10인 직원의 수를 세는 COUNT(*)함수를 사용하고 있다. 따라서 결과는 부서 ID가 10인 직원의 총 수가 된다.

문제 2: 다음 SQL 문에서 사용된 JOIN의 종류는 무엇인가?

```
SELECT a.name, b.salary
```

```
FROM employees a
```

```
INNER JOIN salaries b ON a.id = b.employee_id;
```

- ㄱ) LEFT JOIN
- ㄴ) RIGHT JOIN
- ㄷ) INNER JOIN
- ㄹ) FULL OUTER JOIN

정답: ㄷ)

정답 이유:이 쿼리는 INNER JOIN을 사용하여 employees테이블과 salaries테이블을 결합하고 있다. INNER JOIN은 두 테이블에서 조건에 맞는 행만 반환하므로, 직원 ID가 일치하는 경우에만 결과가 포함된다.

문제 3: 다음 SQL 쿼리의 목적은 무엇인가?

```
SELECT DISTINCT department_id FROM employees;
```

- ㄱ) 모든 직원의 부서 ID를 반환한다.
- ㄴ) 중복되지 않는 부서 ID 목록을 반환한다.
- ㄷ) 부서 ID가 10인 직원의 수를 반환한다.
- ㄹ) 부서 ID가 중복된 직원의 수를 반환한다.

정답: ㄴ)

정답 이유:이 쿼리는 DISTINCT키워드를 사용하여 employees테이블에서 중복되지 않는 부서 ID를 선택하고 있다. 따라서 결과는 각 부서 ID가 한 번만 나타나는 목록이 된다.

문제 4: SQL 문법의 분류 중 데이터 정의어(DDL)에 해당하는 명령어는 무엇인가?

ㄱ) SELECT, INSERT, UPDATE, DELETE

ㄴ) CREATE, ALTER, DROP, TRUNCATE

ㄷ) GRANT, REVOKE

ㄹ) COMMIT, ROLLBACK, SAVEPOINT

정답: ㄴ)

정답 이유: 데이터 정의어(DDL)는 데이터베이스의 구조를 정의하고 변경하는 데 사용되는 명령어들을 포함합니다. CREATE(생성), ALTER(변경), DROP(삭제), TRUNCATE(테이블 내용 삭제)는 모두 데이터베이스 객체의 구조를 조작하는 DDL 명령어입니다