

The background of the slide features a top-down view of a white ceramic coffee cup filled with a frothy beverage, resting on a dark, circular shadow. Below the cup is a book with the title 'MIND CAFE' printed in large, serif capital letters. Underneath the title, in smaller text, it says 'EDITION 2 - HABITS'. The book's cover also displays a photograph of a tree. The entire scene is overlaid with a semi-transparent dark grey filter. Two short, horizontal dark grey lines are positioned on either side of the title text.

Raspberry Pi 소켓 통신

PMOD 전성현

목차

- 1 소켓(Socket)
- 2 소켓 통신의 흐름
- 3 프로토콜 종류(TCP,UDP)
- 4 소켓 통신 함수
- 5 회로도

소켓(Socket)이란?



소켓(Socket)
프로세스가 데이터 송수신을 할 때 사용할 수 있는 소프트웨어 적인 장치이다.

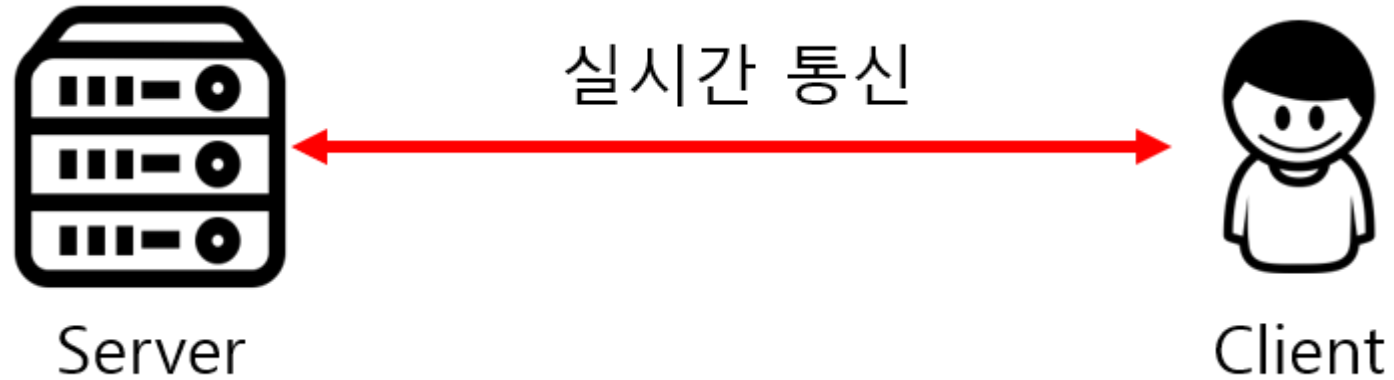
socket 영어
미국 [sɒkɪt]

① 소켓 ② 구멍 ③ 연결 ④ 콘센트 [더보기](#)

실제적인 창구 역할을 함.

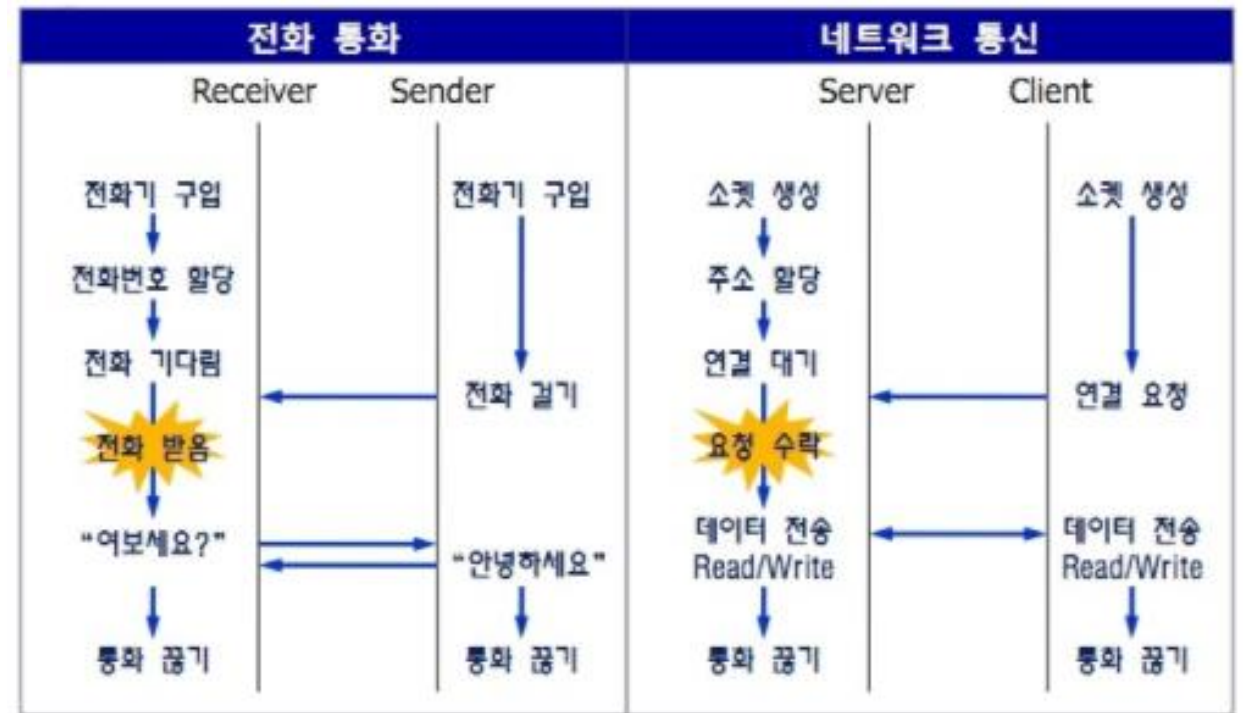
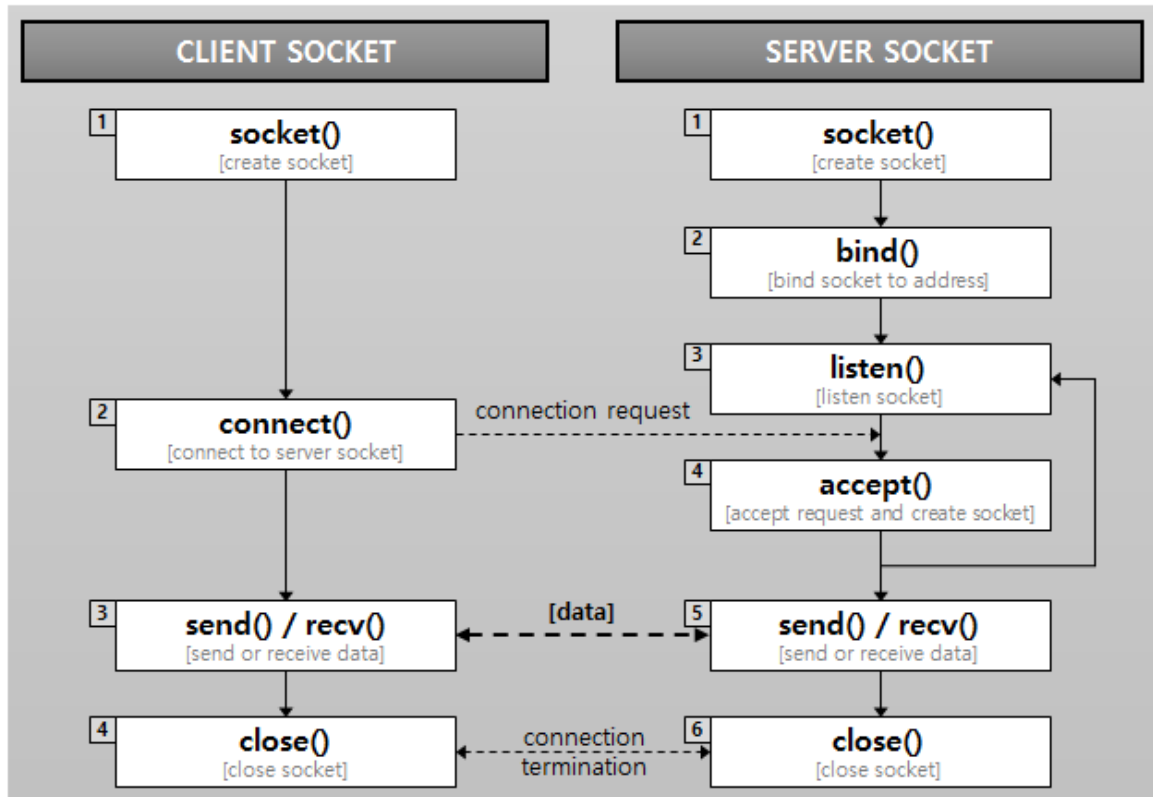
프로토콜, IP 주소, 포트 번호로 정의

소켓 통신



- Server와 Client가 계속 연결을 유지하는 양방향 통신이다.
- 요청 / 허가 없이 지속적으로 상호작용 발생
- 최대 연결 가능 개수가 정해져 있음

소켓 통신의 흐름



프로토콜 종류(TCP,UDP)

TCP

[TCP 서버의 특징]

- 서버소켓은 연결만을 담당한다.
- 연결과정에서 반환된 클라이언트 소켓은 데이터의 송수신에 사용된다형 서비스로 가상 회선 방식을 제공한다.
- 서버와 클라이언트는 1대1로 연결된다.
- 스트림 전송으로 전송 데이터의 크기가 무제한이다.
- 패킷에 대한 응답을 해야하기 때문에(시간 지연, CPU 소모) 성능이 낮다.
- Streaming 서비스에 불리하다.(손실된 경우 재전송 요청을 하므로)

프로토콜 종류(TCP,UDP)

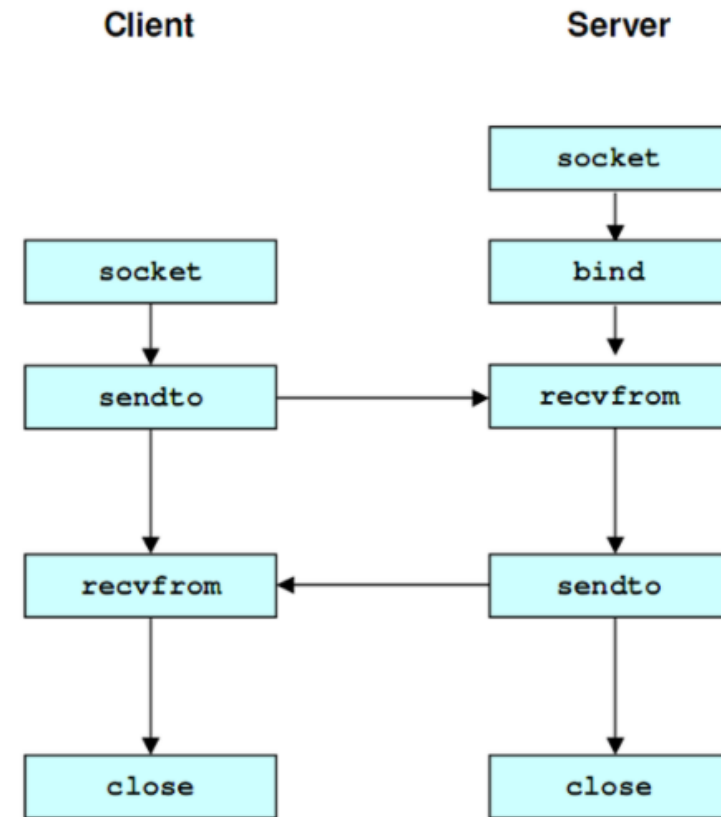
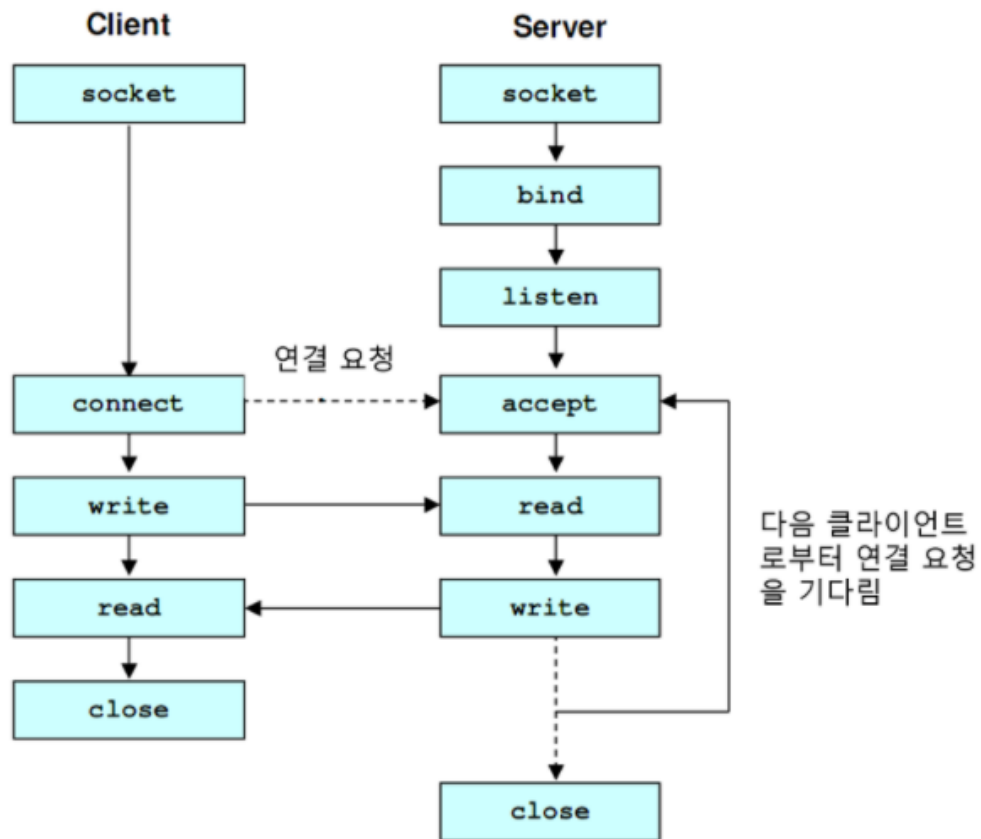
UDP

[UDP 서버의 특징]

- UDP에는 연결 자체가 없어서(connect 함수 불필요) 서버 소켓과 클라이언트 소켓의 구분이 없다.
- 소켓 대신 IP를 기반으로 데이터를 전송한다.
- 서버와 클라이언트는 1대1, 1대N, N대M 등으로 연결될 수 있다.
- 데이터그램(메세지) 단위로 전송되며 그 크기는 65535바이트로, 크기가 초과하면 잘라서 보낸다.
- 흐름제어(flow control)가 없어서 패킷이 제대로 전송되었는지, 오류가 없는지 확인할 수 없다.
- 파일 전송과 같은 신뢰성이 필요한 서비스보다 성능이 중요시 되는 경우에 사용된다.

소켓 통신 종류(TCP,UDP)

TCP 와 UDP 의 통신 흐름



소켓 통신 함수

int socket (int domain, int type, int protocol);

소켓 파일 생성 함수

domain 어떤 영역에서 통신할 것 인지 지정

type 어떤 서비스 타입의 소켓을 생성할지

protocol 소켓에서 사용할 프로토콜

domain	domain 내용
PF_INET, AF_INET	IPv4 인터넷 프로토콜을 사용합니다.
PF_INET6	IPv6 인터넷 프로토콜을 사용합니다.
PF_LOCAL, AF_UNIX	같은 시스템 내에서 프로세스 끼리 통신합니다.
PF_PACKET	Low level socket 을 인터페이스를 이용합니다.
PF_IPX	IPX 노벨 프로토콜을 사용합니다.

type	type 내용
SOCK_STREAM	TCP/IP 프로토콜을 이용합니다.
SOCK_DGRAM	UDP/IP 프로토콜을 이용합니다.

int bind(int sockfd, struct sockaddr *address, socklen_t address_len);

IP주소와 포트 번호를 지정

sockfd 1번함수 리턴값으로 받은 소켓의 디스크립터

struct sockaddr *address 서버의 IP주소

socklen_t address_len 주소의 길이

소켓 통신 함수

```
#include <sys/types.h>
#include <sys/socket.h>
```

```
int listen(int sockfd ,int backlog );
```

소켓을 통해 클라이언트의 접속요청을 기다리도록 설정

sockfd 소켓 디스크립터

backlog 연결대기열의 크기

```
int accept(int sockfd , struct sockaddr*addr, socklen_t*addrlen);
```

sockfd bind함수의 첫 인자와 동일

struct sockaddr*addr 클라이언트의 주소정보를 담고있는 주소체

socklen_t*addrlen 2번째 인자 값의 길이

소켓 통신 함수

```
#include <sys/types.h>
#include <sys/socket.h>
```

```
int connect(int sockfd ,struct sockaddr *serv_addr, socklen_t addrlen);
```

서버로 연결요청하는 함수

struct sockaddr *serv_addr 서버의 주소정보

socklen_t addrlen 2번인자의 크기

```
close(int sockfd);
```

sockfd 종료할 소켓 디스크립터를 인자로 넣어줌

회로도

