

TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정

강사 : Innova Lee(이상훈)

gcccompil3r@gmail.com

학생 : 황수정

sue100012@naver.com

30 일차 (2018. 04. 04)

Hello, Hi 주고 받기

➤ Server 코드

```
#include <stdio.h>
```

```
#include <string.h>
```

```
#include <stdlib.h>
```

```
#include <unistd.h>
```

```
#include <signal.h>
```

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

```
#include <arpa/inet.h>
```

```
typedef struct sockaddr_in si;
```

```
typedef struct sockaddr * sap;
```

```
#define BUF_SIZE 1024
```

```
void err_handler(char *msg)
```

```
{ fputs(msg, stderr);
```

```
  fputc('\n', stderr);
```

```
  exit(1);
```

```
}
```

```
int main (int argc, char **argv)
```

```
{
```

```
  int serv_sock, clnt_sock;
```

```
  int read_cnt, fd;
```

```
  si serv_addr, clnt_addr;
```

```
  socklen_t clnt_addr_size;
```

```
  char msg_1[BUF_SIZE] = {0}; //client 에서 hi 받는 용도로
```

```
  char msg[] = "Hello";
```

```
  if(argc !=2)
```

```
  {
```

```
    printf("use : %s <port>\n", argv[0]);
```

```
    exit(1);
```

```
  }
```

```

serv_sock = socket(PF_INET, SOCK_STREAM, 0);

if(serv_sock == -1)
    err_handler("socket() error");

memset(&serv_addr, 0, sizeof(serv_addr));
serv_addr.sin_family = AF_INET;
serv_addr.sin_addr.s_addr = htonl(INADDR_ANY);
serv_addr.sin_port = htons(atoi(argv[1]));

if(bind(serv_sock, (sap)&serv_addr, sizeof(serv_addr)) == -1)
    err_handler("bind() error");

if(listen(serv_sock, 5) == -1) //hi, hello 만 주고 받을 거지만... 일단
    err_handler("listen() error");

clnt_addr_size = sizeof(clnt_addr);
clnt_sock = accept(serv_sock, (struct sockaddr *)&clnt_addr, &clnt_addr_size);

if(clnt_sock == -1)
    err_handler("accept() error");

write(clnt_sock, msg, sizeof(msg)); // hello 를 client 에 전달
// clnt_sock = 0; // 소켓의 초기화가 필요하다고 생각했는데.. 이런 식은 아닌 것 같다.

for(;;)
{ read_cnt = read(clnt_sock, msg_1, BUF_SIZE);
  if(read_cnt < BUF_SIZE)
  { write(clnt_sock, msg_1, read_cnt);
    break;
  }
  write(clnt_sock, msg_1, BUF_SIZE);
} //처음 코드 만들 때, 읽고 써야 읽을 제대로 읽을 수 있을 거 같다는 생각에 했는데
Hello 로 Hi 만 주고 받는 거라 읽기만 해도 괜찮을 것 같다.

```

```

read(clnt_sock, msg_1, BUF_SIZE);
printf("msg form client : %s\n", msg_1);

//close(fd);
close(clnt_sock);
close(serv_sock);

return 0;
}

```

➤ Client 코드

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <sys/socket.h>

typedef struct sockaddr_in si;
typedef struct sockaddr * sap;

#define BUF_SIZE 32

void err_handler(char *msg)
{
    fputs(msg, stderr);
    fputc('\n', stderr);
    exit(1);
}

int main(int argc, char **argv)
{
    int sock, fd;
    int str_len;

```

```

si serv_addr;
char msg[BUF_SIZE] = {0}; //hello 받을 용도
char msg_1[] = "Hi";

if(argc !=3)
{
    printf("use : %s <IP> <port>\n", argv[0]);
    exit(1);
}

sock = socket(PF_INET, SOCK_STREAM, 0);

if(sock == -1)
    err_handler("socket() error");

memset(&serv_addr, 0, sizeof(serv_addr));
serv_addr.sin_family = AF_INET;
serv_addr.sin_addr.s_addr = inet_addr(argv[1]);
serv_addr.sin_port = htons(atoi(argv[2]));

if(connect(sock, (sap)&serv_addr, sizeof(serv_addr)) == -1)
    err_handler("connect() error");

write(sock, "Hi", 3); // 위 선언한 배열 이용이 아니라 그냥 보내는 거로 해봄
// sock = 0;

str_len = read(sock, msg, sizeof(msg)-1);
if(str_len == -1)
    err_handler("read() error!");
printf("msg form serv :%s\n", msg);
close(sock);
return 0;
}

```