1、 网络连接

控制卡 ip：192.168.0.16　　端口号 ：10000

2、 控制指令

网络连接后发送以下指令控制开关打开或关闭（16 进制发送）：

控制开关打开：FE 05 00 00 FF 00 98 35

控制开关关闭：FE 05 00 00 00 00 D9 C5

3、 示例程序

Linux 环境在编译下例程序，gcc -o client client.c
运行程序 ./client on 打开开关，./client off 关闭开关
具体操作可根据程序进行修改

```
///文件名 client.c
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <linux/in.h>
#include <string.h>

unsigned short portnum = 10000;
char serverip[] = "192.168.0.16";

int main(int arg, char** argv)
{
    if(arg < 2)
    {
        char* mstr = argv[0];
        printf("%s\n", mstr);
        printf("please choose mood: on, off, check\n");
        return;
    }
    //控制指令缓存
    char bufon[8] = {0xFE, 0x05, 0x00, 0x00, 0xFF, 0x00, 0x98, 0x35};
    char bufoff[8] = {0xFE, 0x05, 0x00, 0x00, 0x00, 0x00, 0xD9, 0xC5};
    char bufcheck[8] = {0xFE, 0x01, 0x00, 0x00, 0x00, 0x04, 0x29, 0xC6};

    int cfd;
```

```
char buffer[10] = {0};
struct sockaddr_in s_add,c_add;

cfd = socket(AF_INET, SOCK_STREAM, 0);
if(-1 == cfd)
{
    printf("socket fail ! \r\n");
    return -1;
}
//socket 初始化
bzero(&s_add,sizeof(struct sockaddr_in));
s_add.sin_family=AF_INET;
s_add.sin_addr.s_addr= inet_addr(serverip);
s_add.sin_port=htons(portnum);

printf("s_addr = %#x ,port : %#x\r\n",s_add.sin_addr.s_addr,s_add.sin_port);

if(-1 == connect(cfd,(struct sockaddr *)(&s_add), sizeof(struct sockaddr)))
{
    printf("connect fail !\r\n");
    return -1;
}

printf("Hello,welcome to client !\r\n");
//**********socket complete.******************//

int cmd = -1;
if(!strcmp(para1, "on")     || !strcmp(para1, "ON"))
    cmd = 1;
if(!strcmp(para1, "off")    || !strcmp(para1, "OFF") )
    cmd = 2;
if(!strcmp(para1, "check") || !strcmp(para1, "CHECK"))
    cmd = 3;
printf("cmd = %d\n", cmd);

if(-1 != cmd)
{
    switch(cmd)
    {
        case 1:
            if(-1 == write(cfd, bufon, 8))//打开开关
            {
                printf("write on fail !\n");
                return -1;
                }
                break;
            case 2:
                if(-1 == write(cfd, bufoff, 8))//关闭开关
                {
                    printf("write off fail !\n");
                    return -1;
                }
                break;
            case 3:
                if(-1 == write(cfd, bufcheck, 8))
                {
                    printf("write check failed !\n");
                    return -1;
                }
            default:
                break;
        }
    }

    usleep(10000);
    int recbytes;

    if(-1 == (recbytes = read(cfd,&buffer,sizeof(buffer) ) ))
    {
        printf("read data fail !\r\n");
        return -1;
    }
    int i;
    for(i=0;i<recbytes;i++)
    {
        //if(buffer[i]!='\0')
            printf("%.2X ", buffer[i]);

    }
    printf("read ok !\n");

    close(cfd);
    return 0;

}
```