### 实验报告：

### 一、实验目的

尝试测试210目标板与笔记本电脑Kali Linux间通过串口线路进行串口程序通讯

### 二、实验器材与原料

210目标板、安装Linux操作系统的笔记本电脑、USB-串口线x2、串口通讯程序

### 三、实验原理

两设备串口连接进行通讯

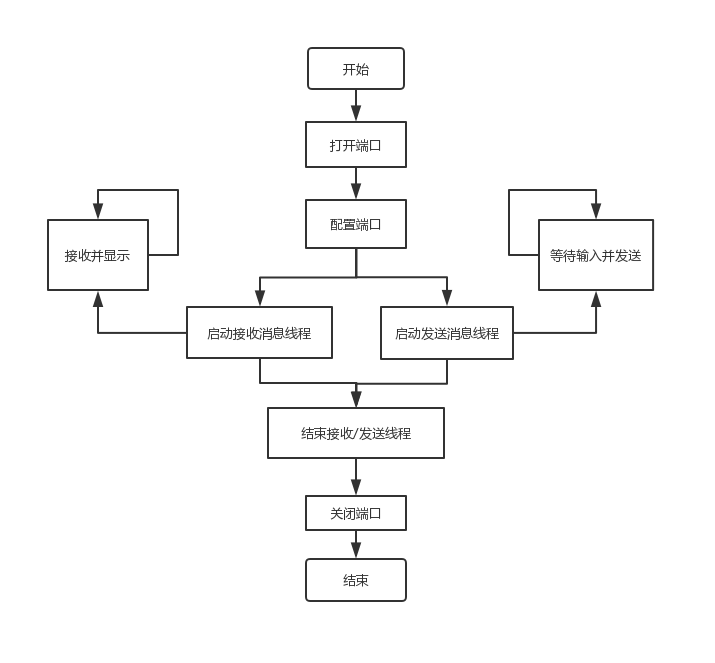
### 四、实验步骤

1、将210目标板与笔记本电脑用USB-串口线连接好（目标板一端连接调试口），目标板与台式机用USB-串口线连接（目标板一段连接tty显示口）。将串口通讯程序客户端放入SD大卡中。

2、将SD大卡插入210目标板，用台式机对210发出命令，使210目标板执行串口通讯程序，笔记本电脑端执行串口通讯程序。

3、在笔记本电脑与210目标板上进行双方chat，测试可以通过USB-串口线进行串口通讯。

### 五、实验数据记录（或调试好的程序包括流程图）

//ComChat\_PC

#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<termios.h>

#include<string.h>

#include<fcntl.h>

#include<pthread.h>

void\* th\_Recv(void \*data)

{

char buf[512] = { 0 };

int f\_stop = 0;

do

{

read(\*(int\*)data, buf, 1);

if (buf[0])

{

printf("%c", buf[0]);

buf[0] = 0;

}

} while (1);

return 0;

}

void\* th\_Send(void \*data)

{

char buf[512] = { 0 };

int f\_stop = 0;

do

{

printf("Input to send(\"![QUIT] to quit\"):\n");

scanf("%s", buf);

if (!strncmp(buf, "![QUIT]", 7))

f\_stop = 1;

else

write(\*(int\*)data, buf, strlen(buf));

} while (!f\_stop);

return 0;

}

int main(void)

{

int fd;

struct termios opt;

pthread\_t thread\_Recv, thread\_Send;

void \*ret;

fd = open("/dev/ttyUSB0", O\_RDWR | O\_NOCTTY | O\_NDELAY);

if (fd == -1)

{

printf("COM open error!\n");

return 0;

}

//get current configuration of COM

tcgetattr(fd, &opt);

tcflush(fd, TCIFLUSH);

//set speed(115200Hz)

cfsetispeed(&opt, B115200);//input

cfsetospeed(&opt, B115200);//output

//data bits(8bits)

opt.c\_cflag &= ~CSIZE;

opt.c\_cflag |= CS8;

//set chk

opt.c\_cflag &= ~PARENB;

opt.c\_iflag &= ~INPCK;

//set stop flag

opt.c\_cflag &= ~CSTOPB;

//time out(150s)

opt.c\_cc[VTIME] = 150;

opt.c\_cc[VMIN] = 0;

//write conf to COM

tcsetattr(fd, TCSANOW, &opt);

tcflush(fd, TCIFLUSH);

pthread\_create(&thread\_Recv, NULL, th\_Recv, &fd);

pthread\_create(&thread\_Send, NULL, th\_Send, &fd);

pthread\_join(thread\_Send, &ret);

pthread\_cancel(thread\_Recv);

close(fd);

return 0;

}

//ComChat\_Emb

#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<termios.h>

#include<string.h>

#include<fcntl.h>

#include<pthread.h>

void\* th\_Recv(void \*data)

{

char buf[512] = { 0 };

int f\_stop = 0;

do

{

read(\*(int\*)data, buf, 1);

if (buf[0])

{

printf("%c", buf[0]);

buf[0] = 0;

}

} while (1);

return 0;

}

void\* th\_Send(void \*data)

{

char buf[512] = { 0 };

int f\_stop = 0;

do

{

printf("Input to send(\"![QUIT] to quit\"):\n");

scanf("%s", buf);

if (!strncmp(buf, "![QUIT]", 7))

f\_stop = 1;

else

write(\*(int\*)data, buf, strlen(buf));

} while (!f\_stop);

return 0;

}

int main(void)

{

int fd;

struct termios opt;

pthread\_t thread\_Recv, thread\_Send;

void \*ret;

fd = open("/dev/ttySAC3", O\_RDWR | O\_NOCTTY | O\_NDELAY);

if (fd == -1)

{

printf("COM open error!\n");

return 0;

}

//get current configuration of COM

tcgetattr(fd, &opt);

tcflush(fd, TCIFLUSH);

//set speed(115200Hz)

cfsetispeed(&opt, B115200);//input

cfsetospeed(&opt, B115200);//output

//data bits(8bits)

opt.c\_cflag &= ~CSIZE;

opt.c\_cflag |= CS8;

//set chk

opt.c\_cflag &= ~PARENB;

opt.c\_iflag &= ~INPCK;

//set stop flag

opt.c\_cflag &= ~CSTOPB;

//time out(150s)

opt.c\_cc[VTIME] = 150;

opt.c\_cc[VMIN] = 0;

//write conf to COM

tcsetattr(fd, TCSANOW, &opt);

tcflush(fd, TCIFLUSH);

pthread\_create(&thread\_Recv, NULL, th\_Recv, &fd);

pthread\_create(&thread\_Send, NULL, th\_Send, &fd);

pthread\_join(thread\_Send, &ret);

pthread\_cancel(thread\_Recv);

close(fd);

return 0;

}

### 六、结论

使用串口通讯要注意以O\_RDWR | O\_NOCTTY | O\_NDELAY方式打开串口。