1. 使用多线程

思路：

1. 建立一个网络通道用于连接客户端和服务器，通道IP地址和端口可通过参数设置；
2. 客户端向服务器发送升级请求，服务器确认可以升级后，向客户端发送应答信号，客户端收到信号后开始发送接收服务器发送来的升级包；
3. 客户端对接收过来的升级包解压缩并校验是否正确，升级设备，完成之后向服务器发送升级结果，表明升级完成。
4. 解压到 dav 目录即可
5. thread1 -> update\_thread\_tid
6. thread2 -> heartbeat\_thread\_tid
7. main 函数要加 容错机制 arg数字不对 避免 段错误，客户端模仿服务器端的方法
8. 主函数 printf 都改成 perror
9. 中文全部改成英文打印
10. 加打印调试宏
11. 加assert宏
12. 加getopt
13. 传输文件名默认指定为"digicap.dav"; 但是可以更改
14. client程序只可以有main函数自己

模块化：

1. 不需要露出接口的不露
2. client\_tool 中只有main函数
3. 服务器端可指定路径，客户端也要指定路径
4. 自己文件的函数用static 声明
5. 使用 空的#define IN #define OUT 来定义参数
6. header.file\_nums \* UPGRADE\_FILE\_HEADER\_SIZE + FIRMWARE\_HEADER\_SIZE 定义成一个变量
7. \*p = (int)maiiloc
8. p=(int \*)malloc
9. 用求差取代求模

gets(s);

相当于

fgets(s, sizeof(s), stdin);

if(s[strlen(s) - 1] == '\n') s[strlen(s) - 1] = '\0'; // 去掉换行符

1. main函数容错处理
2. 流程图重新画
3. 进度条显示重新做
4. 心跳有问题
5. 没有判断覆盖
6. 文件模块化最后做，先弄懂所有程序
7. 改文件名，改输出显示，改变量名
8. process\_cmd -> judge\_cmd