《嵌入式系统》期末复习纲要

1. FS3399M4实验箱中的MPU(RK3399)和MCU(STM32)各自基于ARM家族中的哪个核心（cortex-A系列，cortex-M系列，cortex-R系列）

MPU：**RK3399基于ARM Cortex-A系列的Cortex-A72核心**

MCU：**STM32基于ARM Cortex-M系列的Cortex-M4核心**

最新的 ARM 处理器产品是 ARM Cortex-A 系列、ARM Cortex-R 系列、ARM Cortex-M 系列，其中 Cortex-A 系列又称为 高性能处理器 ，Cortex-R 系列又称为 实时处理器 ，ARM Cortex-M 系列又称为 低成本、低功耗 处理器。嵌入式系统的前身通常称为 单片机 。

1. 实验环境的搭建，了解各个开发平台，软件的名称，如vmware， Ubuntu，putty，xshell，其作用是什么？

### VMware作用：虚拟化软件，用于在一台物理计算机上运行多个虚拟机（VM）。用于模拟嵌入式开发和测试操作系统，可以在单一主机操作系统上模拟不同的目标平台，方便进行跨平台测试和开发

### Ubuntu作用：Linux 发行版，为嵌入式开发提供了开发环境，工具链，必要的库软件包等等，用于编译和调试嵌入式程序。

### Putty作用：简单的串口连接软件。便于快速连接实验箱，进入 U-Boot 或命令行终端，方便对开发板进行调试和配置。PuTTY 支持串口连接，可以作为串口终端仿真器使用。在嵌入式开发中，开发者经常需要通过串口与开发板进行通信，以调试程序或查看输出信息。PuTTY 允许用户配置串口参数（如波特率、数据位、停止位等），并连接到开发板的串口，从而实现数据的双向传输

### Xshell作用：功能强大的串口连接软件，可以通过它对电脑和开发板进行文件传输。

常见的嵌入式操作系统： ① 嵌入式Linux ② VxWorks ③ μC/OS-II ④ Windows CE ⑤ Sysbian ⑥ Android ⑦ iOS

最受欢迎的 10 个 Linux 发行版：Ubuntu 、Fedora 、....

1. 镜像烧写的概念？开发板中的存储设备是什么？

镜像烧写的概念**：镜像烧写的过程是将操作系统或固件的镜像文件从电脑传输并写入到硬件设备的存储介质中，使其能够在设备启动时被加载和执行。**

开发板中的存储设备是**：Flash Memory（闪存） 、 SDRAM**

1. “ARM裸机”的含义，实验中如何进行操作控制的？

**ARM裸机指的是在没有操作系统支持的情况下，直接在 ARM 处理器上运行的程序，程序直接访问硬件。一个没有运行任何操作系统的ARM架构的微处理器或微控制器**

**实验中直接通过设置GPIO的寄存器配置GPIO的功能，并在程序中直接控制或读取GPIO以实现控制硬件或读取硬件信息的功能**

1. 嵌入式系统的交叉开发环境指的是什么？（教材P16-1.4.4）

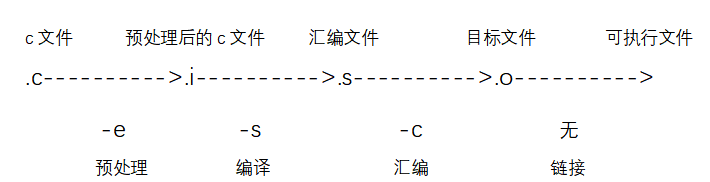
**交叉开发环境是指编译、链接和调试嵌入式应用软件的环境**, 通常采用宿主/目标机模式。**嵌入式系统的交叉开发环境一般包括交叉编译器,交叉调试器和系统仿真器**,其中交叉编译器用于在宿主机上生成能在目标机上运行的代码,而交叉调试器和系统仿真器则用于在宿主机与目标机间完成嵌入式软件的调试。

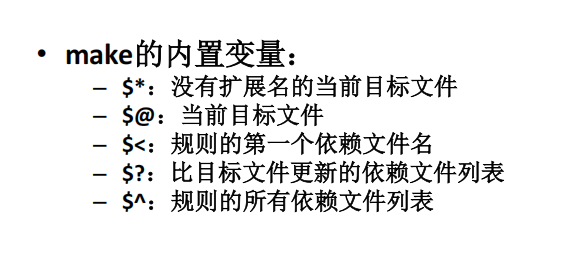
1. 宿主机和目标机之间通信的四种方式（教材P86）

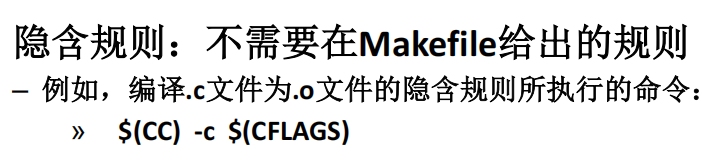
**串口、以太网接口、USB 接口和 JTAG接口**。

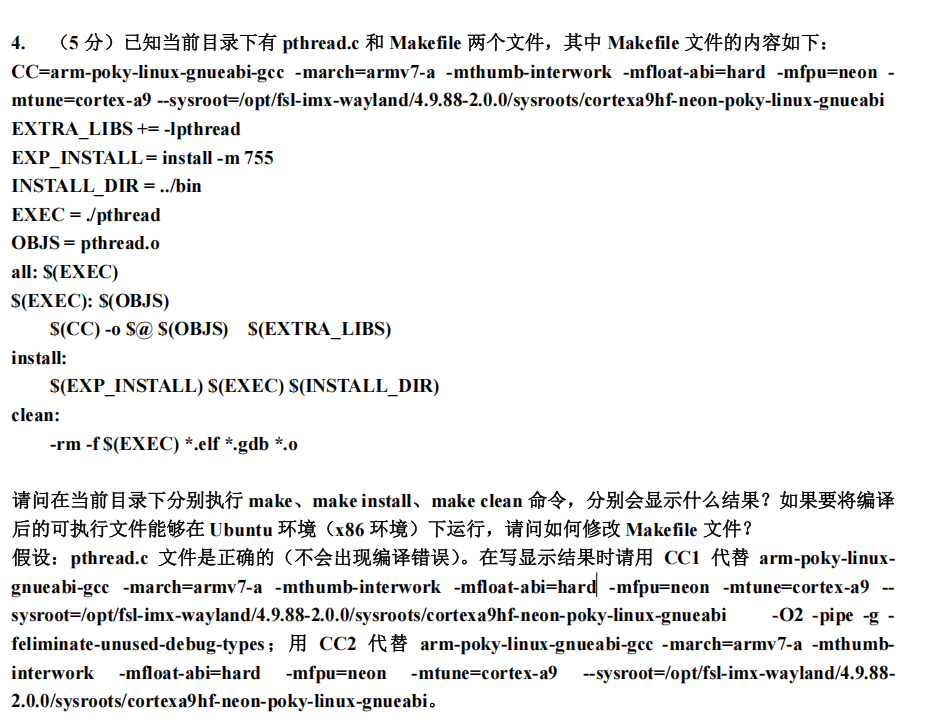
1. GCC的编译过程以及对应的具体命令（教材P90-91）

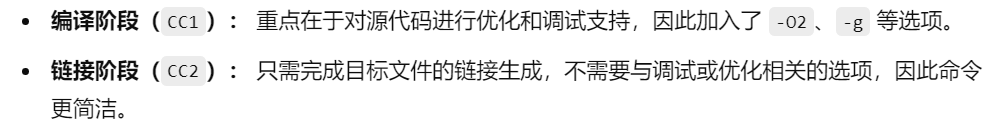
**语序： gcc [选项] -o [输出文件] [源文件]**











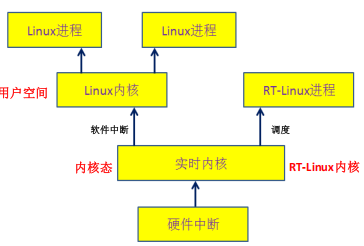
1. ARM状态和THUMB状态是如何切换的？（教材P22-2.2.2ARM状态和Thumb状态）

**ARM 状态和 Thumb 状态切换可以通过 BX(Branch exchange)指令来实现**。BX 指令将通用寄存器 Rn(R0~R15)的值复制到程序寄存器 PC 中来实现 4G地址范围的绝对跳转。**BX指令通过判断 Rn所存储的目标地址的最后一位来判断要跳转到什么状态:若 Rn[0]=0,则跳转到 ARM 状态,若 Rn[0]=1,则跳转到 Thumb 状态**。CPSR 的第5位是T位,即状态控制位,它的值决定了处理器的运行状态。当T=1时,处理器处于 Thumb状态,当T=0时,则处于 ARM 状态。可以通过直接修改 CPSR的T位来达到切换运行状态的目的,但是这样有时会出现问题。

1. RT-Linux（教材P56-3.1.2 &P60-3.3.2 ）

RT-Linux（Real-Time Linux）：具有**硬实时特性的多任务操作系统**。通过在**Linux内核**与**硬件中断**之间增加一个**可抢先的实时内核**，把标准的Linux内核作为实时内核的一个进程与用户进程一起调度，标准的Linux内核的优先级最低，可以被实时进程抢断。正常的Linux进程仍可以在Linux内核上运行，这样既可以使用标准分时操作系统（即Linux的各种服务），又能提供低延时的实时环境。

RT-Linux有两种中断：**硬中断**是实现实时Linux内核的前提，**软中断**是常规Linux内核中断。RT-Linux默认的情况下采用**优先级的调度策略**，也支持其他的调度策略。为保证RT-Linux实时进程与非实时Linux进程进行数据交换， RT-Linux引入了**RT-FIFO队列**。 RT-Linux程序运行于**用户空间和内核态**两个空间，实时处理部分编写成内核模块，并装载到RT-Linux内核中，运行于RT-Linux的**内核态**。非实时部分的应用程序则在Linux下的**用户空间**中执行



1. 嵌入式文件系统分类（教材P66，典型代表JFFS2，YAFFS2,Ramfs）

*\*嵌入式系统主要的存储设备为：Flash ROM（闪存）、RAM*

**（1）基于Flash ROM的文件系统：**

① **JFFS2文件系统 ：**Journalling Flash File System 2（闪存日志型文件系统第2版），主要用于NOR Flash

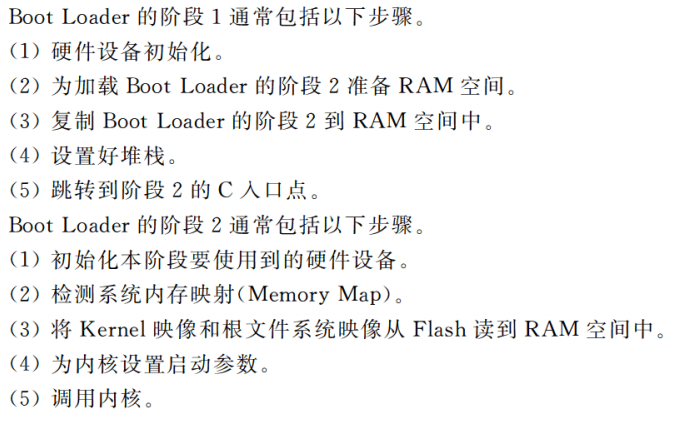
② **YAFFS2文件系统** ：Yet Another Flash File System 2，是一种和JFFS类似的闪存文件系统，主要用于NAND Flash

**（2）基于RAM的文件系统：**

**Ramfs:** 是一种基于内存的文件系统，工作于虚拟文件系统(VFS)层，不能格式化，可以创建多个，在创建时可以指定其最大能使用的内存大小。数据掉电即丢失。

**（3）网络文件系统（NFS）：Network File System（网络文件系统）:** 它允许网络中的计算机之间通过TCP/IP网络共享资源。

1. Bootloader的两阶段主要完成的工作（参看教材P109）



1. 字符设备，块设备与网络设备的定义

字符设备**：字符设备就是采用字节流形式通讯的I/O设备，是无须缓冲直接读写的设备，如鼠标、键盘、显示器、串口等。**

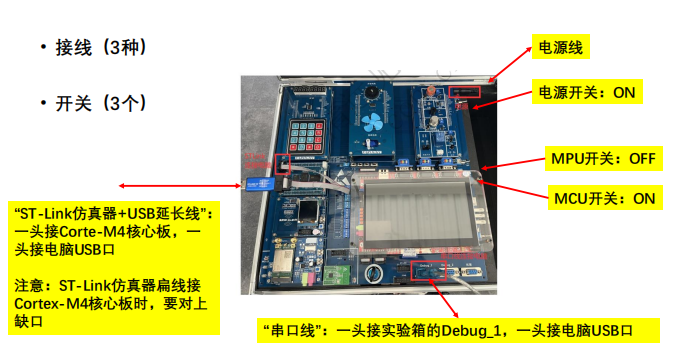
块设备**：是通过缓冲区进行（缓冲区通常为系统内存）、 只能以固定大小的块为单位进行读写的设备，支持随机访问。如硬盘等设备。**

网络设备**：网络设备是指通过网络接口通信的设备，可以通过BSD套接口访问，是以数据包的形式进行。**

1. 呼吸灯实验中的pwm英文解释， MDK的英文全称，JTAG，I2C，CAN

PWM：**脉冲宽度调制，是英文“Pulse Width Modulation”的缩写，简称[脉宽调制](https://www.eeworld.com.cn/zhuanti/LOeLuL" \t "https://news.eeworld.com.cn/mcu/_blank)，是利用微处理器的数字输出来对模拟电路进行控制的一种非常有效的技术。**  
MDK：**Microcontroller Development Kit 微控制器开发套装**  
JTAG：**Joint Test Action Group，联合测试工作组，是一种国际标准测试协议，主要用于芯片内部测试。**

I^2C：**Inter Integrated-Circuit，IIC，I2C，内部集成电路总线。**  
CAN：**Controller Area Network，即控制器局域网**

1. STM32实验是如何接线的（除了电源线），其作用是什么？ 

**ST-LINK仿真器+USB延长线 支持快速擦写和烧录固件，提供了与微控制器核心的调试接口，烧录程序调试程序**

**串口线用于使电脑和实验箱进行通信，可以进行串口调试与数据传输。**

STM32所开展的实验项目中，哪些需要使用串口线？

**直流电机、串口通信、两个按键、继电器、小键盘/数码管，ADC数模转换、红外接收、温度采集、光电开关需要串口线**



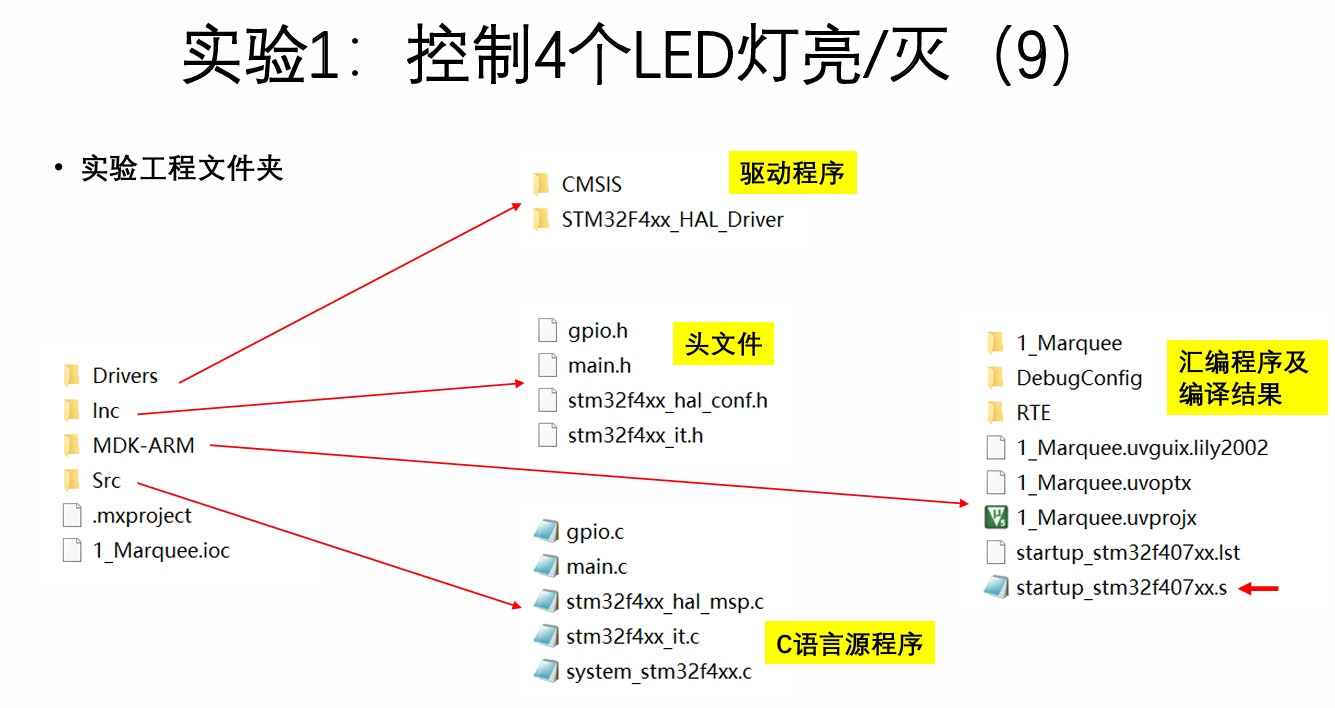
STM32实验程序的工程文件中的主要文件夹，分别存放什么文件？

**Drivers文件夹，通常包含硬件驱动程序，负责与STM32微控制器的外设进行交互。**

**INC文件夹：通常包含头文件。**

**MDK-ARM：包含与MDK-ARM开发环境相关的文件，通常是项目的配置文件、编译设置、调试配置等。**

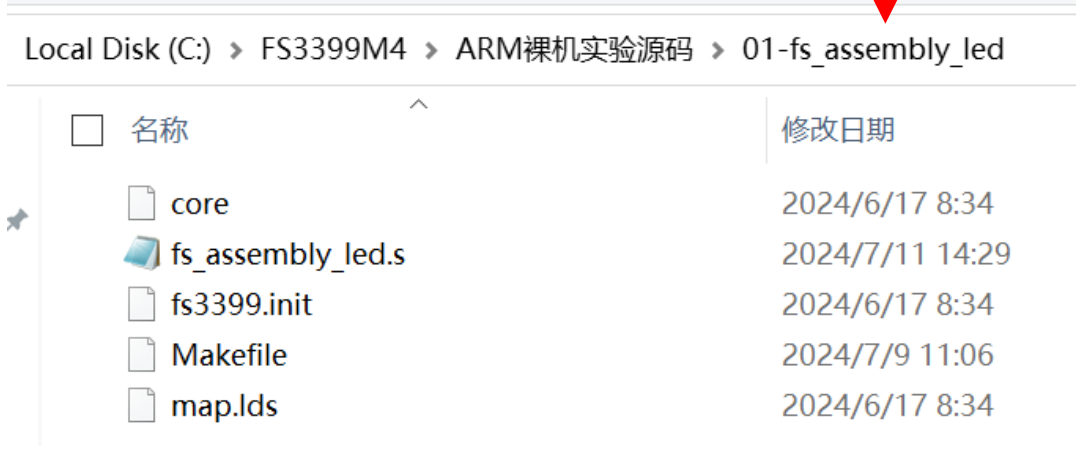
**Src文件夹：包含文件夹的源代码文件。这些文件实现了具体的功能逻辑。**

****

1. ARM裸机实验和Qt实验，会涉及到同样的接线，其用途有何不同？

**两者都接串口线，裸机用于查看串口调试信息，qt还用于传输可执行程序到实验箱，以及对开发板进行调试控制都涉及到连接type-c线，其中裸机实验烧写ubuntu镜像，qt实验烧写linux镜像**

1. ARM裸机实验中LED实验的文件夹中那几个文件都是做什么用的？

  
**core：系统内核文件，用于提供项目的基础功能**

**fs\_assembly\_led.s：汇编语言程序，用于编写与硬件紧密相关的代码。**

**fs3399.init：远程调试初始化文件**

**Makefile：编译脚本**

**map.lds：链接脚本**

1. 设备驱动程序设计中涉及的内存映射都有哪些？函数原型是什么？（教材P206-9.5 以及相应的PPT）

**一、物理地址映射到虚拟地址**

在内核中访问I/O内存（I/O与内存统一编址，访问I/O就像访问内存一样） 之前，我们只有I/O内存的物理地址，这样是无法通过软件直接访问的，需要首先用**ioremap()**函数将设备所处的**物理地址**映射到**内核虚拟地址**空间（3GB~4GB），然后，才能根据映射所得到的内核虚拟地址范围，通过访问指令访问这些I/O内存资源。

**• void \* ioremap(unsigned long phys\_addr, unsigned long size, unsigned long flags)**

– phys\_addr：要映射的起始的I/O地址

– size：要映射的空间的大小

– flags：要映射的I/O空间的和权限有关的标志

**二、内核空间映射到用户空间**

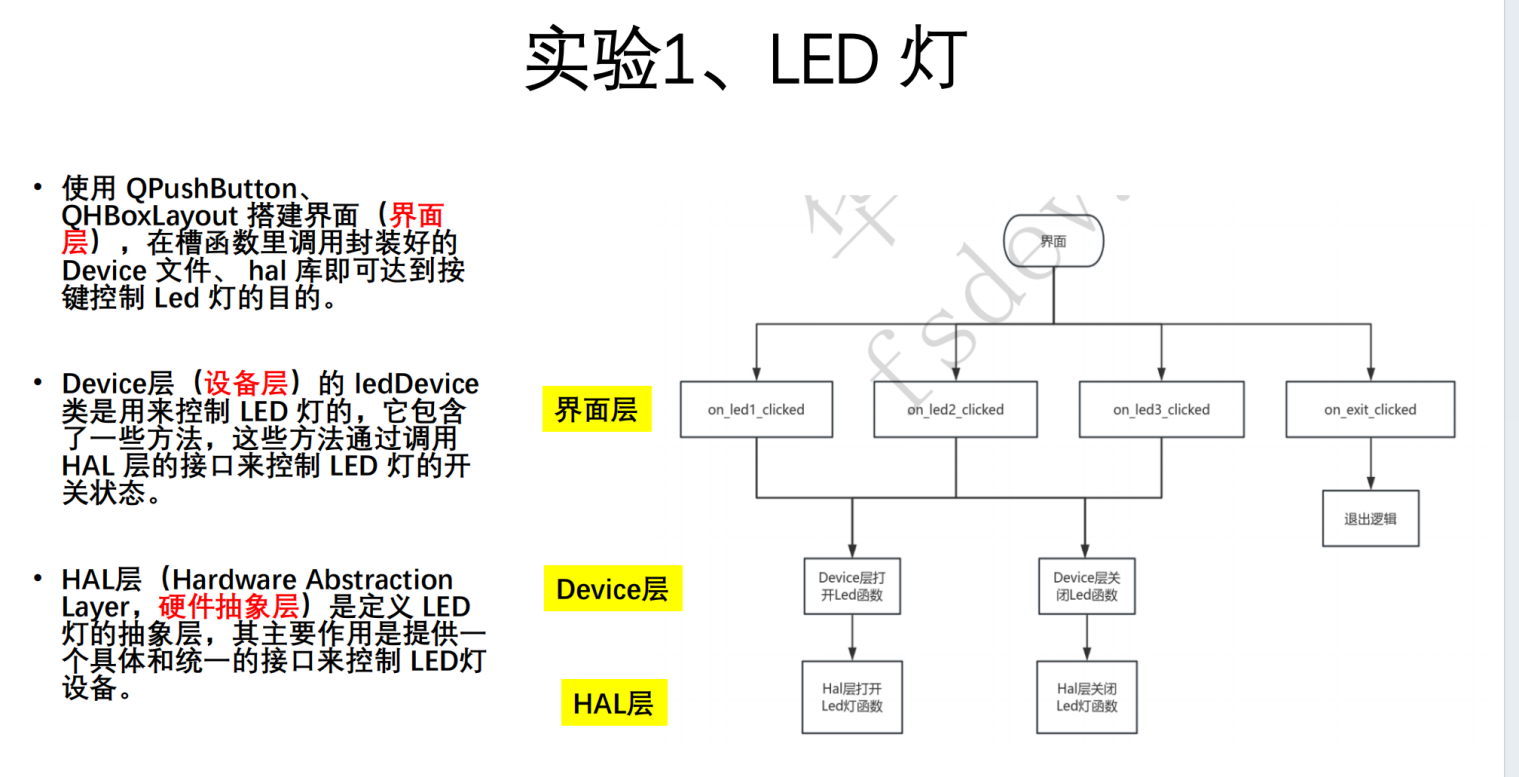
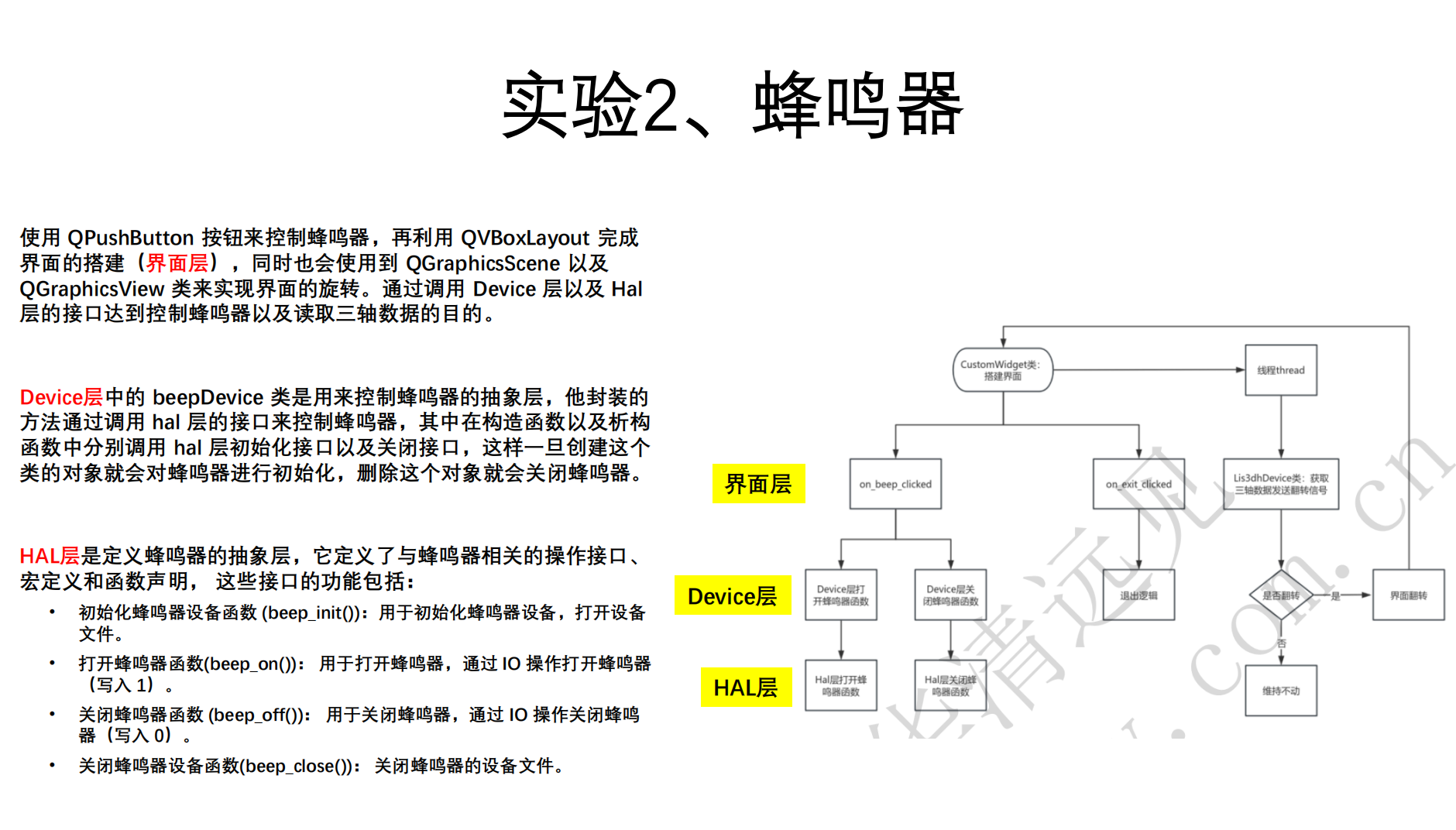
– 使用**mmap**系统调用，可以将**内核空间**的地址映射到**用户空间**

**• void\* mmap(void\* start, size\_t length, int prot, int flags, int fd, off\_t offset);**

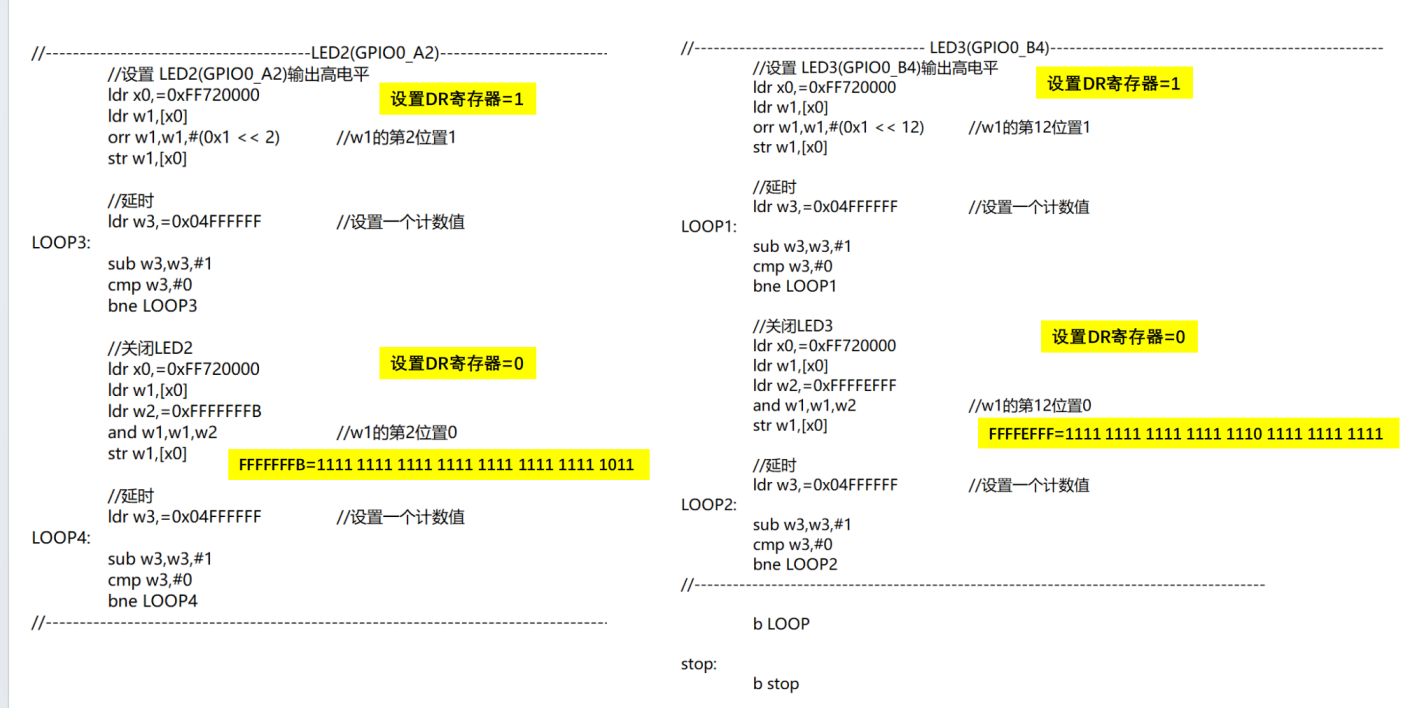
**• int (\* mmap)(struct file\*filp, struct vm\_area\_struct \*vma);**

1. Qt实验程序的流程图框架分成哪三层？（参QT实验课件），并能够将某个实验工程文件夹中的文件对应到各个层中

界面层、Device层、HAL层

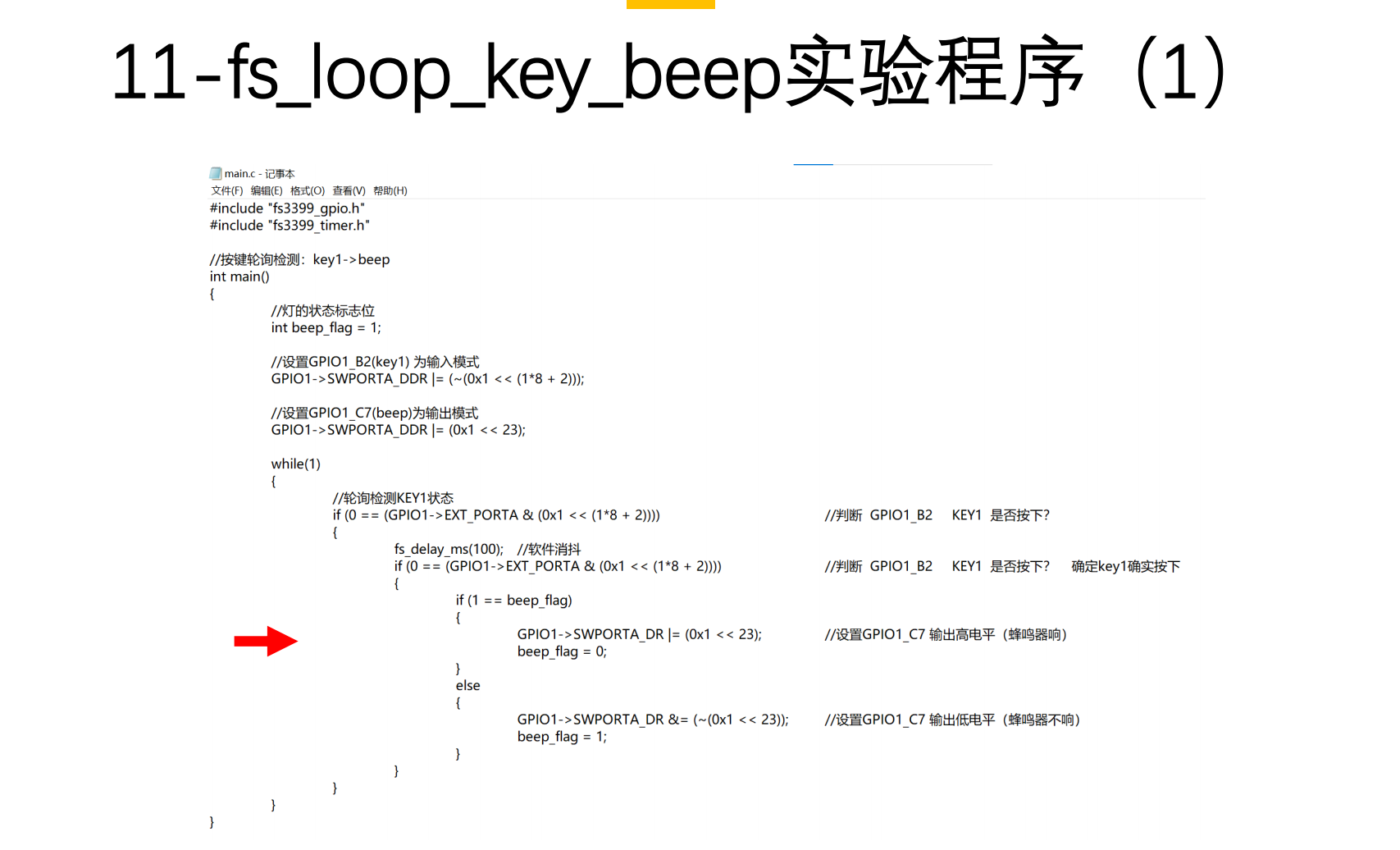
  


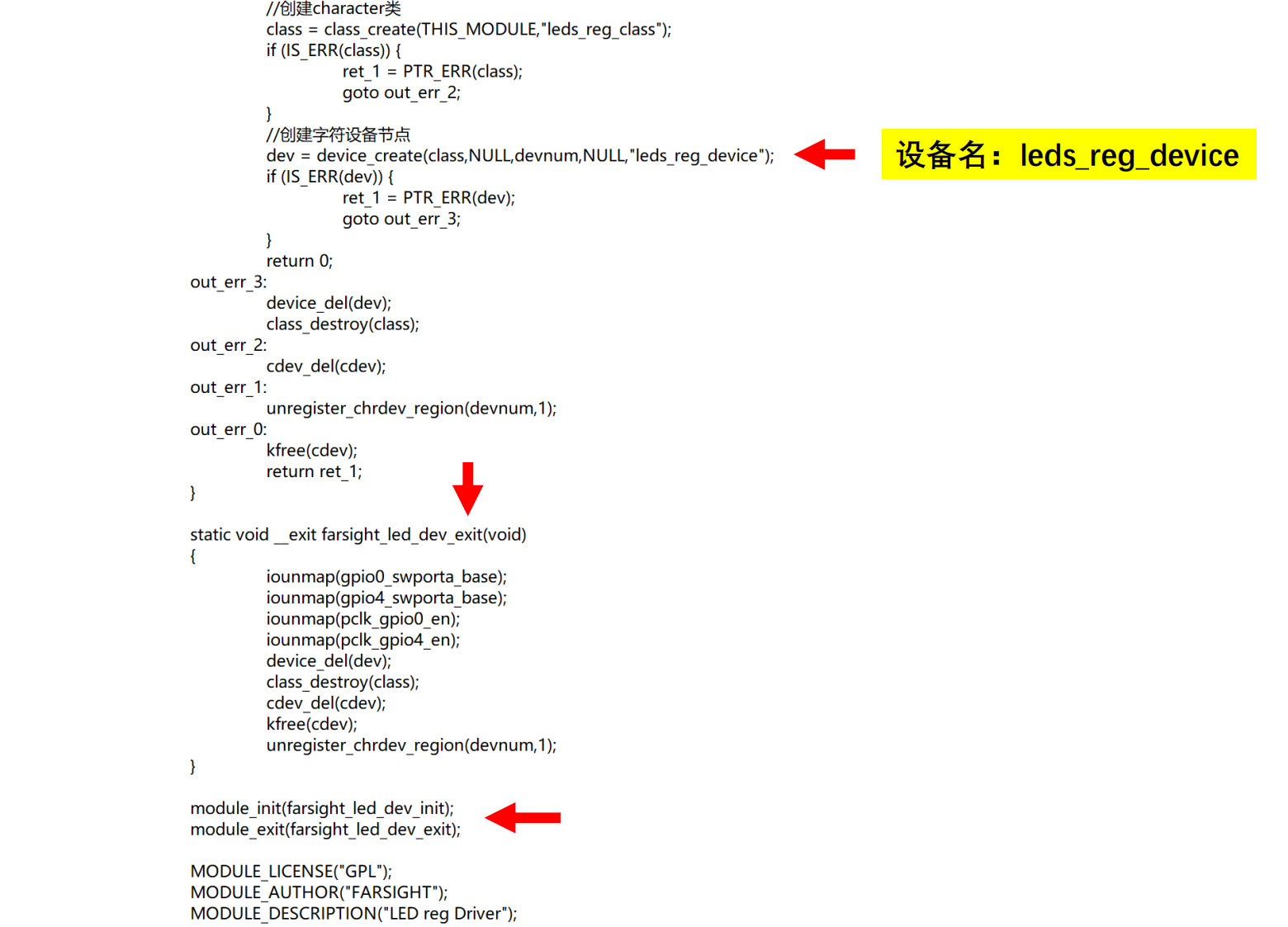
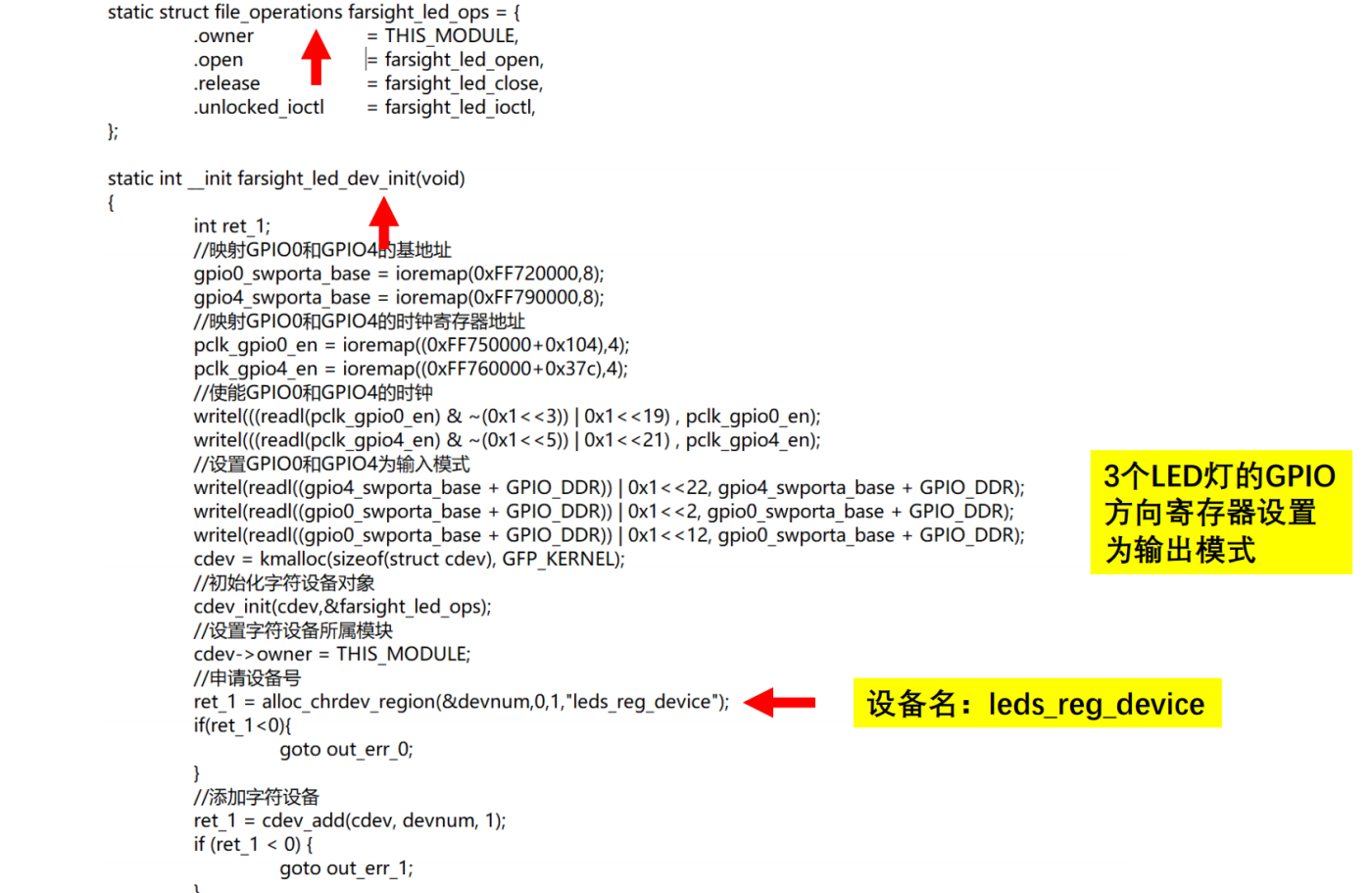
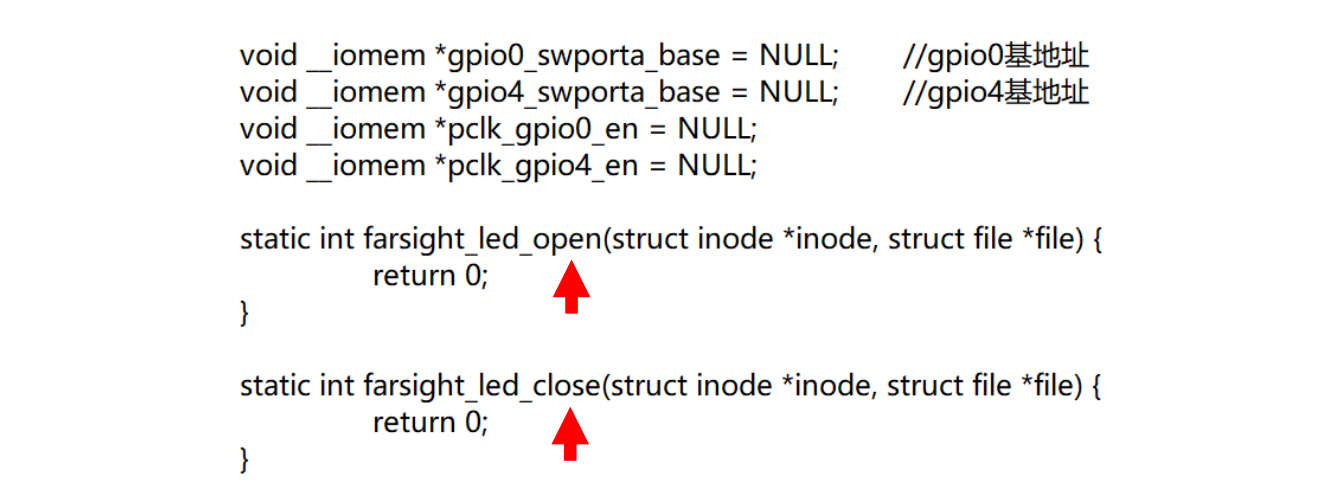
1. ARM裸机实验中的LED（纯汇编）代码，ARM裸机蜂鸣器实验代码，ARM裸机查询方式按键实验代码，基于寄存器控制的LED灯的Linux设备驱动程序代码，按键控制步进电机转动的代码，QT实验中的主窗口代码 （上述代码均为核心代码，如.c, .cpp等）

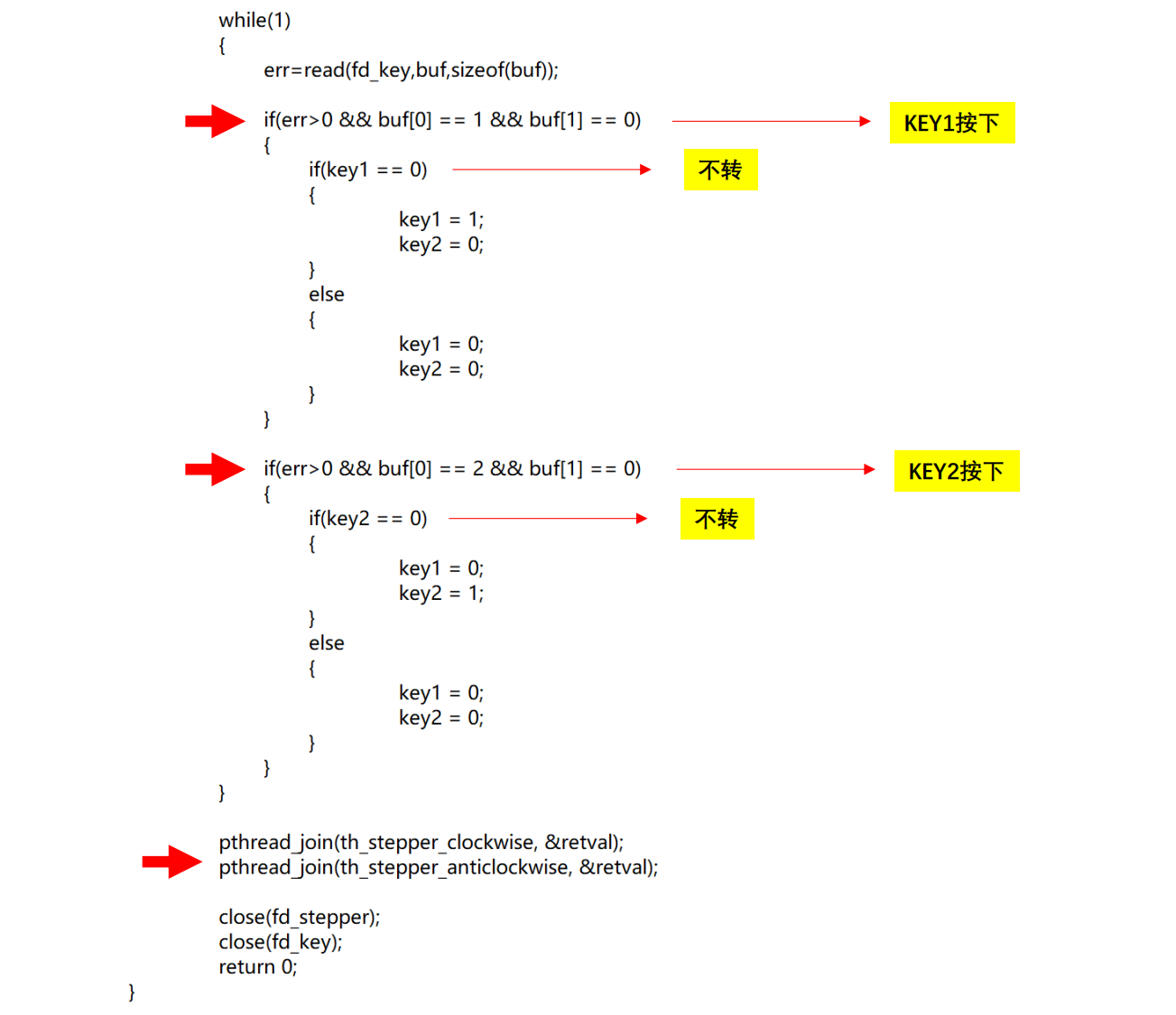
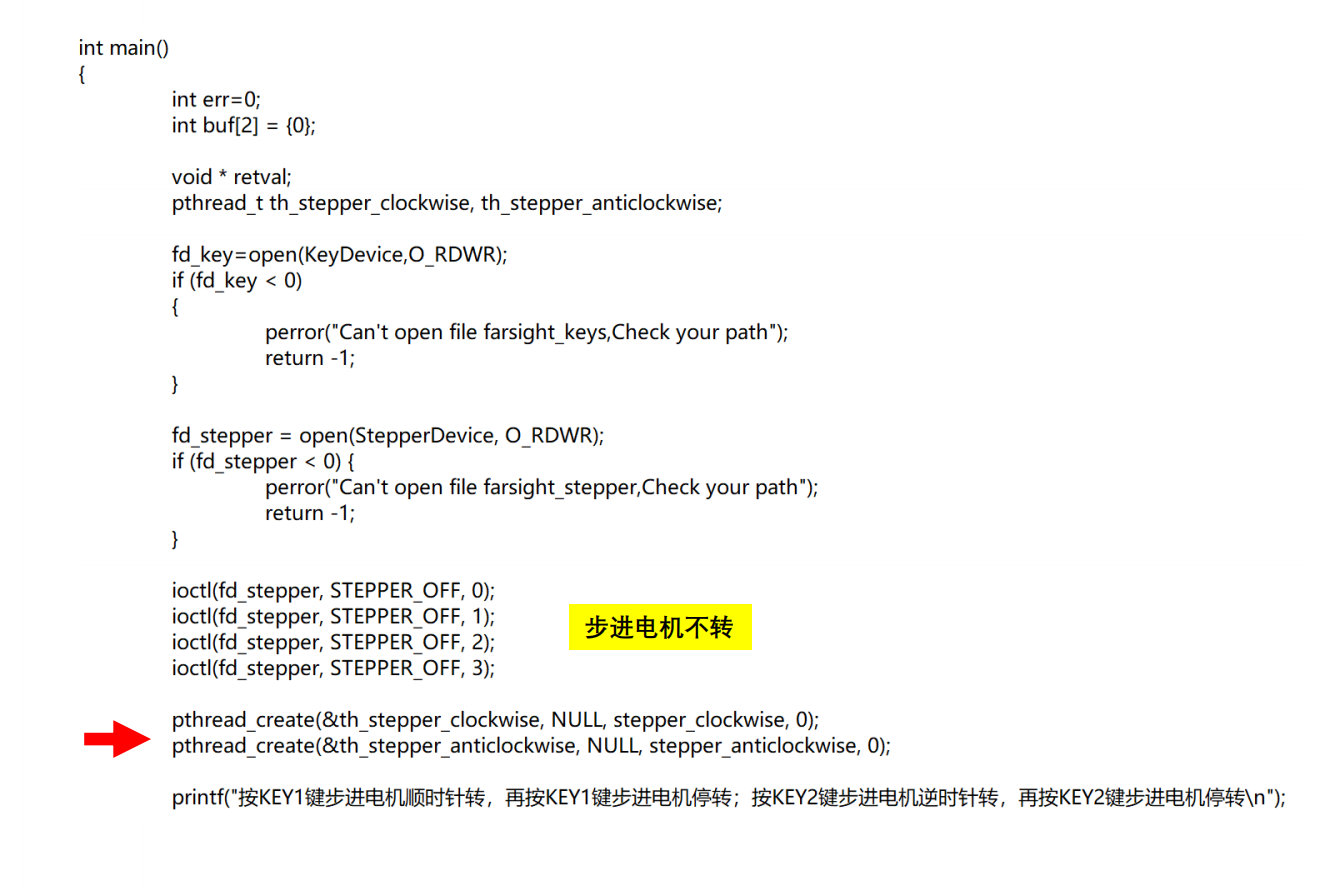
ARM裸机实验中的LED（纯汇编）代码：  


ARM裸机蜂鸣器实验代码  


ARM裸机查询方式按键实验代码



基于寄存器控制的LED灯的Linux设备驱动程序代码  


按键控制步进电机转动的代码  


QT实验中的主窗口代码

ARM处理器的7种运行模式：

① 系统模式（SYS）

② 用户模式（USR）

③ 快速中断模式（FIQ）

④ 管理模式（SVC）

⑤ 数据访问终止模式（ABT）

⑥ 外部中断模式（IRQ）

⑦ 未定义指令终止模式（UND，未定义模式）

特权模式: 除用户模式外

异常模式: 除系统模式、用户模式外