

嵌入式系统

实验与开发环境介绍

计算机学院 刘健培

近期课程安排

时间	地点	安排
10-19（周一）9-11节	教3-217->主楼910	实验介绍，熟悉实验环境
10-26（周一）9-11节	教3-217	理论课
11-02（周一）9-11节	主楼910/920	上机实验一
11-09（周一）9-11节	主楼910/920	上机实验二
11-16（周一）9-11节	教3-217	待定（期中考试？）

实验说明

- 共4个实验，每个10分，占总成绩40%
 - 1体系结构实验+2接口实验+1综合实验
- 每个实验分为基础功能（容易）与扩展功能（较难）
 - 基本功能得分区间0-8，扩展功能得分8-10
- 详细实验要求见ftp上《嵌入式系统实验指导手册.pdf》

实验步骤

- 理解实验要求
- 查阅相关资料：
 - 实验指导书、原理图、数据手册、相关源码等
- 代码实现
- 验收
- 写实验报告（课后），提交实验源代码+实验报告

实验验收

- 地点：主楼907/910机房
- 时间：待定
- 方式：现场演示实验，老师答辩验收
- 提交：源码与电子版实验报告压缩包
 - 提交内容：实验源代码+实验报告
 - 文档命名：实验n-姓名-学号.rar/zip
 - 提交目录：ftp目录\提交实验报告
 - 报告格式：嵌入式系统实验报告模板-实验n-姓名-学号.docx

实验参考资料

- <ftp://es2020:es2020@10.105.240.91>
 - 目录：实验/参考资料
- 重要参考资料
 - Architecture
 - DDI0403E_d_armv7m_arm Armv7-M Architecture Reference Manual.pdf
 - Cortex-M3/4权威指南
 - Processor
 - Cortex-M3技术参考手册 zh.pdf/Cortex-M4技术参考手册_trm_100166_0001_00_en r0p1.pdf

实验参考资料（续）

- MCU

- STM32F4xx中文参考手册.pdf/STM32F4x7-Reference manual.pdf

- Board

- Crotex-M4实验平台指导书.pdf/实验原理图

- Source Code

- 开发板随盘实验源码
- 实验示例代码

- 实验指导书

- 嵌入式系统实验指导书.pdf

实验环境

- 模拟环境：Eclipse+gcc+gdb+Qemu
 - 软件：实验/开发环境/[gnu-mcu-eclipse.rar](#)
 - 绿色，解压缩即可，win7以上64位
 - <https://gnu-mcu-eclipse.github.io/>
- 实物环境：MDK+armcc+ST-LINK+FSM4开发板
 - 软件：MDK523+CMSIS
 - 实验环境搭建与使用参考《Crotex-M4实验平台指导书.pdf》
 - 实验室环境已搭建
 - 可借开发板，注意保管。

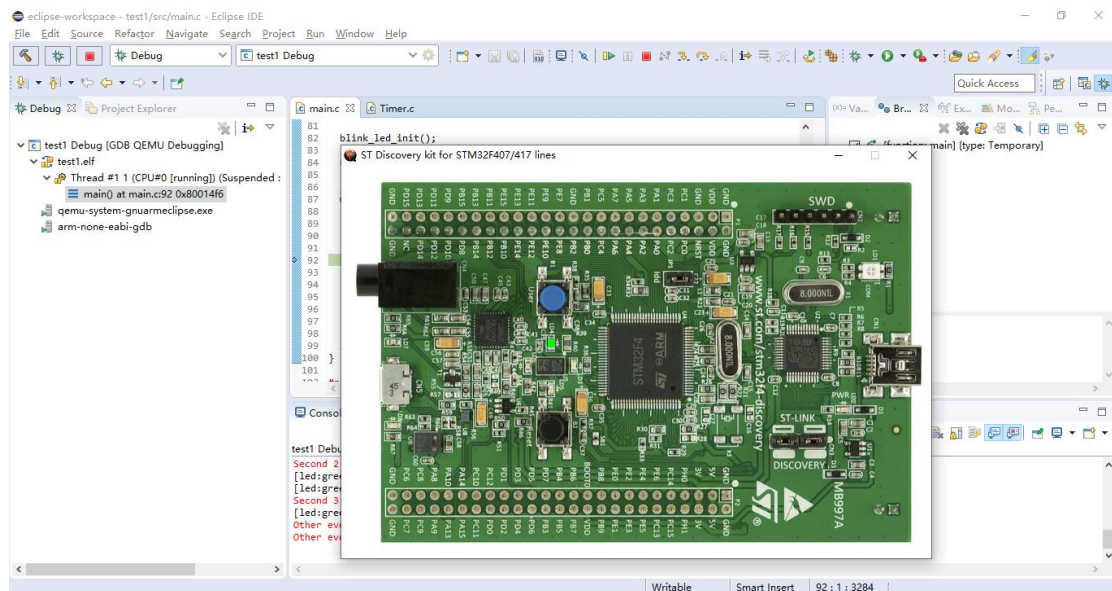
Eclipse模拟开发环境介绍

■ 开发

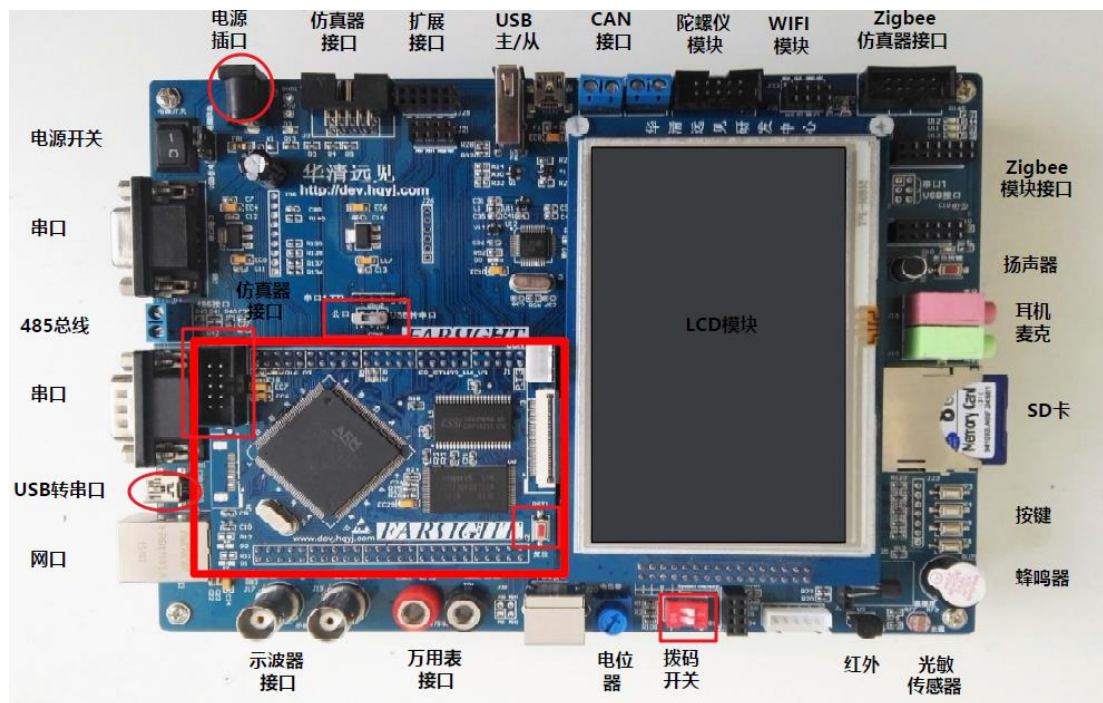
- 打开开发环境
- 创建项目
- 增加源代码
- 编译链接

■ 调试

- 配置调试环境
- 启动调试



FSM4+ST-LINK硬件环境介绍



MDK软件环境介绍

- 相关工具与软件
 - 集成开发环境：RealView MDK
 - Cortex系列处理器软件接口包：CMSIS
 - STM32的图形化配置工具：STM32CUBEMX
- 开发
 - 创建工程
 - 增加代码
 - 编译链接
 - 下载程序
 - 仿真调试

实验1 CPU异常处理与上下文切换

■ 基本功能

- 在主程序中用svc指令触发SVCall异常
- 编写SVCall异常处理程序，打印异常发生前的处理器现场状态（即寄存器R0-R15、xPSR）以及异常发生后发生变化的寄存器（R13/SP、R14/LR/EXC_RETURN、R15/PC、xPSR、CONTROL），据此分析异常发生前后处理器分别处于哪种模式（handler or thread）、使用哪种栈（MSP or PSP）、特权等级（特权与非特权）

■ 扩展功能

- 使用svc异常模拟系统调用，实现CPU上下文切换功能
- 如：func1->context_switch->func2->context_switch->func3

实验2 GPIO与系统状态

■ 基本功能

- 编写程序控制led灯的亮灭（或者控制板上蜂鸣器的出声），输出以字母、数字、空格组成的字符串的摩斯码（以“**Hello Cortex-M4**”为测试用例）
- 特殊要求：不能使用CMSIS库函数操作led灯（蜂鸣器），需用代码直接操作GPIO的寄存器

■ 扩展功能

- 使用按键控制系统状态，**LED**灯显示系统状态
 - 按键K3短按：待机，系统进入低功耗模拟
 - 按键K4长按：系统复位
 - 按键K5双击：led灯闪烁
 - 按键K6长按：随着按动时长，4个led灯依次点亮

实验3 UART与SHELL

- 基本功能
 - 使用uart1控制4个led灯的亮灭闪烁
 - 命令格式: `led n on/off/flash`
- 扩展功能
 - 实现基于uart的简单shell。支持以下功能:
 - `help/?` – 显示命令帮助
 - `Tab`键 – 补全或者显示可选命令
 - `up/down`键 – 切换命令历史
 - `Backspace`键 – 删除字符
 - `d addr n` – 打印addr地址处的n个字节
 - `led n on/off/flash` – 控制led

实验4 中断与多任务调度

- 在实验1、2、3的基础上扩展中断功能，并实现多任务调度。
 - 将实验1的上下文切换功能扩展为多任务调度功能，并通过定时器中断实现时间片轮转调度。（相当于一个“微型”的嵌入式操作系统内核。）
 - 将实验2的按键读取从轮询方式扩展为中断方式，其余功能不变。
 - 将实验3的串口字符收发从轮询方式扩展为中断方式，其余功能不变。
 - 将实验2、实验3的功能实现为本实验中的任务，实现多任务并发运行。
 - 串口shell输入输出
 - Led闪烁
 - 扫描按键