|  |  |  |  |
| --- | --- | --- | --- |
| 姓名：刘赟 | 专业：物联网工程 | 班级：20181191班 | 学号：2018119117 |
| 科目：嵌入式应用系统开发 | | 实验日期：2020/7/6 | |
| 实验题目：任务管理 | | | |
| 【实验目的】   1. Huawei LiteOS 内核提供任务的创建、删除、延迟、挂起、恢复等功能，以及锁定和解锁任务调度，支持任务按优先级高低的抢占调度及同优先级时间片轮转调度。 | | | |
| 【实验内容】  1.实验设备   硬件：PC机一台   小熊派开发板一套   软件：XP/win7、8、8.1、10 系统，IoT Studio，git 客户端 [sourcetree](https://www.sourcetreeapp.com/)，课程使用 sourcetree 和 [github](http://www.github.com/) 作为项目管理和协作系统。  2.实验内容和要求  本实验中将创建两个任务，一个低优先级任务task1，一个高优先级任务task2，两个任务都会每隔2s在串口打印自己的任务id号，在串口终端中观察两个任务的运行情况。 | | | |
| 【实验结果】   1. 代码实现   osal\_task\_demo.c  /\* 使用osal接口需要包含该头文件 \*/  #include <osal.h>  /\* 任务优先级宏定义（shell任务的优先级为10） \*/  #define USER\_TASK1\_PRI  12  //低优先级  #define USER\_TASK2\_PRI  11  //高优先级  /\* 任务ID \*/  uint32\_t user\_task1\_id = 0;  uint32\_t user\_task2\_id = 0;  /\* 任务task1入口函数 \*/  static int user\_task1\_entry()  {      int n = 0;      /\* 每隔2s在串口打印一次，打印5次后主动结束 \*/      for(n = 0; n < 5; n++)      {          printf("task1: my task id is %ld, n = %d!\r\n", user\_task1\_id, n);          /\* 任务主动挂起2s \*/          osal\_task\_sleep(2\*1000);      }      printf("user task 1 exit!\r\n");      /\* 任务结束 \*/      return 0;  }  /\* 任务task2入口函数 \*/  static int user\_task2\_entry()  {      /\* 每隔2s在串口打印一次，不结束 \*/      while (1)      {          printf("task 2: my task id is %ld!\r\n", user\_task2\_id);          /\* 任务主动挂起2s \*/          osal\_task\_sleep(2\*1000);      }  }  /\* 标准demo启动函数，函数名不要修改，否则会影响下一步实验 \*/  int standard\_app\_demo\_main()  {      /\* 创建任务task1 \*/      user\_task1\_id = osal\_task\_create("user\_task1",user\_task1\_entry,NULL,0x400,NULL,USER\_TASK1\_PRI);      /\* 创建任务task2 \*/      user\_task2\_id = osal\_task\_create("user\_task2",user\_task2\_entry,NULL,0x400,NULL,USER\_TASK2\_PRI);      return 0;  }  将我们编写的osal\_task\_demo.c文件添加到makefile中  #example for osal\_task\_demo      ifeq ($(CONFIG\_USER\_DEMO), "osal\_task\_demo")          user\_demo\_src  = ${wildcard $(TOP\_DIR)/targets/STM32L431\_BearPi/Demos/osal\_kernel\_demo/osal\_task\_demo.c}          user\_demo\_defs = -D CONFIG\_OSAL\_TASK\_DEMO\_ENABLE=1  endif   1. 实验现象   61ae4fd8ab8303a4b559ad6217164c1  c0f00b2d1df0ef3df607b387f5909f2  串口现象：  linkmain:V1.2.1 AT 13:20:47 ON July 6 2021  WELCOME TO IOT\_LINK SHELL  LiteOS:/>task 2: my task id is 5!  task1: my task id is 4, n = 0!  task 2: my task id is 5!  task1: my task id is 4, n = 1!  task 2: my task id is 5!  task1: my task id is 4, n = 2!  task 2: my task id is 5!  task1: my task id is 4, n = 3!  task 2: my task id is 5!  task1: my task id is 4, n = 4!  task 2: my task id is 5! | | | |
| **【**教师评语和成绩**】**  **成绩：** **指导教师：** **日期：** | | | |