|  |  |  |  |
| --- | --- | --- | --- |
| 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：内存管理 | | | |
| 【实验目的】   1. Huawei LiteOS 的内存管理模块通过对内存的申请/释放操作，来管理用户和OS对内存的使用，使内存的利用率和使用效率达到最优，同时最大限度地解决系统的内存碎片问题。 | | | |
| 【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  本实验中将创建一个任务，从最小字节开始，不停的申请分配内存，释放分配的内存，直到申请失败，串口终端中观察可以申请到的最大字节。 | | | |
| 【实验结果】   1. 代码实现   osal\_mem\_demo.c  /\* 使用osal接口需要包含该头文件 \*/  #include <osal.h>  /\* 任务入口函数 \*/  static int mem\_access\_task\_entry()  {      uint32\_t i = 0;     //循环变量      size\_t mem\_size;    //申请的内存块大小      uint8\_t\* mem\_ptr = NULL;    //内存块指针      while (1)      {          /\* 每次循环将申请内存的大小扩大一倍 \*/          mem\_size = 1 << i++;          /\* 尝试申请分配内存 \*/          mem\_ptr = osal\_malloc(mem\_size);          /\* 判断是否申请成功 \*/          if(mem\_ptr != NULL)          {              /\* 申请成功，打印信息 \*/              printf("access %d bytes memory success!\r\n", mem\_size);              /\* 释放申请的内存，便于下次申请 \*/              osal\_free(mem\_ptr);              /\* 将内存块指针置为NULL，避免称为野指针 \*/              mem\_ptr = NULL;              printf("free memory success!\r\n");          }          else          {              /\* 申请失败，打印信息，任务结束 \*/              printf("access %d bytes memory failed!\r\n", mem\_size);              return 0;          }      }  }  /\* 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，任务优先级为11，shell任务的优先级为10 \*/      osal\_task\_create("mem\_access\_task",mem\_access\_task\_entry,NULL,0x400,NULL,11);      return 0;  }  将编写的osal\_mem\_demo.c文件添加到makefile中，直接修改Demo文件夹下的user\_demo.mk配置文件  #example for osal\_mem\_demo  ifeq ($(CONFIG\_USER\_DEMO), "osal\_mem\_demo")      user\_demo\_src  = ${wildcard $(TOP\_DIR)/targets/STM32L431\_BearPi/Demos/osal\_kernel\_demo/osal\_mem\_demo.c}  endif  4.实验现象  1f5fb95f6c3aa2efa6e84830e811f11  串口现象：  linkmain:V1.2.1 AT 14:20:47 ON Nov 28 2019  WELCOME TO IOT\_LINK SHELL  LiteOS:/>access 1 bytes memory success!  free memory success!  access 2 bytes memory success!  free memory success!  access 4 bytes memory success!  free memory success!  access 8 bytes memory success!  free memory success!  access 16 bytes memory success!  free memory success!  access 32 bytes memory success!  free memory success!  access 64 bytes memory success!  free memory success!  access 128 bytes memory success! | | | |
| **【**教师评语和成绩**】**  **成绩：** **指导教师：** **日期：** | | | |