

从所有教程的词条中查询...

首页 > 慕课教程 > 物联网/嵌入式工程师 > 8.1 线程创建

全部开发者教程

4.2 进程间通讯-有名管道

4.3 进程间通讯-信号 (一)

4.4 进程间通讯-信号(二)

4.5 进程间通讯-信号 (三)

4.6 进程间通讯-消息队列(一)

4.7 进程间通讯-消息队列(二)

4.8 进程间通讯-共享内存(一)

4.9 进程间通讯-共享内存(二)

5.1 进程间通讯-信号量(一)

5.2 进程间通讯-信号量(二)

5.3 进程间通讯-信号量同步

6.1 线程的概念

7.1 线程相关命令

8.1 线程创建

8.2 线程的退出、等待与分离

8.3 创建多个线程



大白老师 · 更新于 2022-10-31

◀ 上一节 7.1 线程相关命令 8.2 线程的退出... 下一节 ▶

## 一、线程创建

- 创建线程调用 `pthread_create` 函数

函数头文件 `#include <pthread.h>`

函数原型

```
int pthread_create(pthread_t *thread, const pthread_attr_t *attr,
void *(*start_routine) (void *), void *arg);
```

函数功能 创建一个子线程

函数参数

`thread` : 线程 ID 变量指针

`attr` : 线程属性,默认属性可设置为 `NULL`

`start_routine` : 线程执行函数

`arg` : 线程执行函数的参数

函数返回值

- 成功 : 返回 0
- 失败 : 返回 错误码
- 注意 :
  - 一旦子线程创建成功,则会被独立调度执行,并且与其他线程 并发执行
  - 在编译时需要链接 `-lpthread`
- 示例
  - 创建一个线程,并打印线程 ID

<> 代码块

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <pthread.h>
5
6
7
8 void *do_thread(void *arg)
9 {
10     printf("Thread start.\n");
```

📝 意见反馈

📖 收藏教程

🔖 标记书签

```
12
13  int main(void)
14  {
15      int err;
16      pthread_t tid = 0;
17
18      err = pthread_create(&tid,NULL,do_thread,NULL);
19      if (err != 0){
20          fprintf(stderr,"[ERROR] pthread_create : < %s >\n",strerror(err));
21          exit(EXIT_FAILURE);
22      }
23
24      printf(" tid = %ld\n",tid);
25
26      return 0;
27  }
```

- 问题：
  - 上述程序执行的结果只打印了 **tid**,子线程没有执行
- 原因：
  - 子线程还没有来得及执行,主线程已经执行结束,导致所有其他子线程都必须要结束
- 解决:
  - 保证主线程不先于子线程结束
  - 
  - 
  - 
  - 练习
  - 创建两个子线程，并打印两个子线程的 tid







