

3.2 进程多任务_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 3.2 进程多任务涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

全部开发者教程

物联网 / 嵌入式工程师

第 24 周 stm32 芯片 – 智能硬件项目实战与企业笔试

未命名节

第 25 周 大厂必备 – linux 内核与文件系统移植

未命名节

第 26 周 嵌入式开发 – 系统移植 – bootloader、yocto

未命名节

第 27 周 嵌入式底层核心技能 – Linux 设备驱动初级

未命名节

第 28 周 嵌入式底层核心技能 – Linux 设备驱动中级

未命名节

第 29 周 嵌入式底层核心技能 – Linux 设备驱动高级 1

未命名节

第 30 周 嵌入式底层核心技能 – Linux 设备驱动高级 2

未命名节

第 31 周 嵌入式人工智能必备 – Python

未命名节

第 32 周 智能家居项目实战之客户端功能开发

未命名节

第 33 周 智能家居项目实战之网关端功能开发

未命名节

第 34 周 智能家居项目实战之设备端功能开发

未命名节

第 35 周 物联网 / 嵌入式项目答辩和就业指导

未命名节

第 36 周 独立开发阶段 – 三大热门领域项目

未命名节

- 使用 fork() 函数之后，会创建子进程，fork() 之后的代码会在父子进程中都执行一遍
 - 如果父子进程执行相同的任务，则正常执行
 - 如果父子进程执行不同的任务，则需要利用 fork() 函数返回值

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>

int main(void)
{
    pid_t cpid;

    cpid = fork();
    if (cpid == -1){
        perror("[ERROR] fork()");
        exit(EXIT_FAILURE);
    }else if(cpid == 0){
        printf("Child process task.\n");
        exit(EXIT_SUCCESS);
    }else if (cpid > 0){
        printf("Parent process task.\n");
    }

    printf("Child and Process Process task.\n");

    return 0;
}
```

- 在创建多个进程时, 最主要的原则为 由父进程统一创建, 统一管理, 不能进行递归创建

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>
#include <sys/wait.h>

int main(void)
{
    int cpid;

    cpid = fork();
    if (cpid == -1){
        perror("fork(): ");
        exit(EXIT_FAILURE);
    }else if (cpid == 0){

        printf("The child process < %d > running...\n",getpid());
        sleep(2);
        printf("The child process < %d > has exited\n",getpid());
        exit(EXIT_SUCCESS);

    }else if (cpid > 0){
        cpid = fork();
        if (cpid == -1){
            perror("fork(): ");
        }else if (cpid == 0){

            printf("The child process < %d > running...\n",getpid());
            sleep(3);
            printf("The child process < %d > has exited\n",getpid());
            exit(EXIT_SUCCESS);

        }else if (cpid > 0){

        }

    }

    return 0;
}
```

全文完

本文由 简悦 SimpRead 优化, 用以提升阅读体验