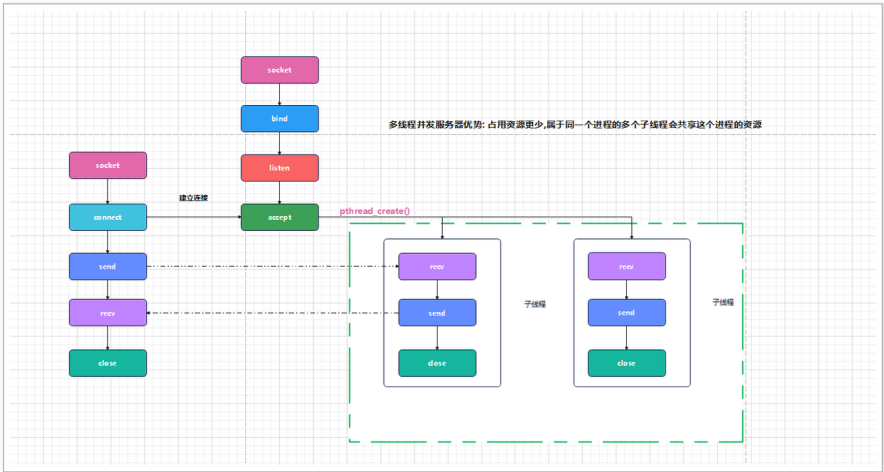


## 4.2 tcp 并发服务器 - 多线程\_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 4.2 tcp 并发服务器 – 多线程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。

- 多线程并发服务器的实现与多进程类似，只不过将创建进程修改成创建线程即可



- 多线程并发服务器实现如下:
  - 在服务器建立与客户端的连接后, 创建线程, 并实现线程执行函数即可
    - 线程执行函数的实现如下:

```
void *do_client(void *arg)
{
    char buffer[1024] = {0};
    ssize_t sbytes, rbytes;
    int cfd = *(int *)arg;

    memset(buffer, 0, sizeof(buffer));
    rbytes = recv(cfd, buffer, sizeof(buffer), 0);
    if (rbytes == -1){
        perror("recv(): ");

        pthread_exit(NULL);
    }else if (rbytes == 0){
        printf("The client is offline.\n");
        pthread_exit(NULL);
    }else if (rbytes > 0){
        sbytes = send(cfd, buffer, sizeof(buffer), 0);
        if (sbytes == -1){
            perror("[ERROR] send(): ");

            pthread_exit(NULL);
        }
    }
    pthread_exit(NULL);
}
```
    - 创建线程, 并让线程分离
    - ```
for(;;){
    bzero(&cli_addr, sizeof(struct sockaddr));
    cfd = accept(sfd, (struct sockaddr *)&cli_addr, &len);
    if (cfd == -1){
        perror("[ERROR] accept(): ");
        exit(EXIT_FAILURE);
    }
}
```

```
printf("ip : %s,port : %d\n",inet_ntoa(cli_addr.sin_addr),ntohs(c
err = pthread_create(&tid,NULL,do_client,(void *)&cfd) ;
if (err != 0){
    fprintf(stderr,"[ERROR] pthread_create(): %s\n",strerror(err)
    exit(EXIT_FAILURE);
}

pthread_detach(tid);
}
```

- 
- 
- 
- 完成基于多线程 echo 服务器

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

