NuttX 操作系统用户手册

By Gregory Nutt

Last Updated: September 28, 2013

翻译：cumtlk

版本：V1.0

# 1.0 介绍

本文档从固件开者的角度提供了NuttX 实时操作系统一般的使用信息。

## 1.1 文档概述

本用户手册分为三个部分，外加一个索引：

* 部分1.0，介绍：本节提供了NuttX用户手册的概述。
* 部分2.0，操作系统接口：本节详细介绍了NuttX提供的程序接口。本节被划分为几个段落来描述操作系统接口的不同部分。
* 部分3.0，操作系统数据结构：本节介绍了在NuttX接口中使用的数据结构。

## 1.2 目标读者与使用范围

目标读者是在NuttX上开发应用的固件开发者。具体来说，这个文档只是在说明应用开发者可以使用的NuttX 实时操作系统的API。因此，这个文档并没有集中在NuttX的组织或实现的任何技术细节。那些技术细节在NuttX Porting Guide中提供。应用开发者也需要关于配置和编译NuttX的信息。那些信息也可以在NuttX Porting Guide中找到。

# 2.0 操作系统接口

本节介绍了NuttX操作系统每一个C可调用的接口。

**函数原型：**提供接口函数的C原型。

**说明：**讨论了接口函数执行的操作。

**输入参数：**列出了所有的输入参数，并对每个参数作了简要的介绍。

**返回值：** 接口函数返回的每一个可能的值都列了出来。返回值作为副作用（通过指针输入参数或者全局编领）将在接口函数的描述中加以声明。

**假设与限制：**通过接口函数产生的任何不寻常的假设或者使用接口函数的任何非显而易见的局限性将会在这里说明。

**POSIX相容性：**NuttX接口与它相应的POSIX接口之间的任何重要的差异都会在这里指出。

**注：**为了实现NuttX接口函数的独立的名称空间，函数名和类型之间的差异可以预期的，在这些段落中不会被定义为差异。

## 2.1 任务控制接口

**任务：**NuttX是一个平面地址操作系统，因此，它不支持这样的进程，比如说linux中的方式。NuttX仅仅支持运行在相同地址空间的简单的线程。然而，编程模型使任务和pthreads之间有区别。

* 任务是具有一定程度独立性的线程。
* Pthreads 共享某些资源。

**文件描述符和流：**这尤其适应于，在打开的文件描述符和数据流区域。当一个任务使用本节的接口启动时，它将会被创建，并最多有三个文件打开。

如果CONFIG\_DEV\_CONSOLE被定义，前三个文件描述符（对应标准输入、标准输出、标准错误）将会被复制给新任务。由于这些文件的文件描述符被复制。一旦这些文件描述符被复制，孩子任务可以足有关闭它们或者以任何方式操作它们，而不会影响父任务。在一个任务中，与文件相关的操作（打开、关闭等等）不会对其他任务造成影响。一旦这三个文件描述符被复制了，它也有可能进行某种程度的重定向。

pthreads，另一方面，将始终与符线程共享文件描述符。在这种情况下，文件操作将会只影响所有从同一父线程启动的线程。

**执行文件系统中的程序：**NuttX还为驻留在文件系统单独建立程序的执行提供了内部接口。然而，这些内部接口是非标准的，而且在NuttX二进制加载器和NXFLAT被文档化。

**任务控制接口：**下面的任务控制接口就是由NuttX提供的：

灵感来自于VxWorks接口非标准任务控制接口：