

迅为电子Linux驱动教程

—设备注册

北京迅为电子有限公司





本节目标

- 关于注册设备的一点说明
- 注册设备



关于注册设备的一点说明

- 早先的Linux会使用单独的文件注册设备，现在大多是使用引入的虚拟平台，使用虚拟平台来注册设备会容易很多
- 如果大家在网上看到大段的注册设备的代码，可简单了解一下，知道有这么个东西就行，不用去深究。不要去学习“屠龙技”，现在基本都是直接在平台文件中注册设备



注册设备

- 注册设备使用结构体platform_device，该结构体在头文件“vim include/linux/platform_device.h”中。头文件中也有注册设备和卸载设备的函数，了解即可。

```
struct platform_device {
    const char    * name;
    int           id;
    struct device  dev;
    u32           num_resources;
    struct resource * resource;

    const struct platform_device_id *id_entry;

    /* MFD cell pointer */
    struct mfd_cell *mfd_cell;

    /* arch specific additions */
    struct pdev_archdata archdata;
};

#define platform_get_device_id(pdev)    ((pdev)->id_entry)

#define to_platform_device(x) container_of((x), struct platform_device, dev)

extern int platform_device_register(struct platform_device *);
extern void platform_device_unregister(struct platform_device *);
11,1 4%
```



注册设备

```
struct platform_device {  
    const char    * name;//设备名称，在 sys/devices 会显示  
    int           id;//设备 id，用于插入总线并且具有相同 name 的设备编号，
```

如果只有一个设备那么-1

```
    struct device    dev;//结构体中内嵌的 device 结构体  
    u32              num_resources;//设备使用资源的数量  
    struct resource * resource;//设备使用的资源数组
```

```
    const struct platform_device_id *id_entry;
```

```
    /* MFD cell pointer */  
    struct mfd_cell *mfd_cell;
```

```
    /* arch specific additions */  
    struct pdev_archdata    archdata;
```



注册设备

- 操作过程

- 注册设备。将设备结构体放到平台文件中，会自动注册设备，不用去调用注册设备的函数。

vi arch/arm/mach-exynos/mach-itop4412.c 仿照led写一个
相当于定义了一个结构体 platform_device s3c_device_hello_ctl

- 在Kconfig文件中添加编译HELLO设备的宏定义（前面教程中已经添加）

vi drivers/char/Kconfig
仿照led写一个HELLO的宏定义
相当于定义了一个宏定义

- 配置menuconfig中的HELLO宏定义，生成新的.config文件

- 生成新的zImage，烧写

- 注册完之后在虚拟平台总线下可以查到注册的设备

- ls /sys/devices/platform/



谢谢！