

KelpLdm

Linux Driver Model

2005 KELP Open Seminar

-- 2005-11-01 01:01:10

[1](#)

[2](#)

[3](#)

[3.1](#) platform API

[3.1.1](#) struct platform_device

[3.1.2](#) platform_device_register

[3.1.3](#) platform_device_unregister

[4](#)

[4.1](#) device_driver API

[4.1.1](#) struct device_driver

[4.1.2](#) driver_register

[4.1.3](#) driver_unregister

[4.1.4](#) dev_set_drvdata

[4.1.5](#) dev_get_drvdata

[5](#) device vs driver

[6](#)

[6.1](#) API 가

[6.2](#) API

[6.2.1](#) struct file_operations

[6.2.2](#) register_chrdev

[6.2.3](#) unregister_chrdev

[6.2.4](#) struct cdev

[6.2.5](#) cdev_init

[6.2.6](#) cdev_add

[6.2.7](#) cdev_del

[6.2.8](#) register_chrdev_region

[6.2.9](#) alloc_chrdev_region

[6.2.10](#) unregister_chrdev_region

[7](#)

[7.1](#) MKDEV

[7.2](#) MAJOR

[7.3](#) MINOR

[8](#)

[8.1](#) subsystem, kset, ktype, kobj

[9](#) kref

[9.1](#) kref API

[9.1.1](#) struct kref

[9.1.2](#) kref_init

[9.1.3](#) kref_get

[9.1.4 kref_put](#)[10 kobject](#)[10.1 kobject API](#)[10.1.1 kobject_set_name](#)[10.1.2 kobject_name](#)[10.1.3 kobject_init](#)[10.1.4 kobject_cleanup](#)[10.1.5 kobject_add](#)[10.1.6 kobject_del](#)[10.1.7 kobject_register](#)[10.1.8 kobject_unregister](#)[10.1.9 kobject_get](#)[10.1.10 kobject_put](#)[10.1.11 kobject_get_path](#)[11 kset](#)[11.1 kset API](#)[11.1.1 struct kset](#)[11.1.2 kset_init](#)[11.1.3 kset_add](#)[11.1.4 kset_register](#)[11.1.5 kset_unregister](#)[11.1.6 to_kset](#)[11.1.7 kset_get](#)[11.1.8 kset_put](#)[12 ktype](#)[13 sysfs](#)[13.1 /sys](#)[13.2 sysfs API](#)[13.2.1 sysfs_create_dir](#)[13.2.2 sysfs_remove_dir](#)[13.2.3 sysfs_rename_dir](#)[13.2.4 sysfs_create_file](#)[13.2.5 sysfs_update_file](#)[13.2.6 sysfs_remove_file](#)[13.2.7 sysfs_create_link](#)[13.2.8 sysfs_remove_link](#)[13.2.9 sysfs_create_bin_file](#)[13.2.10 sysfs_remove_bin_file](#)[13.2.11 struct attribute_group](#)[13.2.12 sysfs_create_group](#)[13.2.13 sysfs_remove_group](#)[13.3](#)[14 subsystem](#)[14.1 subsystem](#)[14.2 subsystem kset](#)[14.3 subsystem on sysfs](#)[14.4 subsystem API](#)

- [14.4.1](#) decl_subsys
- [14.4.2](#) subsystem_init
- [14.4.3](#) subsystem_register
- [14.4.4](#) subsystem_unregister
- [14.4.5](#) subsystem_get
- [14.4.6](#) subsystem_put

[15](#) kobj_map

[15.1](#) kobj_map API

- [15.1.1](#) kobj_probe_t
- [15.1.2](#) struct kobj_map
- [15.1.3](#) kobj_map
- [15.1.4](#) kobj_unmap
- [15.1.5](#) kobj_lookup
- [15.1.6](#) kobj_map_init

[16](#) bus

[16.1](#) bus on sysfs

[16.2](#) bus API

- [16.2.1](#) bus_register
- [16.2.2](#) bus_unregister
- [16.2.3](#) bus_for_each_dev
- [16.2.4](#) bus_for_each_drv
- [16.2.5](#) BUS_ATTR
- [16.2.6](#) bus_create_file
- [16.2.7](#) bus_remove_file

[17](#) device

[17.1](#) device on sysfs

[17.2](#) device API

- [17.2.1](#) struct device
- [17.2.2](#) device_register
- [17.2.3](#) device_unregister
- [17.2.4](#) DEVICE_ATTR
- [17.2.5](#) device_create_file
- [17.2.6](#) device_remove_file

[18](#) driver

[18.1](#) driver API

- [18.1.1](#) driver_register
- [18.1.2](#) driver_unregister
- [18.1.3](#) DRIVER_ATTR
- [18.1.4](#) driver_create_file
- [18.1.5](#) driver_remove_file

[19](#) class

[19.1](#) class on sysfs

[19.2](#) class

- [19.2.1](#) struct class
- [19.2.2](#) class_simple_create
- [19.2.3](#) class_simple_destroy
- [19.2.4](#) class_simple_device_add
- [19.2.5](#) class_simple_device_remove

- [19.2.6 class_create](#)
- [19.2.7 class_destroy](#)
- [19.2.8 class_device_create](#)
- [19.2.9 class_device_destroy](#)
- [19.2.10 class_register](#)
- [19.2.11 class_unregister](#)
- [19.2.12 class_get](#)
- [19.2.13 class_put](#)
- [19.2.14 struct class_attribute](#)
- [19.2.15 CLASS_ATTR](#)
- [19.2.16 class_create_file](#)
- [19.2.17 class_remove_file](#)
- [19.2.18 struct class_device](#)
- [19.2.19 class_get_devdata](#)
- [19.2.20 class_set_devdata](#)
- [19.2.21 class_device_register](#)
- [19.2.22 class_device_unregister](#)
- [19.2.23 class_device_initialize](#)
- [19.2.24 class_device_add](#)
- [19.2.25 class_device_del](#)
- [19.2.26 class_device_rename](#)
- [19.2.27 class_device_get](#)
- [19.2.28 class_device_put](#)
- [19.2.29 struct class_device_attribute](#)
- [19.2.30 CLASS_DEVICE_ATTR](#)
- [19.2.31 class_device_create_file](#)
- [19.2.32 class_device_remove_file](#)
- [19.2.33 class_device_create_bin_file](#)
- [19.2.34 class_device_remove_bin_file](#)
- [19.2.35 struct class_interface](#)
- [19.2.36 class_interface_register](#)
- [19.2.37 class_interface_unregister](#)

[19.3 class](#)

- [19.3.1](#)

- [19.3.2](#)

/

- [19.3.3](#)

[20 hotplug](#)

[21 udev](#)

[22 firmware](#)

[22.1 firmware API](#)

- [22.1.1 struct fw](#)

- [22.1.2 request_firmware](#)

- [22.1.3 request_firmware_nowait](#)

- [22.1.4 release_firmware](#)

[22.2 firmware](#)

[23 References](#)

1

[\[edit\]](#)

API

가

2.6.11 API <http://lwn.net>

- 2005.9.28 -

2

[\[edit\]](#)

OS Monolithic micro


가 , 가 , 가

1.2

2.5 가 , kobject 가

/

API

~~~~~ 

[\[edit\]](#)

# 3

2.6 kobject 가 가 가

platform\_bus\_type 가

[\[edit\]](#)

## 3.1 platform API

### 3.1.1 struct platform\_device

[\[edit\]](#)

platform\_bus\_type가<linux/device.h>

platform\_device.

:

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

|               |      |
|---------------|------|
|               |      |
| name          |      |
| id            | 가id. |
| dev           |      |
| num_resources |      |
| resource      |      |

:

```
static void xxx_release_dev(struct device *dev)
{
}

static struct platform_device xxx_device = {
    .name = "xxx",
    .id = 0,
    .dev = {
        .release = xxx_release_dev,
    },
    .num_resources = 0,
};
```

3.1.2 platform\_device\_register

[\[edit\]](#)

.

:

```
int platform_device_register(struct platform_device *pdev)
```

3.1.3 platform\_device\_unregister

[\[edit\]](#)

:

```
void platform_device_unregister(struct platform_device *pdev)
```

## 4

[\[edit\]](#)

### 4.1 device\_driver API

[\[edit\]](#)

#### 4.1.1 struct device\_driver

[\[edit\]](#)

driver\_register/device\_unregister

&lt;linux/device.h&gt;

.


```
struct device_driver {
    const char *name;
    struct bus_type *bus;

    struct completion unloaded;
    struct kobject kobj;
    struct klist klist_devices;
    struct klist_node knode_bus;

    struct module *owner;

    int (*probe) (struct device *dev);
    int (*remove) (struct device *dev);
    int (*shutdown) (struct device *dev);
    int (*suspend) (struct device *dev, pm_message_t state, u32 level);
    int (*resume) (struct device *dev, u32 level);
};
```

|               |                        |
|---------------|------------------------|
|               |                        |
| name          | ( )                    |
| bus           | &platform_bus_type ( ) |
| unloaded      |                        |
| kobj          | ( )                    |
| klist_devices |                        |
| knode_bus     |                        |
| owner         |                        |
| probe()       | ( )                    |
| remove()      | 가 ( )                  |

|                                                                                                      |        |
|------------------------------------------------------------------------------------------------------|--------|
| shutdown()                                                                                           |        |
| suspend()                                                                                            | (    ) |
| resume()                                                                                             | (    ) |
|  foo() ()가 (    ) . |        |

[\[edit\]](#)

### 4.1.2 driver\_register

```
int driver_register(struct device_driver *drv);
```

: driver\_unregister

```
static struct device_driver foo_driver = {
    .name      = "foo",
    .bus       = &platform_bus_type,
    .probe     = foo_probe,
    .remove    = foo_remove,
    .suspend   = foo_suspend,
    .resume    = foo_resume,
};

static int __init foo_init(void) {
    int err;

    err = driver_register(&foo_driver);

    return err;
}
```

### 4.1.3 driver\_unregister

[\[edit\]](#)

driver\_register() . :

```
void driver_unregister(struct device_driver *drv)
```

: driver\_register

[\[edit\]](#)

### 4.1.4 dev\_set\_drvdata

device driver\_data wrapper .

:

```
void dev_set_drvdata(struct device *dev, void *data)
```



: dev\_get\_drvdata

[\[edit\]](#)

### 4.1.5 dev\_get\_drvdata

device driver\_data 가 wrapper .

:

```
void * dev_get_drvdata(struct device *dev)
```

: dev\_set\_drvdata

[\[edit\]](#)

## 5 device vs driver

2.6 bus device 가 driver . 2.6

가 , .

bus platform bus . 2.6.13

drivers/video/vesafb.c .

```
static struct device_driver vesafb_driver = {
    .name = "vesafb",
    .bus = &platform_bus_type,
    .probe = vesafb_probe,
};

static struct platform_device vesafb_device = {
    .name = "vesafb",
};

static int __init vesafb_init(void)
{
    int ret;
    char *option = NULL;

    ...
    ret = driver_register(&vesafb_driver);

    if (!ret) {
        ret = platform_device_register(&vesafb_device);
        if (ret)
            driver_unregister(&vesafb_driver);
    }
    return ret;
}
```

## 6

[\[edit\]](#)

<linux/cdev.h> .

가

ls -l crw-rw---- c

2.6 API 가 register\_chrdev() 가

2.6 API 2.6.11

[edit]

6.1 API 가

2.6 cdev API register\_chrdev/ unregister\_chrdev major minor 가 0 255 / API minor API cdev\_ alloc\_

[edit]

6.2 API

|                          |            |
|--------------------------|------------|
|                          |            |
| register_chrdev          | .          |
| unregistr_chrdev         | .          |
| cdev_init                | cdev .     |
| cdev_add                 | cdev dev . |
| cdev_del                 | cdev dev . |
| register_chrdev_region   | 가 .        |
| alloc_chrdev_region      | 가 .        |
| unreigster_chrdev_region | .          |

[edit]

6.2.1 struct file\_operations

<linux/fs.h> .

```
struct file_operations {
    struct module *owner;
    loff_t (*llseek) (struct file *, loff_t, int);
    ssize_t (*read) (struct file *, char __user *, size_t, loff_t *);
    ssize_t (*aio_read) (struct kiocb *, char __user *, size_t, loff_t *);
    ssize_t (*write) (struct file *, const char __user *, size_t, loff_t *);
```

```

ssize_t (*aio_write) (struct kiocb *, const char __user *, size_t, loff_t);
int (*readdir) (struct file *, void *, filldir_t);
unsigned int (*poll) (struct file *, struct poll_table_struct *);
int (*ioctl) (struct inode *, struct file *, unsigned int,
              unsigned long);
long (*unlocked_ioctl) (struct file *, unsigned int, unsigned long);
long (*compat_ioctl) (struct file *, unsigned int, unsigned long);
int (*mmap) (struct file *, struct vm_area_struct *);
int (*open) (struct inode *, struct file *);
int (*flush) (struct file *);
int (*release) (struct inode *, struct file *);
int (*fsync) (struct file *, struct dentry *, int datasync);
int (*aio_fsync) (struct kiocb *, int datasync);
int (*fasync) (int, struct file *, int);
int (*lock) (struct file *, int, struct file_lock *);
ssize_t (*readv) (struct file *, const struct iovec *, unsigned long,
                 loff_t *);
ssize_t (*writev) (struct file *, const struct iovec *, unsigned long,
                 loff_t *);
ssize_t (*sendfile) (struct file *, loff_t *, size_t, read_actor_t, void *);
ssize_t (*sendpage) (struct file *, struct page *, int, size_t, loff_t *,
                    int);
unsigned long (*get_unmapped_area)(struct file *, unsigned long,
                                   unsigned long, unsigned long,
                                   unsigned long);

int (*check_flags)(int);
int (*dir_notify)(struct file *filp, unsigned long arg);
int (*flock) (struct file *, int, struct file_lock *);
};

```

## 6.2.2 register\_chrdev

[\[edit\]](#)

, minor 가 0 256 .

```

int register_chrdev(unsigned int major, const char *name,
                   struct file_operations *fops);

```

## 6.2.3 unregister\_chrdev

[\[edit\]](#)

register\_chrdev .

```

void unregister_chrdev(unsigned int major, const char *name);

```

## 6.2.4 struct cdev

[\[edit\]](#)

struct cdev; . <linux/cdev.h> .

```

struct cdev {
    struct kobject kobj;
    struct module *owner;
};

```

```

struct file_operations *ops;
struct list_head list;
dev_t dev;
unsigned int count;
};

```

|       |                    |
|-------|--------------------|
|       |                    |
| kobj  | kobject            |
| owner | , ID . THIS_MODULE |
| ops   | .                  |
| list  |                    |
| dev   |                    |
| count |                    |



THIS\_MODULE

&lt;linux/module.h&gt;

[\[edit\]](#)

## 6.2.5 cdev\_init

cdev .

```
void cdev_init(struct cdev *cdev, struct file_operations *fops)
```

cdev 0 , cdev->kobj.ktype ktype\_cdev\_default cdev->kobj kobject\_init  
 . cdev->ops fops .  
 cdev ops 가 fops .  
 :

```

static struct file_operations xxx_fops = {
    .read = xxx_read,
    .write = xxx_write,
    .open = xxx_open,
    .release = xxx_release,
    .owner = THIS_MODULE,
};

static struct cdev xxx_cdev = {
    .kobj = { .name = "xxx", },
    .owner = THIS_MODULE,
};

static int __init xxx_init(void)
{
    dev_t dev = MKDEV(XXX_MAJOR, 0);

```

```

if (register_chrdev_region(dev, MAX_XXX_MINORS, "xxx"))
    goto error;

cdev_init(&xxx_cdev, &xxx_fops);

...

```

### 6.2.6 cdev\_add

[\[edit\]](#)

cdev\_map

cdev\_map dev

```

int cdev_add(struct cdev *p, dev_t dev, unsigned count)

```

```

:
drivers/char/raw.c raw_init()

```

```

static int __init raw_init(void)
{
    int i;
    dev_t dev = MKDEV(RAW_MAJOR, 0);

    if (register_chrdev_region(dev, MAX_RAW_MINORS, "raw"))
        goto error;

    cdev_init(&raw_cdev, &raw_fops;
    if (cdev_add(&raw_cdev, dev, MAX_RAW_MINORS)) {
        unregister_chrdev_region(dev, MAX_RAW_MINORS);
        goto error;
    }

    ...
    return 0;

error:
    ...
    return 1;
}

```

### 6.2.7 cdev\_del

[\[edit\]](#)

cdev\_add

```

void cdev_del(struct cdev *p)

```

: cdev\_add

:

drivers/char/raw.c raw\_exit

```
static void __exit raw_exit(void)
{
    ...
    cdev_del(&raw_cdev);
    unregister_chrdev_region(MKDEV(RAW_MAJOR, 0), MAX_RAW_MINORS);
}
```

## 6.2.8 register\_chrdev\_region

[\[edit\]](#)

cdev\_init      cdev      .      name      from      major      minor  
                  count      .      <linux/fs.h>      .  
                  가      .      reigster\_chrdev\_region  
 alloc\_chrdev\_region      .

```
int register_chrdev_region(dev_t from, unsigned count, const char *name)
```



count      major      major      가      가      .  
 : alloc\_chrdev\_region, unregister\_chrdev\_region

[\[edit\]](#)

## 6.2.9 alloc\_chrdev\_region

unregister\_chrdev\_region      register\_chrdev\_region      .

```
int alloc_chrdev_region(dev_t *dev, unsigned baseminor, unsigned count,
                        const char *name);
```

dev      . name  
 minor      baseminor      count      .

## 6.2.10 unregister\_chrdev\_region

[\[edit\]](#)

register\_crhdev\_region      , from      count      .  
 <linux/fs.h>      .

```
void unregister_chrdev_region(dev_t from, unsigned count)
```

: register\_chrdev\_region

:

drivers/char/raw.c raw\_exit

```
static void __exit raw_exit(void)
{
    ...
    cdev_del(&raw_cdev);
    unregister_chrdev_region(MKDEV(RAW_MAJOR, 0), MAX_RAW_MINORS);
}
```

## 7

[\[edit\]](#)

dev\_t u32 . 20 가 12 가 .  
 \*chrdevs? MAX\_PROBE\_HASH 가 . MAX\_PROBE\_HASH 255 . , 가 가  
 , 20 255 . %  
 MAX\_PROBE\_HASH linked list .

[\[edit\]](#)

### 7.1 MKDEV

ma major mi minor .

```
dev_t MKDEV(ma, mi)
```

### 7.2 MAJOR

[\[edit\]](#)

major 가 .

```
unsigned int MAJOR(dev_t n)
```

### 7.3 MINOR

[\[edit\]](#)

minor 가 .

```
unsigned int MINOR(dev_t n)
```

## 8

[\[edit\]](#)

C++ JAVA . C  
 . C  
 C 가 . ,  
 .  
 C , struct C C  
 . gcc ,  
 .  
 , USB 가 가 .

, USB 가 . USB

?

, 가 . ,

가 . 가

2.5 kobject

, kobject 가 . , , , ,

가 ,

struct 가 가

가 kobject

refcount가 0 kobject refcount kobject , 0 kobject kobject

hotplug event 가

[\[edit\]](#)

## 8.1 subsystem, kset, ktype, kobj

kobject 가 kset, ktype, subsystem, class

, kset, ktype, kobj

kset struct kset;, ktype struct kobj\_type;, kobj struct kobject;

:

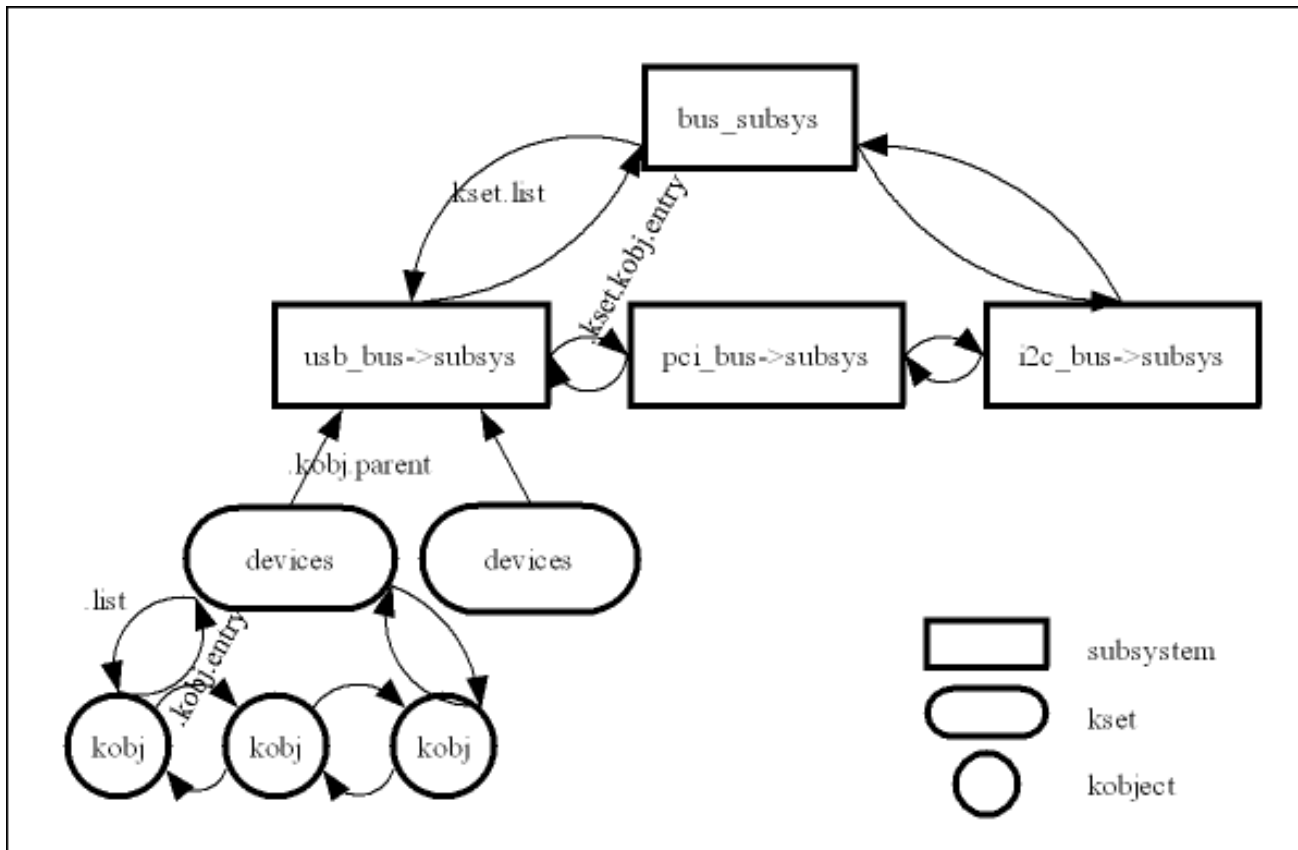
```
kset --- kobj
|
+--->ktype+--->release()
|
|      +--->sysfs_ops
+--->kset_hotplug_ops
```

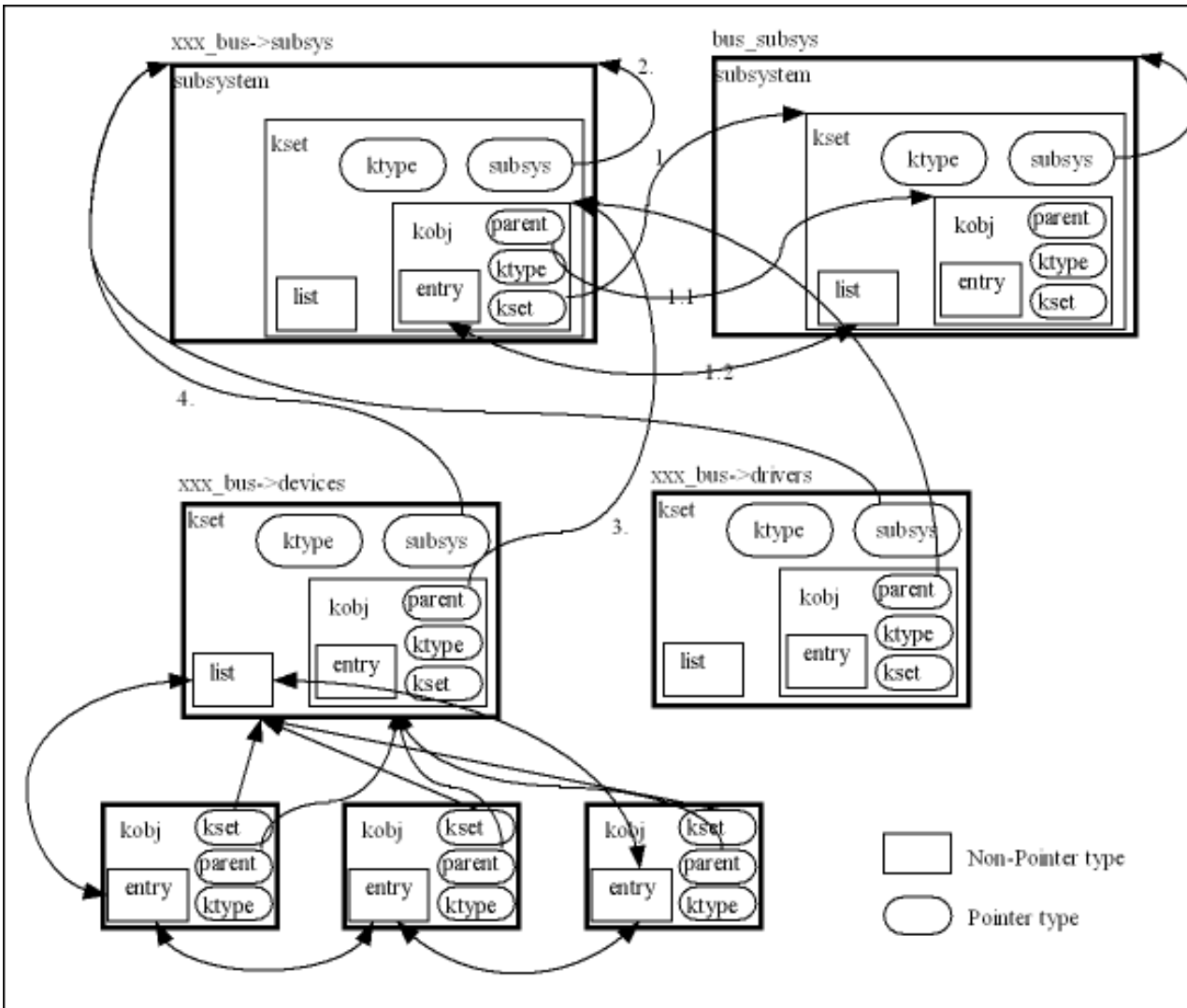
kset (type) kobj . ktype , release sysfs , ktype . kobject , ktype .

kset 가 hotplug .



kset, ktype, kobj가  
가





bus\_subsys decl\_subsys() . bus\_subsys kobject "bus" . bus\_subsys .kset.ktype bus\_ktype  
 xxx\_bus->subsys .kset.kobj.kset bus\_subsys kset .

[\[edit\]](#)

## 9 kref

kobject 가 refcount . ,

kobject가 refcount kref .

```
struct kref {
    atomic_t refcount;
};
```

atomic\_t .

가

[\[edit\]](#)

## 9.1 kref API

|           |          |   |   |     |         |
|-----------|----------|---|---|-----|---------|
|           |          |   |   |     |         |
| kref_init | refcount | 1 | . |     |         |
| kref_get  | refcount | 1 | 가 | .   |         |
| kref_put  | refcount | 1 | . | , 0 | release |

### 9.1.1 struct kref

[\[edit\]](#)

```
struct kref {
    atomic_t refcount;
};
```

### 9.1.2 kref\_init

[\[edit\]](#)

refcount 1 .

```
void kref_init(struct kref *kref);
```

### 9.1.3 kref\_get

[\[edit\]](#)

atomic\_inc refcount 1 가 .

```
void kref_get(struct kref *kref);
```

### 9.1.4 kref\_put

[\[edit\]](#)

atomic\_dec\_and\_test() refcount 1 . , 0 release() 1  
0 .

```
int kref_put(struct kref *kref, void(*release)(struct kref *kref));
```

## 10 kobject

[\[edit\]](#)

kobject struct kobject; . <linux/kobject.h> .

kobject가 LDM 가 . , 가  
. API kobject . LDM  
, kobject .

```
struct kobject {
    char *k_name;
    char name[KOBJ_NAME_LEN];
    struct kref kref;
    struct list_head entry;
```

Linux kernel v2.6.11

k\_name kobject name . , k\_name 가  
name k\_name 가 k\_name name .

## 10.1 kobject API

[\[edit\]](#)

### 10.1.1 kobject\_set\_name

```
int kobject_set_name(struct kobject *kobj, const char *fmt, ...)
```

```

fmt          printk          .          kobject가
, 20          , 20          kobject  name
          가          , 20          .
:
{drivers/base/bus.c}} bus_register()

```

```

retval = kobject_set_name(&bus->subsys.kset.kobj, "%s", bus->name);
if (retval)
    goto out;

```

### [10.1.2 kobject\\_name](#)

[\[edit\]](#)

kobj k\_name .

```
char * kobject_name(struct kobject *kobj);
```

### [10.1.3 kobject\\_init](#)

[\[edit\]](#)

```

kobject      refcount      1          entry          . kset_get          kobject가          kset
kobj  refcount      가          , kobject_init      kobj가          kset      (kobj->kset
NULL      ) kset      kobj  refcount      가          .

```

```
void kobject_init(struct kobject *kobj);
```

### [10.1.4 kobject\\_cleanup](#)

[\[edit\]](#)

kobject .

```
vodi kobject_cleanup(struct kobject *);
```

### [10.1.5 kobject\\_add](#)

[\[edit\]](#)

kset kobj 가 . <linux/kobject.h> .

```
int kobject_add(struct kobject *kobj);
```

kobj parent가 kobj kset kobj parent .

kobject\_add kobject sysfs . parent

[\[edit\]](#)

### [10.1.6 kobject\\_del](#)

```
void kobject_del(struct kobject *);
```

sysfs\_remove\_dir          unlink          .

[\[edit\]](#)

### [10.1.7 kobject\\_register](#)

가 .

```
int kobject_register(struct kobject *);
```

kobject\_register          kobject\_init          kobject\_add          .          kobject\_hotplug  
가 ADD          .

[\[edit\]](#)

### [10.1.8 kobject\\_unregister](#)

hotplug          가 (REMOVE)          .          kobject\_del          kobj          sysfs  
          kobject\_put          kobject          refcount          .          refcount가 0          ktype          release  
          kobject          .

```
void kobject_unregister(struct kobject *);
```

### [10.1.9 kobject\\_get](#)

[\[edit\]](#)

refcount          가 .

```
struct kobject *kobject_get(struct kobject *);
```

### [10.1.10 kobject\\_put](#)

[\[edit\]](#)

refcount          .

```
void kobject_put(struct kobject *);
```

kref\_put          .

[\[edit\]](#)

### [10.1.11 kobject\\_get\\_path](#)

```
char *kobject_get_path(struct kobject *kobj, int gfp_mask);
```

## [11 kset](#)

[\[edit\]](#)

```

kset kobject
kset 가
kobject 가
.kset
subsys
kset kobjects
가 sysfs
.kobject
kset

```

```

kset struct kset; . <linux/kobject.h> .

```

```

struct kset {
    struct subsystem *subsys;
    struct kobj_type *ktype;
    struct list_head list;
    struct kobject kobj;
    struct kset_hotplug_ops *hotplug_ops;
};

```

|             |  |
|-------------|--|
|             |  |
| subsys      |  |
| ktype       |  |
| list        |  |
| kobj        |  |
| hotplug_ops |  |

[\[edit\]](#)

## 11.1 kset API

|                 |  |
|-----------------|--|
|                 |  |
| kset_init       |  |
| kset_add        |  |
| kset_register   |  |
| kset_unregister |  |
| to_kset         |  |
| kset_get        |  |
| kset_put        |  |

[\[edit\]](#)

### 11.1.1 struct kset

```

kset struct kset; . <linux/kobject.h> .

```

```

struct kset {
    struct subsystem *subsys;
    struct kobj_type *ktype;
    struct list_head list;
    struct kobject kobj;
};

```

```
struct kset_hotplug_ops *hotplug_ops;
};
```

|             |  |
|-------------|--|
|             |  |
| subsys      |  |
| ktype       |  |
| list        |  |
| kobj        |  |
| hotplug_ops |  |

[\[edit\]](#)

### [11.1.2 kset\\_init](#)

kset . kset kobj (kobject\_init) list

```
void kset_init(struct kset *k);
```

### [11.1.3 kset\\_add](#)

[\[edit\]](#)

kset 가 .

```
int kset_add(struct kset *k);
```

k kobj parent(kobj.parent가 NULL) kset(kobj.kset가 NULL) k가  
subsys(k->subsys가 NULL) , k->kobj parent subsys kset.kobj 가 .  
kobject\_add k->kobj kset 가 .  
kset\_add kset 가 kset\_register .  
subsystem\_register kset\_add .

### [11.1.4 kset\\_register](#)

[\[edit\]](#)

```
int kset_register(struct kset *k);
```

### [11.1.5 kset\\_unregister](#)

[\[edit\]](#)

```
void kset_unregister(struct kset *k);
```

### [11.1.6 to\\_kset](#)

[\[edit\]](#)

```
struct kset *to_kset(struct kobject *kobj);
```



### [11.1.7 kset\\_get](#)

[\[edit\]](#)

```
struct kset *kset_get(struct kset *k);
```

### [11.1.8 kset\\_put](#)

[\[edit\]](#)

```
void kset_put(struct kset *k);
```

## [12 ktype](#)

[\[edit\]](#)

```
struct kobj_type {
    void (*release)(struct kobject *);
    struct sysfs_ops *sysfs_ops;
    struct attribute **default_attrs;
};
```

## [13 sysfs](#)

[\[edit\]](#)

### [13.1 /sys](#)

[\[edit\]](#)

sysfs 가 (/) sys 가 .  
가 .  
/ .  
/ / 가 .  
/sys .

```
/sys
|-- block
|-- bus
|-- class
|-- devices
|-- firmware
|-- kernel
|-- module
`-- power
```

### [13.2 sysfs API](#)

[\[edit\]](#)

<linux/sysfs.h>

```
struct sysfs_ops {
    ssize_t (*show)(struct kobject *, struct attribute *, char *);
```

```
ssize_t (*store)(struct kobject *, struct attribute *, const char *, size_t);
};
```

|                       |   |
|-----------------------|---|
|                       |   |
| sysfs_create_dir      | . |
| sysfs_remove_dir      | . |
| sysfs_rename_dir      | . |
| sysfs_create_file     |   |
| sysfs_update_file     |   |
| sysfs_remove_file     |   |
| sysfs_create_link     |   |
| sysfs_remove_link     |   |
| sysfs_create_bin_file |   |
| sysfs_remove_bin_file |   |
| sysfs_create_group    |   |
| sysfs_remove_group    |   |

[\[edit\]](#)

### [13.2.1 sysfs\\_create\\_dir](#)

`<linux/sysfs.h>`

```
int sysfs_create_dir(struct kobject *kobj);
```

가 `/lib/kobject.c` `static int creat_dir()`

[\[edit\]](#)

### [13.2.2 sysfs\\_remove\\_dir](#)

```
void sysfs_remove_dir(struct kobject *kobj);
```

가 `.kobject_del` `sysfs_remove_dir`

[\[edit\]](#)

### [13.2.3 sysfs\\_rename\\_dir](#)

```
int sysfs_rename_dir(struct kobject *kobj, const char *new_name);
```

, kobject\_rename . kobject\_rename

가 sysfs\_rename .

[\[edit\]](#)

### [13.2.4 sysfs\\_create\\_file](#)

sysfs .

```
int sysfs_create_file(struct kobject *, const struct attribute *);
```

:  
drivers/base/driver.c driver\_create\_file()

```
int driver_create_file(struct device_driver *drv, struct driver_attribute *attr)
{
    int error;
    if (get_driver(drv)) {
        error = sysfs_create_file(&drv->kobj, &attr->attr);
        put_driver(drv);
    } else
        error = -EINVAL;
    return error;
}
```

### [13.2.5 sysfs\\_update\\_file](#)

[\[edit\]](#)

attribute timestamp .

```
int sysfs_update_file(struct kobject *kobj, const struct attribute *attr);
```

attribute , attribute hotplug timestamp .  
pci 가 attribute timestamp .  
timestamp 가

### [13.2.6 sysfs\\_remove\\_file](#)

[\[edit\]](#)

sysfs .

```
void sysfs_remove_file(struct kobject *, const struct attribute *);
```

:  
drivers/base/driver.c driver\_remove\_file()

```
void driver_remove_file(struct device_driver *drv, struct driver_attribute *attr)
{
    if(get_driver(drv)) {
        sysfs_remove_file(&drv->kobj, &attr->attr);
        put_driver(drv);
    }
}
```

### 13.2.7 sysfs\_create\_link

[\[edit\]](#)

```
int sysfs_create_link(struct kobject *kobj, struct kobject *target,
                    const char *name);
```

kobj    name                    가                    . target                    가 가                    .  
       가  
                                   가            sysfs\_create\_link            .  
 :  
                                   2.6.11    drivers/base/bus.c    bus\_add\_device()            .

```
int bus_add_device(struct device *dev)
{
    struct bus_type *bus = get_bus(dev->bus);
    int error = 0;

    if (bus) {
        ...
        sysfs_create_link(&bus->device.kobj, &dev->kobj, dev->bus_id);
    }
    return error;
}
```

sysfs\_create\_link                    /sys/bus/xxx/devices/                    /sys/devices/                    bus\_id  
                                   .  
       /sys

```
greendrm@devil:/sys/bus/pci/devices$ tree /sys/bus/pci/devices/
/sys/bus/pci/devices/
|-- 0000:00:00.0 -> ../../../../devices/pci0000:00/0000:00:00.0
|-- 0000:00:01.0 -> ../../../../devices/pci0000:00/0000:00:01.0
|-- 0000:01:00.0 -> ../../../../devices/pci0000:00/0000:00:01.0/0000:01:00.0
|-- 0000:03:03.0 -> ../../../../devices/pci0000:00/0000:00:1e.0/0000:03:03.0
`-- 0000:03:06.0 -> ../../../../devices/pci0000:00/0000:00:1e.0/0000:03:06.0
```

### 13.2.8 sysfs\_remove\_link

[\[edit\]](#)

```
void sysfs_remove_link(struct kobject *kobj, char *name);
```

### [13.2.9 sysfs\\_create\\_bin\\_file](#)

[\[edit\]](#)

```
int sysfs_create_bin_file(struct kobject *kobj, struct bin_attribute *attr);
```

sysfs\_create\_file, text

[\[edit\]](#)

### [13.2.10 sysfs\\_remove\\_bin\\_file](#)

```
int sysfs_remove_bin_file(struct kobject *kobj, struct bin_attribute *attr);
```

### [13.2.11 struct attribute\\_group](#)

[\[edit\]](#)

attribute  
attribute\_group API  
<linux/sysfs.h>

```
struct attribute_group {
    const char *name;
    struct attribute **attrs;
};
```

### [13.2.12 sysfs\\_create\\_group](#)

[\[edit\]](#)

attribute

```
int sysfs_create_group(struct kobject *, const struct attribute_group *);
```

:  
drivers/base/power/sysfs.c

```
static ssize_t state_show(struct device *dev, struct device_attribute *attr,
                          char *buf)
{
    ...
}
```

```
static ssize_t state_store(struct device *dev, struct device_attribute *attr,
                          const char *buf, size_t n)
{
    ...
}

static DEVICE_ATTR(state, 0644, state_show, state_store);

static struct attribute *power_attrs[] = {
    &dev_attr_state.attr,
    NULL,
};

static struct attribute_group pm_attr_group = {
    .name = "power",
    .attrs = power_attrs,
};

int dpm_sysfs_add(struct device *dev)
{
    return sysfs_create_group(&dev->kobj, &pm_attr_group);
}

void dpm_sysfs_remove(struct device *dev)
{
    sysfs_remove_group(&dev->kobj, &pm_attr_group);
}
```

### 13.2.13 sysfs\_remove\_group

[\[edit\]](#)

attribute

```
void sysfs_remove_group(struct kobject *, const struct attribute_group *);
```

:

drivers/base/power/sysfs.c

```
static ssize_t state_show(struct device *dev, struct device_attribute *attr,
                          char *buf)
{
    ...
}

static ssize_t state_store(struct device *dev, struct device_attribute *attr,
                          const char *buf, size_t n)
{
    ...
}

static DEVICE_ATTR(state, 0644, state_show, state_store);
```

```
static struct attribute *power_attrs[] = {
    &dev_attr_state.attr,
    NULL,
};

static struct attribute_group pm_attr_group = {
    .name = "power",
    .attrs = power_attrs,
};

int dpm_sysfs_add(struct device *dev)
{
    return sysfs_create_group(&dev->kobj, &pm_attr_group);
}

void dpm_sysfs_remove(struct device *dev)
{
    sysfs_remove_group(&dev->kobj, &pm_attr_group);
}
```

### 13.3

[\[edit\]](#)

- <http://linux-diag.sourceforge.net/Sysfsutils.html>

libsysfs

sysfs

common API

sysfs

[\[edit\]](#)

## 14 subsystem

### 14.1 subsystem

[\[edit\]](#)

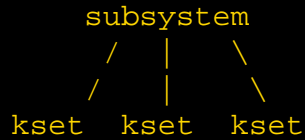
가 . sysfs 가 , block, bus, class, devices sysfs , acpi bus/acpi . , subsystem struct subsystem; <linux/kobject.h> .

```
struct subsystem {
    struct kset kset;
    struct rw_semaphore rwsem;
};
```

subsystem kset . subsystem . , kset serialization . rw\_semaphore . subsystem kset subsystem subsystem kset .

[\[edit\]](#)

### 14.2 subsystem kset



kset                      subsys                      subsystem                      , subsystem                      kset

```

struct subsystem {
    struct kset kset;
    ...
};

struct kset {
    struct subsystem *subsys;
    ...
    struct list_head list;
    ...
};
  
```

### 14.3 subsystem on sysfs

[\[edit\]](#)

subsystem                      sysfs                      /sys/                      .                      sysfs

```

/sys
|-- block
|-- bus
|-- class
|-- devices
|-- firmware
|-- kernel
|-- module
`-- power
  
```

subsystem                      system\_subsys, block\_subsys, bus\_subsys, class\_subsys, devices\_subsys, firmware\_subsys, class\_obj\_subsys, acpi\_subsys, edd\_subsys, vars\_subsys, efi\_subsys, cdev\_subsys, module\_subsys, power\_subsys, pci\_hotplgu\_slocks\_subsys

[\[edit\]](#)

### 14.4 subsystem API

|                |  |
|----------------|--|
|                |  |
| decl_subsys    |  |
| subsystem_init |  |



|                      |  |
|----------------------|--|
| subsystem_register   |  |
| subsystem_unregister |  |
| subsystem_get        |  |
| subsystem_put        |  |

### 14.4.1 decl\_subsys

[\[edit\]](#)

subsystem . <linux/kobject.h> .

```
delc_subsys(name, type, hotplug_ops);
```

|             |               |             |           |   |   |
|-------------|---------------|-------------|-----------|---|---|
|             |               |             |           |   |   |
| name        | subsystem     | name_subsys | subsystem | 가 | . |
| type        | ktype         | . NULL      | 가         | . |   |
| hotplug_ops | hotplug event | hotplus_ops | . NULL    | 가 |   |

 name kobject name kobject .

:  
drivers/block/genhd.c block subsystem .

```
...

1: static struct kobj_type ktype_block = {
2:     .release = disk_release,
3:     .sysfs_ops = &disk_sysfs_ops,
4:     .default_attrs = default_attrs,
5: };

...

6: static struct kset_hotplug_ops block_hotplug_ops = {
7:     .filter = block_hotplug_filter,
8:     .hotplug = block_hotplug,
9: };

/* declare block subsys */
10: static decl_subsys(block, &ktype_block, &block_hotplug_ops);
```

10 block\_sussys block subsystem . block kset ktype ktype\_block  
hotplug event block\_hotplug\_ops . , ktype hotplug block  
kset NULL

[\[edit\]](#)

### 14.4.2 subsystem\_init

&lt;linux/kobject.h&gt;

```
void subsystem_init(struct subsystem *subsys);
```

```

. subsystem_register()
.

```

[\[edit\]](#)

### 14.4.3 subsystem\_register

```

system
  subsystem . <linux/kobject.h>

```

```
int subsystem_register(struct subsystem *s);
```

subsystem, kset\_add s kset 가 . rwsem  
kset subsys가 가 .

[\[edit\]](#)

#### 14.4.4 subsystem\_unregister

subsystem .

```
void subsystem_unregister(struct subsystem *subsys);
```

```
subsystem_unregister    kset_unregister          kset_unregister    kobject_register
kobject_register
```

### 14.4.5 subsystem\_get

[\[edit\]](#)

```
struct subsystem *subsys_get(struct subsystem *subsys);
```

#### 14.4.6 subsystem\_put

[edit]

```
void subsys_put(struct subsystem *subsys);
```

## 15 kobj\_map

[edit]

```
API    <linux/kobj_map.h>
```

[edit]

## 15.1 kobj\_map API

|               |     |        |     |     |   |
|---------------|-----|--------|-----|-----|---|
|               |     |        |     |     |   |
| kobj_map      | map | dev    | 가   | .   |   |
| kobj_unmap    | map | dev    |     | .   |   |
| kobj_lookup   |     | domain | map | dev | . |
| kobj_map_init | map |        |     | .   |   |

### [15.1.1 kobj\\_probe\\_t](#)

[\[edit\]](#)

```
typedef struct kobject *kobj_probe_t(dev_t, int *, void *);
```

### [15.1.2 struct kobj\\_map](#)

[\[edit\]](#)

kobj\_map                      struct kobj\_map                      .  
 <drivers/base/map. c                      .

```
struct kobj_map {
    struct probe {
        struct probe *next;
        dev_t dev;
        unsigned long range;
        struct module *owner;
        kobj_probe_t *get;
        int (*lock)(dev_t, void *);
        void *data;
    }probe[255];
    struct semaphore *sem;
};
```

### [15.1.3 kobj\\_map](#)

[\[edit\]](#)

map                      (dev)                      가                      .

```
int kobj_map(struct kobj_map *domain, dev_t dev, unsigned long range,
            struct module *, kobj_probe_t *probe,
            int (*lock)(dev_t dev, void *), void *data);
```

:[\[BR\]](#) fs/char\_dev. c

```
int cdev_add(struct cdev *p, dev_t dev, unsigned count)
{
    ...
    return kobj_map(cdev_map, dev, count, NULL, exact_match, exact_lock, p);
}
```

```
void blk_register_region(dev_t dev, unsigned long range, struct module *module,
```

```

        struct kobject *(*probe)(dev_t, int *, void *),
        int (*lock)(dev_t, void *), void *data)
{
    kobj_map(bdev_map, dev, range, module, probe, lock, data);
}

```

#### 15.1.4 kobj\_unmap

[\[edit\]](#)

```
void kobj_unmap(struct kobj_map *, dev_t, unsigned long);
```

#### 15.1.5 kobj\_lookup

[\[edit\]](#)

domain (cdev\_map, bdev\_map)      dev      .  
kobject      .

```
struct kobject *kobj_lookup(struct kobj_map *domain, dev_t dev, int *index);
```

:  
fs/char/dev.c      .

```

int chrdev_open(struct inode *inode, struct file *filp)
{
    ...
    if (!p) {
        ...
        kobj = kobj_lookup(cdev_map, inode->i_rdev, &idx);
        ...
    }
    ...
}

```

drivers/block/genhd.c      .

```

struct gendisk *get_gendisk(dev_t dev, int *part)
{
    struct kobject *kobj = kobj_lookup(bdev_map, dev, part);
    return kobj ? to_disk(kobj) : NULL;
}

```

#### 15.1.6 kobj\_map\_init

[\[edit\]](#)

map      .

```
struct kobj_map *kobj_map_init(kobj_probe_t *, struct semaphore *);
```

:

fs/char\_dev. c

```
void __init chrdev_init(void)
{
    cdev_map = kobj_map_init(base_probe, &chrdev_lock);
}
```

drivers/block/genhd. c

```
static int __init genhd_device_init(void)
{
    bdev_map = kobj_map_init(base_probe, &block_subsys_sem);
    ...
}
```

## [16](#) bus

[\[edit\]](#)

### [16.1](#) bus on sysfs

[\[edit\]](#)

```
/sys/bus
|-- i2c
|-- ide
|-- ieee1394
|-- pci
|-- platform
|-- pnp
|-- scsi
|-- serio
`-- usb
```

### [16.2](#) bus API

[\[edit\]](#)

|                  |  |
|------------------|--|
|                  |  |
| bus_register     |  |
| bus_unregister   |  |
| bus_for_each_dev |  |
| bus_for_each_drv |  |
| BUS_ATTR         |  |
| bus_create_file  |  |
| bus_remove_file  |  |

#### [16.2.1](#) bus\_register

[\[edit\]](#)

```
int bus_register(struct bus_type *bus);
```

### [16.2.2 bus\\_unregister](#)

[\[edit\]](#)

```
void bus_unregister(struct bus_type *bus);
```

### [16.2.3 bus\\_for\\_each\\_dev](#)

[\[edit\]](#)

```
int bus_for_each_dev(struct bus_type *bus, struct device *start, void *data,
                    int (*fn)(struct device *, void *));
```

### [16.2.4 bus\\_for\\_each\\_drv](#)

[\[edit\]](#)

```
int bus_for_each_drv(struct bus_type *bus, struct device_driver *start,
                    void *data, int (*fn)(struct device_driver *, void *));
```

### [16.2.5 BUS\\_ATTR](#)

[\[edit\]](#)

```
BUS_ATTR(name, mode, show, store);
```

### [16.2.6 bus\\_create\\_file](#)

[\[edit\]](#)

```
int bus_create_file(struct bus_type *bus, struct bus_attribute *attr);
```

### [16.2.7 bus\\_remove\\_file](#)

[\[edit\]](#)

```
void bus_remove_file(struct bus_type *bus, struct bus_attribute *attr);
```

## [17 device](#)

[\[edit\]](#)

### [17.1 device on sysfs](#)

[\[edit\]](#)

```
/sys/devices
|-- pci0000:00
|-- platform
|-- pnp0
`-- system
```

### [17.2 device API](#)

[\[edit\]](#)

|                    |  |
|--------------------|--|
| device_register    |  |
| device_unregister  |  |
| DEVICE_ATTR        |  |
| device_create_file |  |
| device_remove_file |  |

[\[edit\]](#)

### [17.2.1](#) struct device

device                      struct device;                      .                      <linux/device.h>                      .

```
struct device {
    ...
    struct device *parent;

    struct kobject kobj;
    char bus_id[BUS_ID_SIZE];
    ...

    struct bus_type *bus;
    struct device_driver *driver;

    void *driver_data;
    void *platform_data;

    void *firmware_data;

    ...

    void (*release)(struct device *dev);
};
```

### [17.2.2](#) device\_register

[\[edit\]](#)

```
int device_register(struct device *dev);
```

### [17.2.3](#) device\_unregister

[\[edit\]](#)

```
void device_unregister(struct device *dev);
```

### [17.2.4](#) DEVICE\_ATTR

[\[edit\]](#)

```
DEVICE_ATTR(name, mode, show, store);
```

### [17.2.5](#) device\_create\_file

[\[edit\]](#)

```
int device_create_file(struct device *dev, struct device_attribute *attr);
```

### [17.2.6 device\\_remove\\_file](#)

[\[edit\]](#)

```
void device_remove_file(struct device *dev, struct device_attribute *attr);
```

## [18 driver](#)

[\[edit\]](#)

### [18.1 driver API](#)

[\[edit\]](#)

|                    |  |
|--------------------|--|
|                    |  |
| driver_register    |  |
| driver_unregister  |  |
| DRIVER_ATTR        |  |
| driver_create_file |  |
| driver_remove_file |  |

#### [18.1.1 driver\\_register](#)

[\[edit\]](#)

```
int driver_register(struct device_driver *drv);
```

#### [18.1.2 driver\\_unregister](#)

[\[edit\]](#)

```
void driver_unregister(struct device_driver *drv);
```

#### [18.1.3 DRIVER\\_ATTR](#)

[\[edit\]](#)

<linux/device.h>      . driver\_attr\_##name      attribute      .

```
DRIVER_ATTR(name, mode, show, store);
```

#### [18.1.4 driver\\_create\\_file](#)

[\[edit\]](#)

sysfs      .

```
int driver_create_file(struct device_driver *drv, struct driver_attribute *attr);
```

#### [18.1.5 driver\\_remove\\_file](#)

[\[edit\]](#)

sysfs      .



```
void driver_remove_file(struct device_driver *drv, struct driver_attribute *attr);
```

## 19 class

[\[edit\]](#)

```

    kobject      subsystem      .      가      가
,      subsystem      .      class

    subsystem, kset      .      class

    input      ,      class

```

[\[edit\]](#)

### 19.1 class on sysfs

```

/sys/class
|-- graphics
|-- i2c-adapter
|-- ieee1394
|-- ieee1394_host
|-- ieee1394_node
|-- ieee1394_protocol
|-- input
|-- mem
|-- misc
|-- net
|-- pci_bus
|-- printer
|-- scsi_device
|-- scsi_generic
|-- scsi_host
|-- sound
|-- tty
|-- usb
|-- usb_host
`-- vc

```

### 19.2 class

[\[edit\]](#)

2.6.13( )

- class\_simple\_create
- class\_simple\_destroy
- class\_simple\_device\_add
- class\_simple\_device\_remove

2.6.13( )

- class\_create
- class\_destroy

- class\_device\_destroy
- class\_device\_create

### 19.2.1 struct class

[\[edit\]](#)

class struct class; . <linux/device.h> .

```
struct class {
    const char *name;
    struct module *owner;

    struct subsystem subsys;
    struct list_head children;
    struct list_head interfaces;
    struct semaphore sem;

    struct class_attribute *class_attrs;
    struct class_device_attribute *class_dev_attrs;

    int (*hotplug)(struct class_device *dev, char **envp,
                  int num_envp, char *buffer, int buffer_size);
    void (*release)(struct class_device *dev);
    void (*class_release)(struct class *class);
};
```

### 19.2.2 class\_simple\_create

[\[edit\]](#)

```
struct class_simple *class_simple_create(struct module *owner, char *name)
```



2.6.13 .

### 19.2.3 class\_simple\_destroy

[\[edit\]](#)

```
void class_simple_destroy(struct class_simple *cs)
```



2.6.13 .

### 19.2.4 class\_simple\_device\_add

[\[edit\]](#)

```
struct class_device *class_simple_device_add(struct class_simple *cs, dev_t dev,
                                             struct device, const char *fmt, ...)
```



2.6.13 .

### 19.2.5 class\_simple\_device\_remove

[\[edit\]](#)

```
void class_simple_device_remove(dev_t dev)
```



2.6.13

[\[edit\]](#)

### [19.2.6 class\\_create](#)

```
struct class *class_create(struct module *owner, char *name)
```

```
class_destroy
```

```
: class_destroy
```

### [19.2.7 class\\_destroy](#)

[\[edit\]](#)

```
void class_destroy(struct class *cls)
```

```
:
```

```
struct *xxx_class;
EXPORT_SYMBOL(xxx_class);

static void __exit cleanup_xxx(void)
{
    ...
    class_destroy(xxx_class);
}

static int __init init_xxx(void)
{
    ...
    xxx_class = class_create(THIS_MODULE, "xxx");
    if (IS_ERR(sound_class))
        return PTR_ERR(sound_class);

    return 0;
}
```

```
: class_create
```

[\[edit\]](#)

### [19.2.8 class\\_device\\_create](#)

```
class device . class_device class_device_register .
```

```
struct class_device *class_device_create(struct class *cls, dev_t devt,
                                         struct device *device, char *fmt, ...)
```

```

:
sound/sound_core.c    sound_insert_unit()
.

```

```

static int sound_insert_unit(...)
{
    ...
    if (...)
        ...
    else
        sprintf(s->name, "sound/%s%d", name, r / SOUND_STEP);
    ...
    class_device_create(sound_class, MKDEV(SOUND_MAJOR, s->unit_minor),
                        NULL, s->name+6);
    ...
}

```

### [19.2.9 class\\_device\\_destroy](#)

[\[edit\]](#)

```

class device    class_device_unregister
.

```

```

void class_device_destroy(struct class *cls, dev_t devt)

```

```

:
sound/sound_core.c    sound_remove_unit()
.

```

```

static void sound_remove_unit(...)
{
    ...
    if (p) {
        ...
        class_device_destroy(sound_class, MKDEV(SOUND_MAJOR, p->unit_minor));
        ...
    }
}

```

### [19.2.10 class\\_register](#)

[\[edit\]](#)

```

int class_register(struct class *cls);

```

### [19.2.11 class\\_unregister](#)

[\[edit\]](#)

```

void class_unregister(struct class *cls);

```

### [19.2.12 class\\_get](#)

[\[edit\]](#)

```
struct class * class_get(struct class *cls);
```

### [19.2.13 class\\_put](#)

[\[edit\]](#)

```
void class_put(struct class *cls);
```

### [19.2.14 struct class\\_attribute](#)

[\[edit\]](#)

```
struct class_attribute {  
    struct attribute attr;  
    ssize_t (*show)(struct class *, char *buf);  
    ssize_t (*store)(struct class *, const char *buf, size_t count);  
};
```

### [19.2.15 CLASS\\_ATTR](#)

[\[edit\]](#)

```
CLASS_ATTR(name, mode, show, store);
```

### [19.2.16 class\\_create\\_file](#)

[\[edit\]](#)

```
int class_create_file(struct class *cls, const struct class_attribute *attr);
```

### [19.2.17 class\\_remove\\_file](#)

[\[edit\]](#)

```
void class_remove_file(struct class *cls, const struct class_attribute *attr);
```

### [19.2.18 struct class\\_device](#)

[\[edit\]](#)

```
struct class_device {  
    struct list_head node;  
  
    struct kobject kobj;  
    struct class *class;  
    dev_t devt;  
    struct class_device_attribute *devt_attr;  
    struct device *dev;  
    void *class_data;  
  
    char class_id[BUS_ID_SIZE];  
};
```

|            |               |
|------------|---------------|
| node       |               |
| kobj       |               |
| class      |               |
| devt       | sysfs "dev" . |
| devt_attr  |               |
| dev        |               |
| class_data |               |
| class_id   | ID            |

### [19.2.19 class\\_get\\_devdata](#)

[\[edit\]](#)

dev class\_data가 가 class\_set\_data() 가 .

```
void * class_get_devdata(struct class_device *dev);
```

: class\_set\_devdata

### [19.2.20 class\\_set\\_devdata](#)

[\[edit\]](#)

```
void class_set_devdata(struct class_device *dev, void *data);
```

class\_device , (data) class\_device  
class\_data .  
: class\_get\_devdata

### [19.2.21 class\\_device\\_register](#)

[\[edit\]](#)

```
int class_device_register(struct class_device *);
```

### [19.2.22 class\\_device\\_unregister](#)

[\[edit\]](#)

```
void class_device_unregister(struct class_device *);
```

### [19.2.23 class\\_device\\_initialize](#)

[\[edit\]](#)

```
void class_device_initialize(struct class_device *);
```

### [19.2.24 class\\_device\\_add](#)

[\[edit\]](#)

```
int class_device_add(struct class_device *);
```

#### [19.2.25 class\\_device\\_del](#)

[\[edit\]](#)

```
void class_device_del(struct class_device *);
```

#### [19.2.26 class\\_device\\_rename](#)

[\[edit\]](#)

```
int class_device_rename(struct class_device *, char *);
```

#### [19.2.27 class\\_device\\_get](#)

[\[edit\]](#)

```
struct class_device * class_device_get(struct class_device *);
```

#### [19.2.28 class\\_device\\_put](#)

[\[edit\]](#)

```
void class_device_put(struct class_device *);
```

#### [19.2.29 struct class\\_device\\_attribute](#)

[\[edit\]](#)

```
struct class_device_attribute {  
    struct attribute      attr;  
    ssize_t (*show)(struct class_device *, char * buf);  
    ssize_t (*store)(struct class_device *, const char * buf, size_t count);  
};
```

#### [19.2.30 CLASS\\_DEVICE\\_ATTR](#)

[\[edit\]](#)

```
CLASS_DEVICE_ATTR(_name, _mode, _show, _store)
```

#### [19.2.31 class\\_device\\_create\\_file](#)

[\[edit\]](#)

```
int class_device_create_file(struct class_device *,  
                             const struct class_device_attribute *);
```

#### [19.2.32 class\\_device\\_remove\\_file](#)

[\[edit\]](#)

```
void class_device_remove_file(struct class_device *,  
                              const struct class_device_attribute *);
```

### [19.2.33 class\\_device\\_create\\_bin\\_file](#)

[\[edit\]](#)

```
int class_device_create_bin_file(struct class_device *, struct bin_attribute *);
```

### [19.2.34 class\\_device\\_remove\\_bin\\_file](#)

[\[edit\]](#)

```
void class_device_remove_bin_file(struct class_device *, struct bin_attribute *);
```

### [19.2.35 struct class\\_interface](#)

[\[edit\]](#)

```
struct class_interface {
    struct list_head node;
    struct class *class;

    int (*add) (struct class_device *)
    void *(remove) (struct class_device *);
};
```

### [19.2.36 class\\_interface\\_register](#)

[\[edit\]](#)

```
int class_interface_register(struct class_interface *);
```

### [19.2.37 class\\_interface\\_unregister](#)

[\[edit\]](#)

```
void class_interface_unregister(struct class_interface *);
```

## [19.3 class](#)

[\[edit\]](#)

2.6.13 drivers/mtd/mtdchar.c

class

.

### [19.3.1](#)

[\[edit\]](#)

```
static int __init init_mtdchar(void)
{
    if (register_chrdev(...)) {
        ...
    }

    mtd_class = class_create(THIS_MODULE, "mtd");

    if (IS_ERR(mtd_class)) {
        ...
    }
}
```



```

register_mtd_user(&notifier);
return 0;
}

```

class\_create sysfs /sys/class/mtd 가 .

[19.3.2](#) / [\[edit\]](#)

. mtd mtd\_notifier add remove  
class\_device\_create/remove .

```

static void mtd_notify_add(struct mtd_info *mtd)
{
    if (!mtd)
        return;

    class_device_create(mtd_class, MKDEV(MTD_CHAR_MAJOR, mtd->index*2),
                        NULL, "mtd%d", mtd->index);
    class_device_create(mtd_class, MKDEV(MTD_CHAR_MAJOR, mtd->index*2+1),
                        NULL, "mtd%dro", mtd->index);
}

static void mtd_notify_remove(struct mtd_info *mtd)
{
    if (!mtd)
        return;

    class_device_destroy(mtd_class, MKDEV(MTD_CHAR_MAJOR, mtd->index*2));
    class_device_destroy(mtd_class, MKDEV(MTD_CHAR_MAJOR, mtd->index*2+1));
}

static struct mtd_notifier notifier = {
    .add = mtd_notify_add,
    .remove = mtd_notify_remove,
};

```

add remove register\_mtd\_user unregister\_mtd\_user .

[19.3.3](#) [\[edit\]](#)

```

static void __exit cleanup_mtdchar(void)
{
    unregister_mtd_user(&notifier);
    class_destroy(mtd_class);
    unregister_chrdev(MTD_CHAR_MAJOR, "mtd");
}

```

|                     |            |        |                 |     |                        |
|---------------------|------------|--------|-----------------|-----|------------------------|
| unregister_mtd_user | notifier   | remove | /sys/class/mtd/ | mtd |                        |
| class_destroy       | /sys/class | mtd    |                 |     | <a href="#">[edit]</a> |

20 hotplug

|             |               |                |         |        |                        |
|-------------|---------------|----------------|---------|--------|------------------------|
| LDM hotplug | hotplug event | API            | hotplug | /sbin/ |                        |
|             | /sbin/hotplug | /etc/hotplug.d |         |        | <a href="#">[edit]</a> |

21 udev

|      |                                                                                                        |                 |  |   |                        |
|------|--------------------------------------------------------------------------------------------------------|-----------------|--|---|------------------------|
| 2.3  | devfs                                                                                                  |                 |  |   |                        |
|      | devfs                                                                                                  |                 |  |   |                        |
|      | 2.5                                                                                                    | udev            |  |   |                        |
| udev | udev                                                                                                   |                 |  |   |                        |
| udev | :                                                                                                      |                 |  |   |                        |
|      | <ul style="list-style-type: none"><li>/dev</li><li></li><li>(sysfs) API</li></ul>                      |                 |  |   |                        |
|      | <ul style="list-style-type: none"><li>udev - /dev</li><li>namedev -</li><li>libsysfs - sysfs</li></ul> | common API      |  |   |                        |
| udev | sysfs hotplug                                                                                          | 가 , / / / event |  |   |                        |
|      | 가 2.6                                                                                                  | 가 가 ,           |  | 가 | <a href="#">[edit]</a> |

22 firmware

|          |   |     |       |  |                        |
|----------|---|-----|-------|--|------------------------|
| firmware | 가 | 2.6 | sysfs |  | <a href="#">[edit]</a> |
| firmware |   |     |       |  |                        |

## 22.1 firmware API

firmware `<linux/firmware.h>` .

|                         |  |
|-------------------------|--|
|                         |  |
| request_firmware        |  |
| request_firmware_nowait |  |
| release_firmware        |  |

[\[edit\]](#)

### 22.1.1 struct fw

firmware `struct fw` 가 .

```
struct firmware {
    size_t size;
    u8 *data;
};
```

### 22.1.2 request\_firmware

[\[edit\]](#)

sysfs .

```
int request_firmware(const struct firmware **fw, char *name, struct device *device);
```

firmware `fw` . firmware `name` . device .

firmware `fw->data` , `fw->size` 가 .

### 22.1.3 request\_firmware\_nowait

[\[edit\]](#)

request\_firmware firmware 가 , request\_firmware\_nowait .

```
int request_firmware_nowait(struct module *module, char *name, struct device *device,
                           void *context, void (*cont)(const struct firmware *fw,
                                                           void *context));
```

### 22.1.4 release\_firmware

[\[edit\]](#)

request\_firmware firmware .

```
void release_firmware(struct firmware *fw);
```

## 22.2 firmware

[\[edit\]](#)

`/sys/class/firmware` 가 .  
 loading, data, device 가 .  
 loading firmware 0 , -1, .  
 data firmware write .  
 device `/sys/device` .  
 firmware timeout , `/sys/class/firmware/timeout`  
 timeout .

## 23 References

[\[edit\]](#)

- <http://lwn.net/Articles/driver-porting/>
- <http://www.win.tue.nl/%7Eaeb/linux/lk/lk-12.html>
- [Nested classes](http://lwn.net/Articles/154557/) (<http://lwn.net/Articles/154557/>)

Retrieved from <http://www.fst.or.kr/wiki/wiki.php/KelpLdm>  
 last modified 2005-11-02 02:01:25