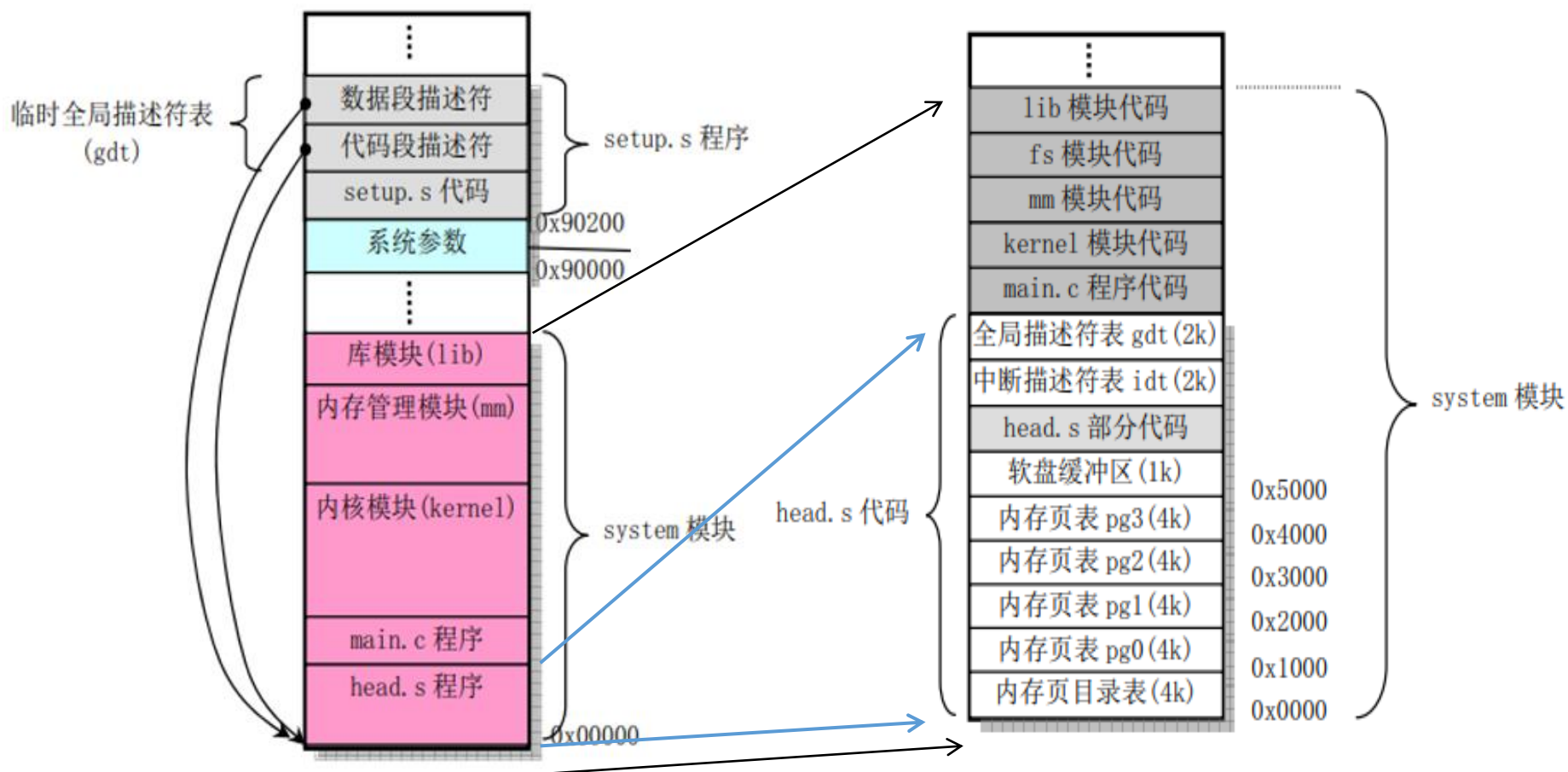
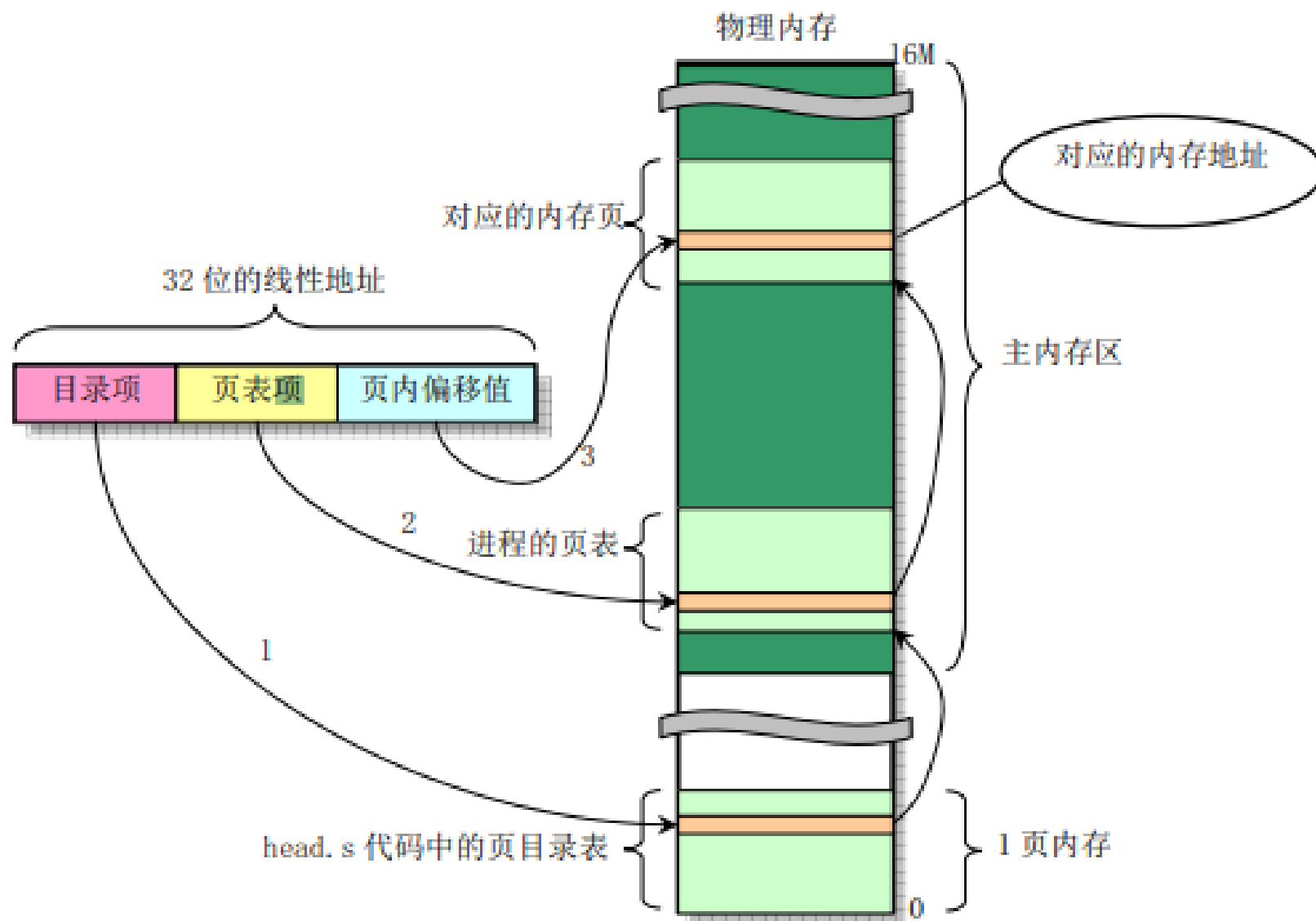


\boot\head.s

\init\main.c

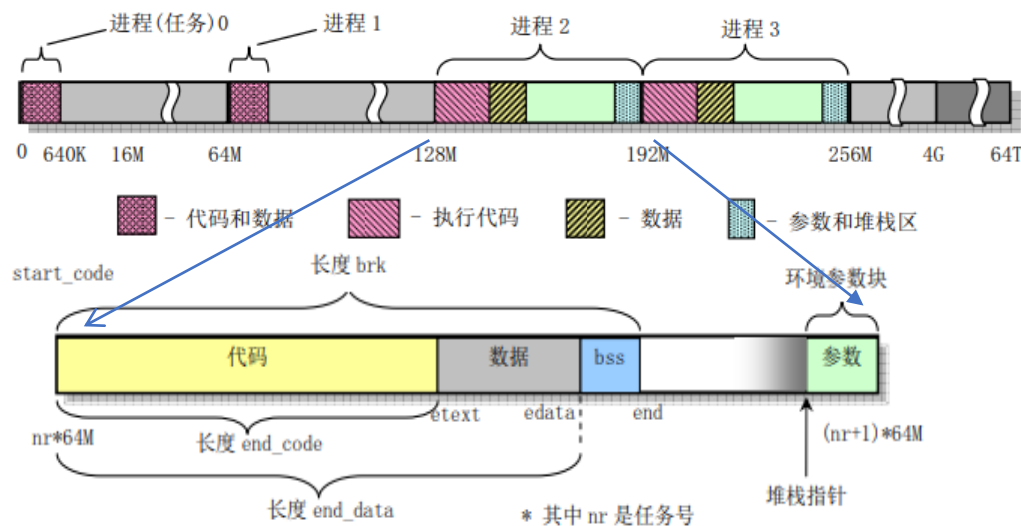
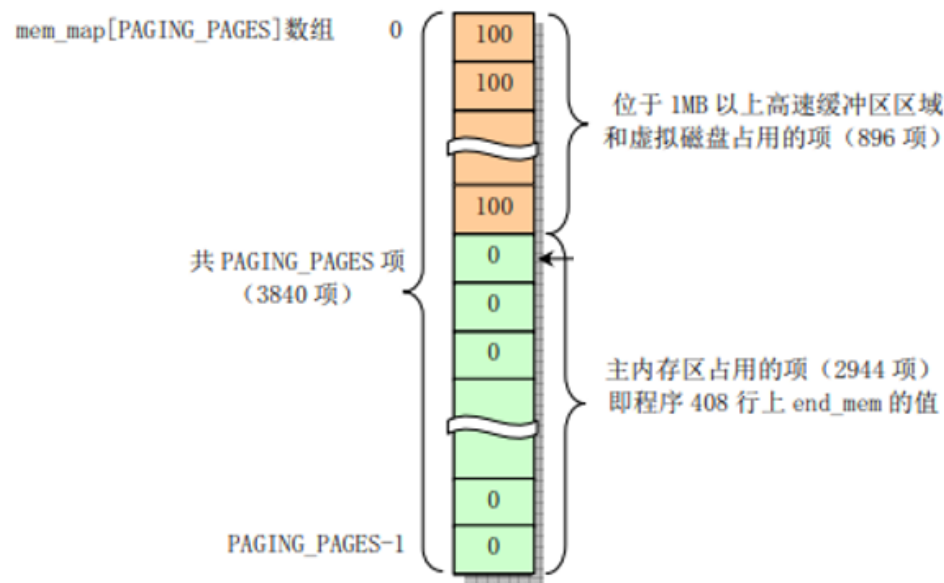
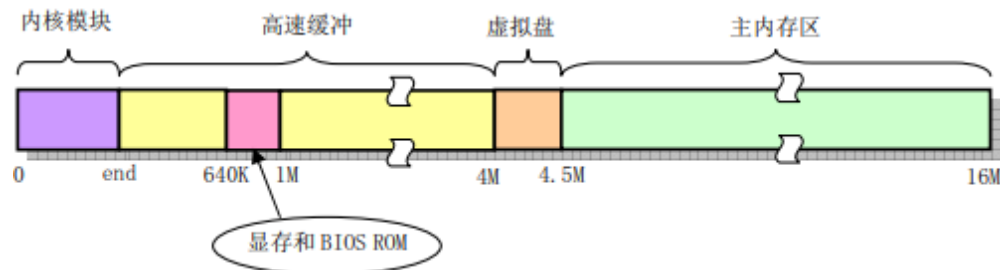
\boot\head.s






\init\main.c

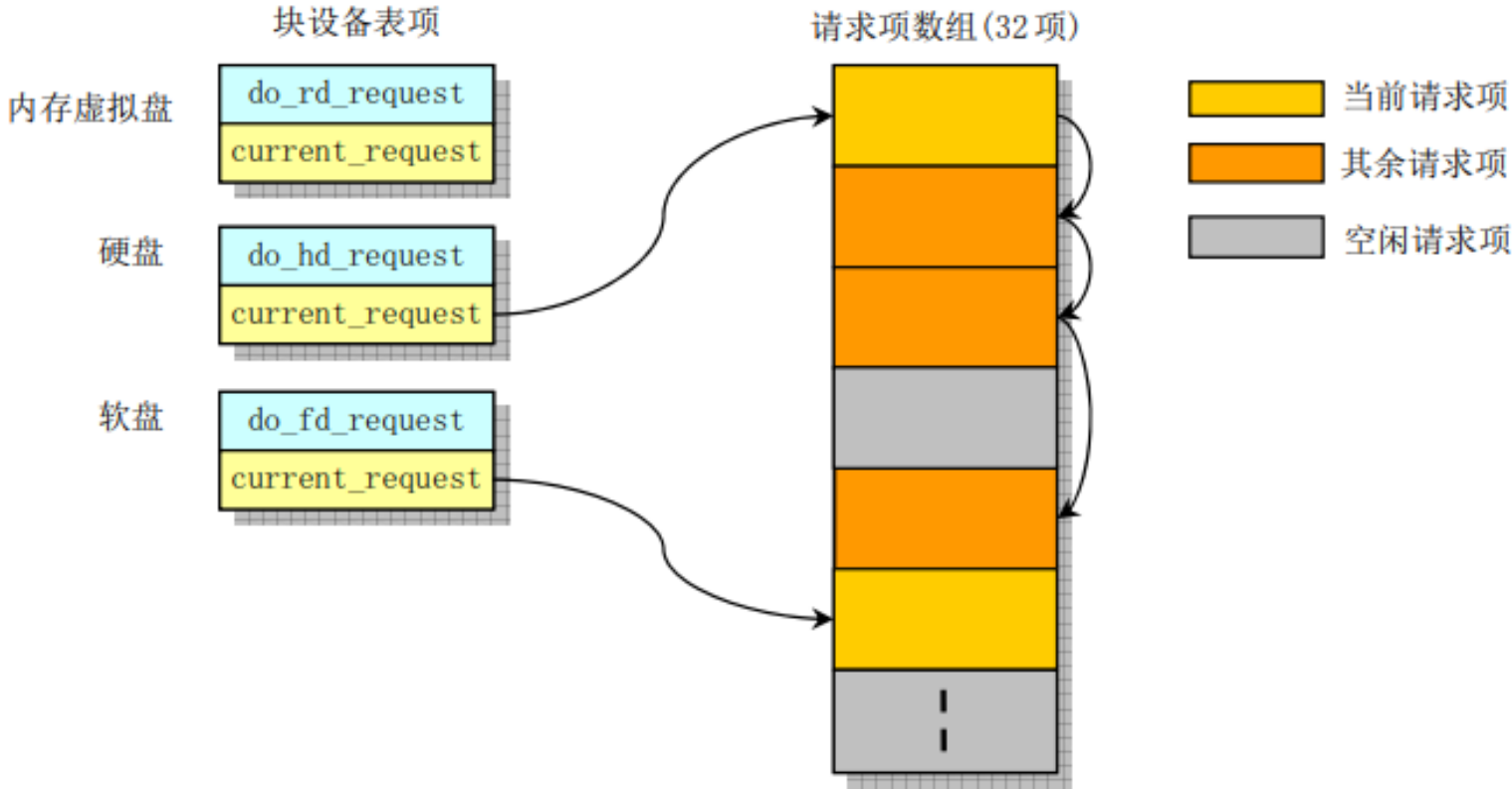
```
mem_init(main_memory_start, memory_end) ;
trap_init() ;
blk_dev_init() ;
chr_dev_init() ;
tty_init() ;
time_init() ;
sched_init() ;
buffer_init(buffer_memory_end) ;
hd_init() ;
floppy_init() ;
sti() ;
move_to_user_mode() ;
if (!fork()) {          /* we count on t
    init() ;
}
```





```
mem_init(main_memory_start, memory_end) ;
trap_init() ;
blk_dev_init() ;
chr_dev_init() ;
tty_init() ;
time_init() ;
sched_init() ;
buffer_init(buffer_memory_end) ;
hd_init() ;
floppy_init() ;
sti() ;
move_to_user_mode() ;
if (!fork()) {          /* we count on t
    init() ;
}
```

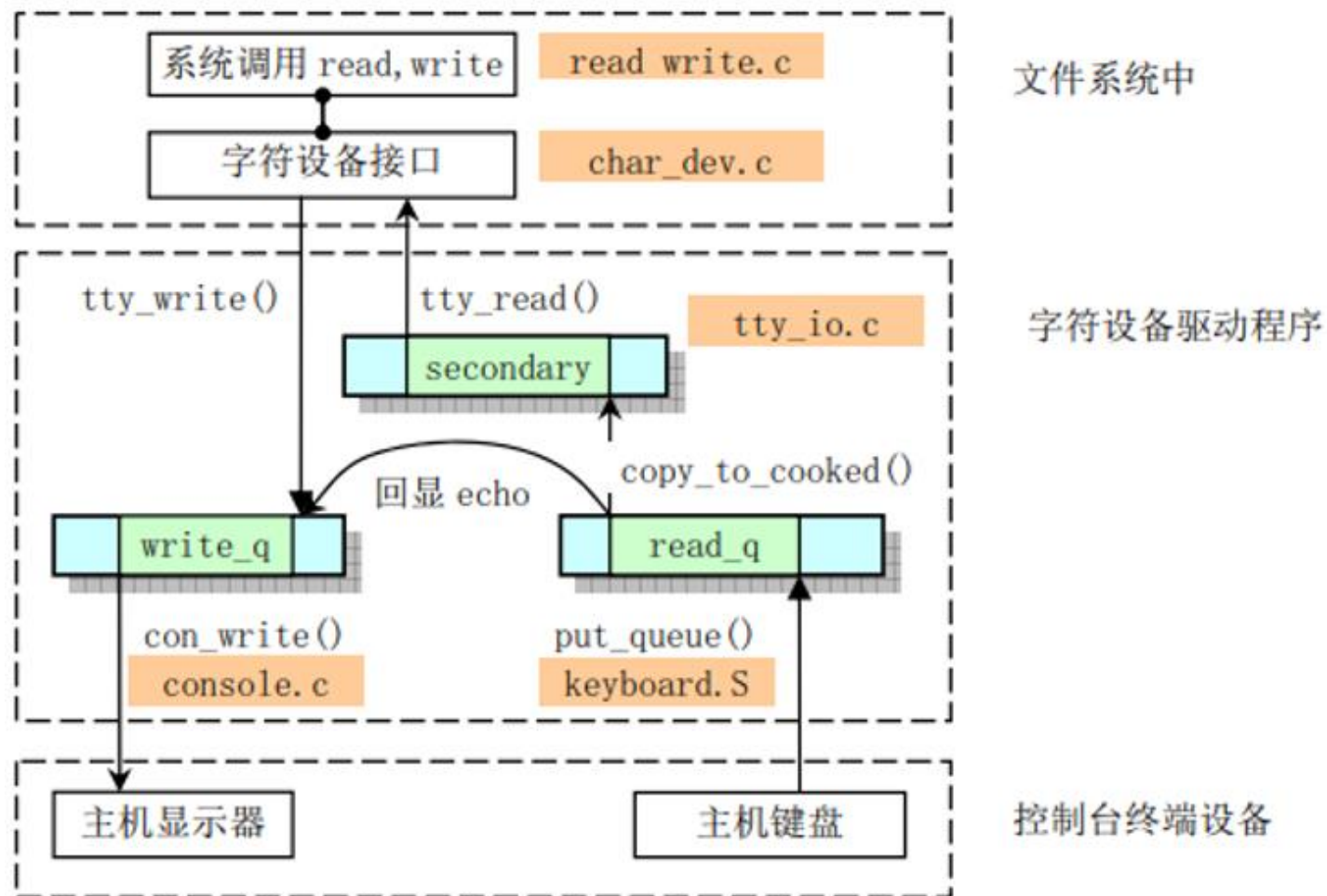
```
struct blk_dev_struct {
    void (*request_fn) (void);
    struct request * current_request;
};
```




```

struct tty_struct {
    struct termios termios;
    int pgrp;
    int stopped;
    void (*write)(struct tty_struct * tty);
    struct tty_queue read_q;
    struct tty_queue write_q;
    struct tty_queue secondary;
};

```




```
mem_init(main_memory_start, memory_end) ;  
trap_init() ;  
blk_dev_init() ;  
chr_dev_init() ;  
tty_init() ;  
time_init() ;  
sched_init() ;  
buffer_init(buffer_memory_end) ;  
hd_init() ;  
floppy_init() ;  
sti() ;  
move_to_user_mode() ;  
if (!fork()) {          /* we count on t  
    init() ;  
}
```

```

struct task_struct {
/* these are hardcoded - don't touch */
    long state; /* -1 unrunnable, 0 runnable, >0 stopped */
    long counter;
    long priority;
    long signal;
    struct sigaction sigaction[32];
    long blocked; /* bitmap of masked signals */
/* various fields */
    int exit_code;
    unsigned long start_code, end_code, end_data, brk, start_stack;
    long pid, father, pgrp, session, leader;
    unsigned short uid, euid, suid;
    unsigned short gid, egid, sgid;
    long alarm;
    long utime, stime, cutime, cstime, start_time;
    unsigned short used_math;
/* file system info */
    int tty; /* -1 if no tty, so it must be signed */
    unsigned short umask;
    struct m_inode * pwd;
    struct m_inode * root;
    struct m_inode * executable;
    unsigned long close_on_exec;
    struct file * filp[NR_OPEN];
/* ldt for this task 0 - zero 1 - cs 2 - ds&ss */
    struct desc_struct ldt[3];
/* tss for this task */
    struct tss_struct tss;
};

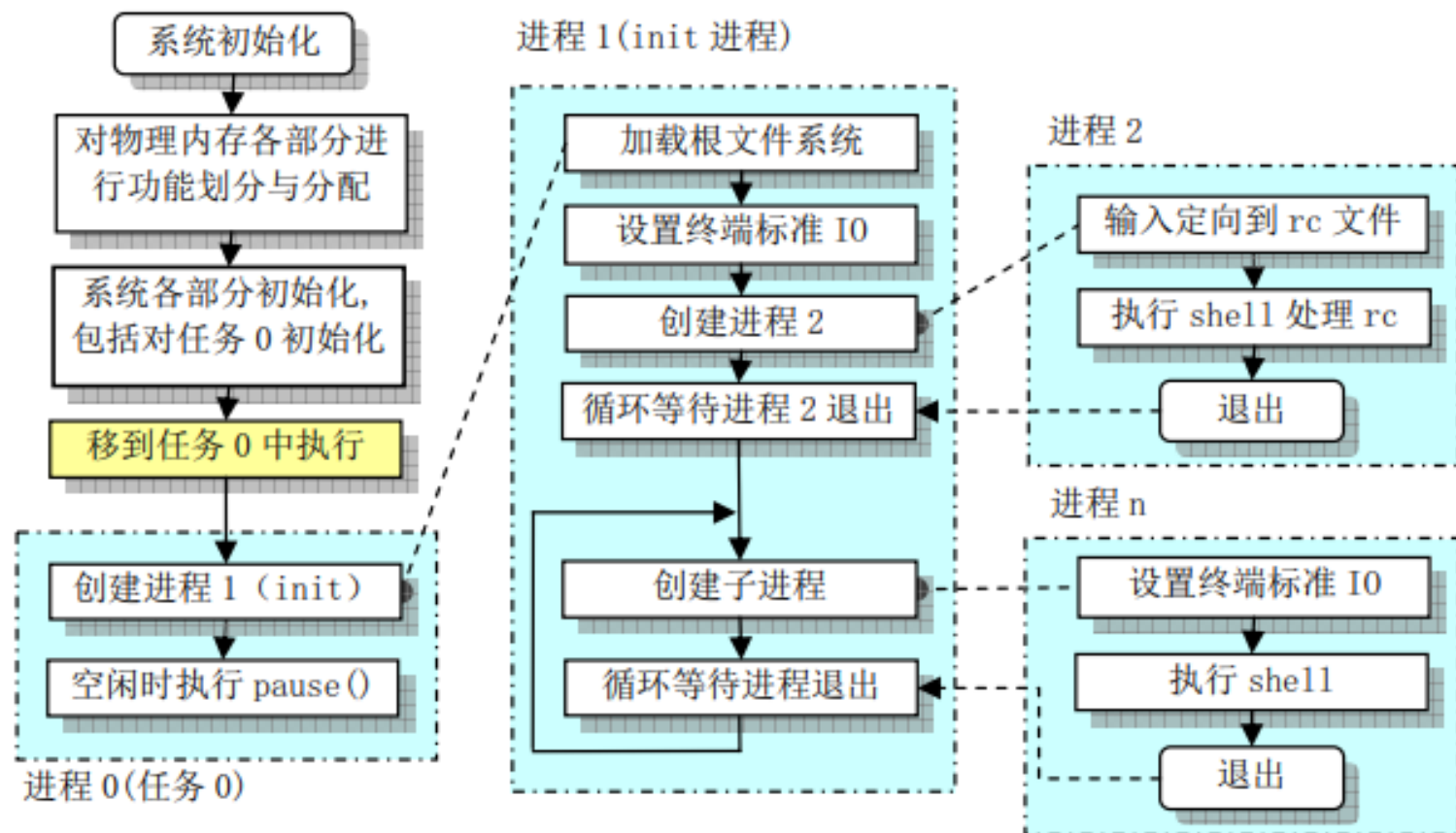
struct tss_struct {
    long back_link;
    long esp0;
    long ss0;
    long esp1;
    long ss1;
    long esp2;
    long ss2;
    long cr3;
    long eip;
    long eflags;
    long eax, ecx, edx, ebx;
    long esp;
    long ebp;
    long esi;
    long edi;
    long es;
    long cs;
    long ss;
    long ds;
    long fs;
    long gs;
    long ldt;
    long trace_bitmap;
    struct i387_struct i387;
};

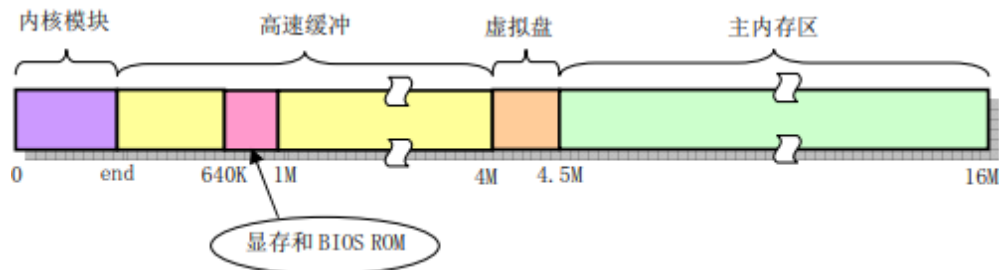
#define INIT_TASK \
/* state etc */ { 0, 15, 15, \
/* signals */ 0, {}, 0, \
/* ec, brk... */ 0, 0, 0, 0, 0, 0, \
/* pid etc.. */ 0, -1, 0, 0, 0, \
/* uid etc */ 0, 0, 0, 0, 0, 0, \
/* alarm */ 0, 0, 0, 0, 0, 0, \
/* math */ 0, \
/* fs info */ -1, 0022, NULL, NULL, NULL, 0, \
/* filp */ {NULL,}, \
    { \
        {0, 0}, \
/* ldt */ {0x9f, 0xc0fa00}, \
        {0x9f, 0xc0f200}, \
    }, \
/*tss*/ {0, PAGE_SIZE+(long)&init_task, 0, \
        0, 0, 0, 0, 0, 0, 0, \
        0, 0, 0x17, 0x17, 0x17, 0x17, 0x17, 0x17, \
        _LDT(0), 0x80000000, \
        {} \
    }, \
}

```



```
mem_init(main_memory_start, memory_end) ;
trap_init() ;
blk_dev_init() ;
chr_dev_init() ;
tty_init() ;
time_init() ;
sched_init() ;
buffer_init(buffer_memory_end) ;
→ hd_init() ;
floppy_init() ;
sti() ;
move_to_user_mode() ;
if (!fork()) {          /* we count on t
    init() ;
}
```





Bochs for Windows - Display

Bochs BIOS, 1 cpu, \$Revision: 1.138 \$ \$Date: 2005/05/07 15:55:26 \$

Booting from Floppy...

Loading system ...

Ram disk: 2097152 bytes, starting at 0x400000

Loading 1048576 bytes into ram disk... 0000kUnable to send byte to FDC

Reset-floppy called done

10/1024 free blocks

286/341 free inodes

3462 buffers = 3545088 bytes buffer space

Free mem: 10485760 bytes

=== Ok.

[/usr/root]# cd /etc

[/etc]# ls

group magic mtab passwd profile rc termcap update

[/etc]# cat rc

/etc/update &

echo "=== Ok. "

[/etc]# _

CTRL + 3rd button enables mouse

A: NUM CAPS SCRL