[2][1][3]

# 西南林业大学

# 本科毕业（设计）论文

(二〇一七届)

**题　　目：** 一个简单操作系统的实现 — RongOS

**分院系部：** 计算机与信息科学学院

**专　　业：** 计算机科学与技术专业

**姓　　名：** 蒲启元

**导师姓名：** 王晓林

**导师职称：** 讲师

**二〇一七年六月**

# 一个简单操作系统的实现 — RongOS

蒲启元

（西南林业大学 计算机与信息科学学院，云南 昆明 650224）

**摘　要：** 操作系统管理着计算机的硬件和软件资源，它是向上层应用软件提供服务（接口）的核心系统软件，这些服务包括进程管理，内存管理，文件系统，网络通信，安全机制等。操作系统的设计与实现则是软件工业的基础与内核。为此，在国务院提出的《中国制造 2025》中专门强调了操作系统的开发。但长期以来，操作系统核心开发技术都掌握在外国人手中，技术受制，对于我们的软件工业来说很不利。本文拟从零开始设计开发一个简单的操作系统，包括 boot loader，中断，内存管理，图形接口，多任务，以及在这个系统上的几个小应用等。尽管这个系统很简单，但它为自主开发操作系统做了一个小小的尝试。

**关键词：** 操作系统，开发，自主

# The implement of a simple OS — RongOS

Qiyuan Pu

School of Computer and Information Science
Southwest Forestry University
Kunming 650224, Yunnan, China

**Abstract:** Operating system manages the sources of hardware and software, it lie in the core of the system software and provide service(interface) to upper application. These service including process management, memory management, file system, network communication, security mechanism etc. The design and implement of operating system is the foundation and core of software industry. Therefore, «Made in China 2025» emphasize the development of operating system that put forward by The State Council. For a long time, however, the kernel development technology grasped in the hand of foreigner, it's bad for our software industry cause of limited technology. So this article will design and develop a simple operating system, including boot loader, interrupt, memory management, graphic interface, multitasking, and some little application depend on this system. In spite of the simple of this system, it's a small trying for autonomous development operating system.

**Key words:** operating system, development, autonomous

# 目　录

# 插图目录

# 表格目录

# 1 Chapter — Preliminary Works

## 1.1 Development Environment

Operating System: Debian 4.11.0-1-amd64

Debug System: QEMU emulator version 2.8.1(Debian 1:2.8+dfsg-7)

Emacs version: GNU Emacs 25.2.2

## 1.2 Tools

Some tools used to develop RongOS, see tools.[1].

## 1.3 Install

Debian System: there is a small tutorial.[2]

QEMU, for my x86_64 architecture:

```
$ sudo apt-get install qemu-system-x86_64
```

Note that the tools is exe formate, so on Debian system, you need to install wine:

```
$ sudo apt-get update
$ sudo apt-get install wine
```

Maybe you also need to add i386 architecture cause of AMD64 on your machine to use these tools:

```
$ sudo dpkg --add-architecture i386
$ sudo apt-get update
```

[1] https://github.com/Puqiyuan/RongOS/tree/master/Tools

[2] http://cs2.swfc.edu.cn/~wx672/lecture_notes/linux/install.html

# 2 Chapter — Boot Loader

## 2.1 Chose Disk

There are many ways to boot a operating system, from hard disk, USB, floppy disk etc. I chose floppy disk, although it is out of date. For my purpose is that develop a simple operating system, pay my attention on how to development. The structure of floppy disk is simple and for my simple operating system it's enough.

## 2.2 The Structure of Floppy Disk

This picture show the inside of floppy disk:

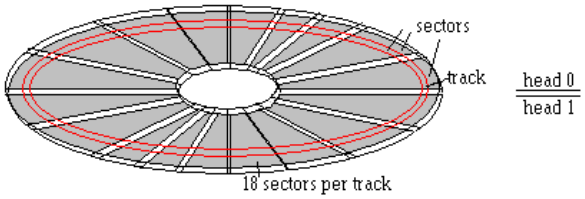


图 2-1 Floppy Disk Structure 1

## 2.3 The Source Codes and Comments of Boot Loader

```
1         CYLS equ 10
2
3 org 0x7c00
4         jmp entry
5         ; The next codes specify the format of standard FAT12 floppy disk.
6 db 0x90 ; db is the abbreation of "define byte", it literally places that byt
7         ; right there in the executable.
8 db "RONGBOOT" ; The name of boot sector, muse be 8 byte.
```

```
9   dw 512
10  db 1
11  dw 1
12  db 2
13  dw 224
14  dw 2880
15  db 0xf0
16  dw 9
17  dw 18
18  dw 2
19  dd 0
20  dd 2880
21  db 0,0,0x29
22  dd 0xffffffff
23  db "RONGBOOTOS "
24  db "FAT12   "
25  resb 18
26
27
28  entry:
29          mov ax, 0
30          mov ss, ax
31          mov sp, 0x7c00
32          mov ds, ax
33
34          mov si, msg_init
35          jmp init
36
37
38  init:
39          mov al, [si]
```

```
40          add si, 1
41          cmp al, 0
42          je load
43          mov ah, 0x0e
44          mov bx, 15
45          int 0x10
46          jmp init
47
48
49  msg_init:
50  db 0x0a
51  db 0x0d
52  db "Copyright: GPL"
53  db 0x0a
54  db 0x0d
55  db "Author: Qiyuan Pu"
56  db 0x0a
57  db 0x0d
58  db "https://github.com/Puqiyuan/RongOS"
59  db 0x0a
60  db 0x0d
61  db "IPL is loading, please waiting..."
62  db 0x0a
63  db 0x0d
64  db "......"
65
66
67  load:
68
69          mov ax, 0
70
```

```
71          mov ax, 0x0820
72          mov es, ax
73          mov ch, 0
74          mov dh, 0
75          mov cl, 2
76
77  readloop:
78          mov si, 0
79
80  retry:
81          mov ah, 0x02
82          mov al, 1
83          mov bx, 0
84          mov dl, 0x00
85          int 0x13
86          jnc next
87          add si, 1
88          cmp si, 5
89          jae error
90          mov ah, 0x00
91          mov dl, 0x00
92          int 0x13
93          jmp retry
94
95  next:
96          mov ax, es
97          add ax, 0x0020
98          mov es, ax
99          add cl, 1
100         cmp cl, 18
101         jbe readloop
```

```
102         mov cl, 1
103         add dh, 1
104         cmp dh, 2
105         jb readloop
106         mov dh, 0
107         add ch, 1
108         cmp ch, CYLS
109         jb readloop
110         jmp correct
111
112
113 fin:
114         hlt
115         jmp fin
116
117
118 error:
119         mov si, msg
120
121
122 correct:
123         mov si, msg_corr
124
125
126 putloop:
127         mov al, [si]
128         add si, 1
129         cmp al, 0
130         mov [0x0ff0], ch
131         je 0xc200
132         mov ah, 0x0e
```

```
133         mov bx, 15
134         int 0x10
135         jmp putloop
136
137
138 msg_corr:
139 db 0x0a
140 db 0x0d
141 db 0x0a
142 db 0x0d
143 db "OK: IPL loaded"
144 db 0x0a
145 db 0x0d
146 db 0
147
148
149 msg:
150 db 0x0a
151 db "IPL load error"
152 db 0x0a
153 db 0
154 resb 0x7dfe-$
155
156
157 db 0x55, 0xaa
```

# 参考文献

[1] 国务院。*中国制造 2025*，2015-05。。

[2] WiKipedia. *Operating System*, 2017-08..

[3] 川合秀实. *30 天自制操作系统*. 人民邮电出版社, 2012-08.

# 指导教师简介

指导教师简介（约百余字）

# 致　谢

首先我想感谢我的老师，王晓林。大学期间，他给了我很多指导，包括专业方面和上大学的意义等。很多时候，他对学生的要求看起来都是不近情理的，但正是通过这个“痛苦”的过程，我锻炼了坚强的意志，和战胜困难的信心。谢谢你，王老师。我最想感谢的是我的女友，她容忍我在完成这个设计时的很多个夜晚不陪她，给我支持，鼓励我，不抱怨。所以我愿意把这个简单操作系统命名为 RongOS, 蓉便是她名字的最后一个字。谢谢你，我最亲爱的。