

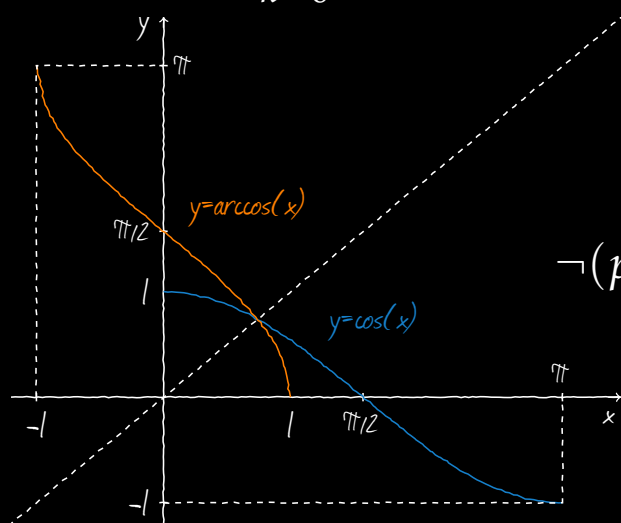
ZJUNIX

实验操作流程

浙江大学

2017.08.20

$$(a+b)^n = \sum_{k=0}^n \binom{n}{k} a^k b^{n-k}$$

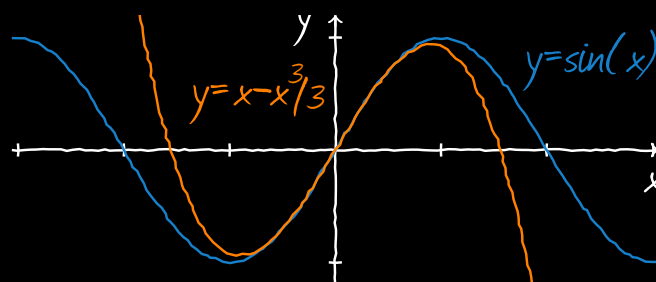


$$\zeta_k = |a|^{1/n} e^{i(\arg(a) + 2k\pi)/n}$$

$$e^{i\pi} + 1 = 0$$

$$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$$

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$



1.1	实验目的	2
1.2	实验步骤	2
	目录结构 – 新增的代码 – 修改的代码	
1.3	预期结果	5

1.1 实验目的

- 理解 ZJUNIX 操作系统中对于异常的处理
- 编写异常机制初始化代码
- 编写系统调用初始化代码
- 测试系统调用的可用性

1.2 实验步骤

1.2.1 目录结构

代码 1.1: 实验 4 目录结构

```
exp4
|-- arch
|   |-- Makefile
|   |-- mips32
|       |-- arch.c
|       |-- arch.h
|       |-- exc.c
|       |-- exc.h
|       |-- intr.c
|       |-- intr.h
|       |-- Makefile
|       |-- start.s
|-- include
|   |-- assert.h
|   |-- driver
|       |-- ps2.h
|       |-- sd.h
|       |-- vga.h
|   |-- init_place_holder.h
|   |-- zjunix
|       |-- log.h
|       |-- syscall.h
|       |-- time.h
|       |-- utils.h
|-- kernel
|   |-- driver
|       |-- Makefile
|       |-- ps2.c
|       |-- ps2.h
|       |-- sd.c
|       |-- sd.h
|       |-- vga.c
|       |-- vga.h
```

```
--- init.c
--- Makefile
--- syscall
    --- Makefile
    --- syscall4.c
    --- syscall4.h
    --- syscall.c
--- time
    --- Makefile
    --- time.c
    --- time.h
--- Makefile
--- usr
    --- Makefile
    --- ps.c
    --- ps.h
--- utils
    --- assert.c
    --- log.c
    --- Makefile
    --- utils.c
```

1.2.2 新增的代码

从这次实验开始，由于修改与新增的代码众多，难以一一列出每个代码的具体内容，所以这里列出所有新增以及修改过的代码，并对关键代码进行标注。具体的代码可以根据下面的标注，对比 exp3 和 exp4 的工程源码

代码 1.2: 实验 4 新增代码

```
exp4
--- include
    --- assert.h # 运行时断言
    --- driver
        --- ps2.h # 键盘驱动
        --- sd.h # sd卡读写驱动
        --- vga.h # 显示驱动
    --- zjunix
        --- log.h # 日志打印
        --- time.h # 时间操作
        --- utils.h # memset等辅助工具
--- kernel
    --- driver
        --- Makefile
        --- ps2.c
        --- ps2.h
        --- sd.c
        --- sd.h
```

```

|-- vga.c
|-- vga.h
|-- time
|   |-- Makefile
|   |-- time.c
|   |-- time.h
|-- usr
|   |-- Makefile
|   |-- ps.c # 简易 shell
|   |-- ps.h # 简易 shell
|-- utils
|   |-- assert.c
|   |-- log.c
|   |-- Makefile
|   |-- utils.c
```

在新加入的代码中，比较重要的部分有

代码	说明
kernel/driver/*	一些重要的输入输出设备驱动
kernel/time/*	时间操作模块
usr/ps.c	一个简单的 shell，可以检验输入输出设备驱动的可用性，同时提供一些测试命令
utils/assert.c	运行时断言，即运行时如果有一个条件不满足，则使系统停机，在开发阶段有助于避免一些错误
utils/log.c	日志输出功能，开机启动时会调用
utils/utils.c	一些工具函数，例如 memset 等

表 1.1: 关键代码说明

1.2.3 修改的代码

代码 1.3: 实验 4 修改的代码

```

exp4
|-- include
|   |-- init_place_holder.h
|-- kernel
|   |-- init.c
|   |-- Makefile
|-- Makefile
```

1.3 预期结果

根据 init.c 的代码，在系统启动时会在屏幕上输出一些日志信息，然后等待按键，准备进入 shell

根据 shell 代码 usr/ps.c，在 shell 中可运行的命令如表 1.2所示

命令	说明
clear	清屏
echo	输出字符
gettime	打印当前时间
syscall4	调用 4 号系统调用，点亮 LED 灯的低 8 位
sdwi	向 SD 卡第 7 扇区写入非 0 数据
sdwz	向 SD 卡第 7 扇区写入 0 数据
sdr	读取 SD 卡第 7 扇区

表 1.2: 当前可运行命令