

任务 23：看门狗

课程名称： 嵌入式 实验类型： 综合

实验项目名称： 任务 23：看门狗

学生姓名： 刘婉颐 专业： 计科 1201 学号： 3120100485

同组学生姓名： _____ 指导老师： 翁恺

实验地点： 寝室 实验日期： 2015 年 3 月 28 日

看门狗

配置内核中的硬件看门狗，使得一定时间内不喂狗就重启 Acadia 或 RPi 或 WRTnode，写一个程序或脚本保持一定频率的喂狗，当关闭这个程序或脚本时形成重启。实验报告要记录和表现出重启。

实现目的

1. 掌握看门狗的概念；
2. 掌握 Acadia 或 RPi 或 WRTnode 上编写看门狗程序的方法。

实验器材

硬件

- Acadia 或 RPi 或 WRTnode 板一块；
- 5V/1A 电源一个；
- microUSB 线一根；
- USB-TTL 串口线一根（FT232RL 芯片或 PL2303 芯片）。

以下为自备（可选）器材：

- PC（Windows/Mac OS/Linux）一台；
- 以太网线一根（可能还需要路由器等）。

软件

- PC 上的 USB-TTL 串口线配套的驱动程序;
- PC 上的串口终端软件, 如 minicom、picocom、putty 等;
- PC 上的 SSH 软件, 如 putty 等。

实验步骤

1. 了解看门狗:

看门狗是一个定时器电路, 一般有一个输入, 一个输出到 MCU 的 RST 端, MCU 正常工作的时候, 每隔一段时间输出一个信号到喂狗端, 给 WDT 清零, 如果超过规定的时间不喂狗, (一般在程序跑飞时), WDT 定时超过, 就会给出一个复位信号到 MCU, 使 MCU 复位. 防止 MCU 死机. 看门狗的作用就是防止程序发生死循环, 或者说程序跑飞。

查看/dev 目录, 看到里面的 watchdog 即为字符设备。

```
ptmx      ptyqd  ptyxc  tty37      ttyed  ttyva  video17
pts       ptyqe  ptyxd  tty38      ttyee  ttyvb  video18
ptya0     ptyqf  ptyxe  tty39      ttyef  ttyvc  video19
ptya1     ptyr0  ptyxf  tty4       ttymxc0 ttyvd  video20
ptya2     ptyr1  ptyy0  tty40      ttymxc2 ttyve  watchdog
ptya3     ptyr2  ptyy1  tty41      ttyp0  ttyvf  zero
ptya4     ptyr3  ptyy2  tty42      ttyp1  ttyw0
ptya5     ptyr4  ptyy3  tty43      ttyp2  ttyw1
ptya6     ptyr5  ptyy4  tty44      ttyp3  ttyw2
```

可以重新启动 watchdog:

```
root@Acadia:~# /etc/init.d/watchdog stop
* Stopping watchdog daemon... [ OK ]
* Starting watchdog keepalive daemon... [ OK ]
root@Acadia:~# /etc/init.d/watchdog start
* Stopping watchdog keepalive daemon... [ OK ]
* Starting watchdog daemon... [ OK ]
root@Acadia:~#
```

2. 编辑喂狗脚本:

```
root@Acadia:~/dog# nano feed_dog.c
```

程序内容如下:

```

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
int main(void)
{
    int fd=open("/dev/watchdog",O_WRONLY);
    int ret=0;
    if(fd==-1)
    {
        perror("open error");
        exit(EXIT_FAILURE);
    }
    while(1)
    {
        printf("feed dog\n");
        ret=write(fd,"1",1);
        if(ret!=1)
        {
            ret=-1;
            break;
        }
        //ret=fsync(fd);

```

```

//ret=fsync(fd);
//if(ret)
//    break;
sleep(10);
}
close(fd);
return ret;
}

```


3. 编译运行喂狗程序:

```

root@Acadia:~/dog# gcc -o feed_dog feed_dog.c
root@Acadia:~/dog# ./feed_dog
I'm feeding the dog!!!
I'm feeding the dog!!!
I'm feeding the dog!!!
I'm feeding the dog!!!
I'm feeding the dog!!!
I'm feeding the dog!!!
I'm feeding the dog!!!
I'm feeding the dog!!!
^Cimx2-wdt imx2-wdt.0: Unexpected close: Expect reboot!
root@Acadia:~/dog# 

```

4. Ctrl+C 中断程序后等待一分钟，pcDuino 进行重启:

```
I'm feeding the dog!!!  
I'm feeding the dog!!!  
^Cimx2-wdt imx2-wdt.0: Unexpected close: Expect reboot!  
  
root@Acadia:~/dog#   
  
U-Boot 2009.08 (Dec 25 2014 - 21:37:33)  
  
CPU: Freescale i.MX6 family T01.2 at 792 MHz  
Thermal sensor with ratio = 170  
Temperature: 35 C, calibration data 0x5334ad69  
mx6q pll1: 792MHz  
mx6q pll2: 528MHz  
mx6q pll3: 480MHz  
mx6q pll8: 50MHz  
ipg clock : 660000000Hz  
ipg per clock : 660000000Hz  
uart clock : 800000000Hz  
cspi clock : 600000000Hz
```

5. 验证结果:

喂狗程序可以启动看门狗，监测喂狗程序是否正常执行，如果超过一段时间没有喂狗，就自动重启系统。