SigmaStar Camera Watchdog 使用参考

© 2019 SigmaStar Technology Corp. All rights reserved.

SigmaStar Technology makes no representations or warranties including, for example but not limited to, warranties of merchantability, fitness for a particular purpose, non-infringement of any intellectual property right or the accuracy or completeness of this document, and reserves the right to make changes without further notice to any products herein to improve reliability, function or design. No responsibility is assumed by SigmaStar Technology arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights, nor the rights of others.

SigmaStar is a trademark of SigmaStar Technology Corp. Other trademarks or names herein are only for identification purposes only and owned by their respective owners.

REVISION HISTORY

Revision No.	Description	Date
{000001}	• {Initial release}	{07/28/2018}

Copyright © 2019 SigmaStar Technology Corp. All rights reserved.

TABLE OF CONTENTS

REVISION HISTORY ..... 错误!未定义书签。
TABLE OF CONTENTS ..... 错误!未定义书签。
1. 概述 ..... 错误!未定义书签。
1.1. 概述 ..... **错误!未定义书签。**
2. WATCHDOG 控制 ..... 错误!未定义书签。
2.1. 打开 WATCHDOG ..... **错误!未定义书签。**
2.2. 关闭 WATCHDOG ..... **错误!未定义书签。**
2.3. 设定 TIMEOUT ..... **错误!未定义书签。**
2.4. KEEP ALIVE ..... **错误!未定义书签。**

Copyright © 2019 SigmaStar Technology Corp. All rights reserved.

1. 概述

1.1. 概述

Watchdog 采用标准的 linux 框架，提供硬件的 watchdog，上层应用可以设定 time out 时间，自己来 keep alive。

Watchdog 默认是关闭的，客户可自行决定是否开启。开启建议在主线程中操作，如果在其他线程中操作，watchdog 会随着线程的关闭而关闭。

Copyright © 2019 SigmaStar Technology Corp. All rights reserved.

2. WATCHDOG 控制

2.1. 打开 WATCHDOG

打开/dev/watchdog 设备，watchdog 将被启动。

参考代码如下：

```
int wdt_fd = -1;
wdt_fd = open("/dev/watchdog", O_WRONLY);
if (wdt_fd == -1)
{
    // fail to open watchdog device
}
```

2.2. 关闭 WATCHDOG

参考代码如下：

```
int option = WDIOS_DISABLECARD;
ioctl(wdt_fd, WDIOC_SETOPTIONS, &option);
if (wdt_fd != -1)
{
    close(wdt_fd);
    wdt_fd = -1;
}
```

2.3. 设定 TIMEOUT

通过标准的 IOCTL 命令 WDIOC_SETTIMEOUT，来设定 timeout，单位是 second，timeout 的时间建议大于 5s，参考代码如下：

```
#define    WATCHDOG_IOCTL_BASE 'W'
#define    WDIOC_SETTIMEOUT          _IOWR(WATCHDOG_IOCTL_BASE, 6, int)
int timeout = 20;
ioctl(wdt_fd, WDIOC_SETTIMEOUT, &timeout);
```

2.4. KEEP ALIVE

通过标准的 IOCTL 命令 WDIOC_KEEPALIVE 来喂狗，喂狗时间按照设定的 timeout 来决定，喂狗时间应该比 timeout 小，参考代码如下：

Copyright © 2019 SigmaStar Technology Corp. All rights reserved.

```
#define     WATCHDOG_IOCTL_BASE 'W'
#define     WDIOC_KEEPALIVE             _IOR(WATCHDOG_IOCTL_BASE, 5, int)
 ioctl(wdt_fd, WDIOC_KEEPALIVE, 0);
```


Copyright © 2019 SigmaStar Technology Corp. All rights reserved.