

目录

目录.....	1
1. 芯片基本信息.....	2
2. 芯片驱动流程.....	2
2.1. 初始化模块:	2
2.2. 中断模块.....	2
2.3. 睡眠模块.....	3
2.4. 唤醒模块.....	3
3. 细节步骤	3
3.1. 清除芯片寄存器: gsl_clear_reg.....	3
3.2. 复位芯片: gsl_reset_core.....	3
3.3. 启动芯片: gsl_start_core	4
3.4. 核查芯片工作状态: check_mem_data.....	4
4. 芯片寄存器介绍.....	4
5. 对触摸点数据介绍:	4
6. 版本修订记录.....	6
7. 联系.....	6

芯片驱动工作流程说明书

1. 芯片基本信息

支持芯片型号	GSL1680、GSL1688、GSL2681、GSL2682、GSL2680、GSL3670、GSL3680
I2C 设备地址	0x40（7 位）
I2C 寄存器地址	8 位
工作电压	2.6V—3.3V
支持 IO 电压	1.8V/2.8V（芯片内部自带 1.8V 输出 IO）

2. 芯片驱动流程

Silead GSL 系列芯片在驱动中，工作的流程可以分为下面几个模块：

a、初始化模块 b、中断模块 c、睡眠模块 d、唤醒模块

2.1. 初始化模块：

初始化模块主要是让芯片正常工作起来。在初始化模块中依次对芯片进行了下面步骤的处理：

- a、拉高电源引脚 VDD。
- b、操作 reset 引脚：拉低 reset 引脚、延迟 20 毫秒、拉高 reset 引脚、延迟 20 毫秒。
- c、清除芯片寄存器：gsl_clear_reg
- b、复位芯片：gsl_reset_core
- d、下载芯片 firmware：gsl_fw_load
- e、启动芯片：gsl_start_core
- f：延迟 20 毫秒，然后核查芯片工作状态：check_mem_data

2.2. 中断模块

中断模块主要是向系统上报触摸点坐标。如果整机对 TP 设备没有中断引脚，也可以采用轮询的方式向系统上报触摸点坐标。中断模块中依次经过下面步骤处理：

- a、读取 0x80 寄存器的 44 个字节。
- b、对读出的 44 个字节进行分析，得到初步触摸点坐标。

c、将分析之后的结果经过 `gsl_alg_id_main` 函数处理，得到具体触摸点坐标。（有的平台可能不需要这步）

d、将触摸点坐标坐标上报给系统。

2.3.睡眠模块

睡眠模块主要为了降低手机在待机状态的漏电流。睡眠模块主要经过下面步骤：

a、拉低 `reset` 引脚。

2.4.唤醒模块

唤醒模块是因为有睡眠模块，所以才有了唤醒模块。在睡眠模块中 `TP` 进入不工作状态，为了让 `TP` 再次能工作起来，所以有了唤醒模块。唤醒模块主要经过下面步骤：

a、拉高 `reset` 引脚，然后延迟 20 毫秒。

b、复位芯片：`gsl_reset_core`

c、启动芯片：`gsl_start_core`

d、延迟 20 毫秒，然后核查芯片工作状态：`check_mem_data`

3. 细节步骤

3.1.清除芯片寄存器：`gsl_clear_reg`

a、向 `0xe0` 写 `0x88`；

b、向 `0x80` 写 `0x1`；

c、向 `0xe4` 写 `0x4`；

d、向 `0xe0` 写 `0x0`；

3.2.复位芯片：`gsl_reset_core`

a、向 `0xe0` 写 `0x88`；

b、向 `0xe4` 写 `0x04`；

c、向 0xbc 写 0x0;

3.3.启动芯片: `gsl_start_core`

a、向 0xe0 写 0x0;

3.4.核查芯片工作状态: `check_mem_data`

读取 0xb0 寄存器的值, 并判断读出来的值是不是 0x5a5a5a5a。如果不是说明芯片没有正常工作起来, 在进行: 拉低 reset 引脚、延迟 20 毫秒、拉高 reset 引脚、延迟 20 毫秒, 清除芯片寄存器, 复位芯片, 下载芯片 firmware, 启动芯片。如果是说明芯片已经正常工作, 则不作任何处理。

4. 芯片寄存器介绍

- 1、0xe0 : 芯片复位寄存器
- 2、0xe4 : 芯片时钟寄存器
- 3、0xbc : 掉电检测寄存器
- 4、0xf0 : 页码寄存器
- 5、0x80 : 触摸点数寄存器
- 6、0x84~0xa8 : 触摸点坐标寄存器
- 7、0xb4 : 芯片内部中断计数寄存器
- 8、0xb0 : 芯片检验码寄存器
- 9、0xac : 靠近关屏寄存器

5. 对触摸点数据介绍:

从芯片 0x80 读回数据 read_buf[44]

Read_buf	Read_buf[0]	Read_buf[1]	Read_buf[2]	Read_buf[3]
描述	触摸点数			



芯片驱动工作流程说明书

第一个触摸点:

Read_buf[4]	Read_buf[5]	Read_buf[6]	Read_buf[7]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位

第二个触摸点:

Read_buf[8]	Read_buf[9]	Read_buf[10]	Read_buf[11]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位

第三个触摸点:

Read_buf[12]	Read_buf[13]	Read_buf[14]	Read_buf[15]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位

第四个触摸点:

Read_buf[16]	Read_buf[17]	Read_buf[18]	Read_buf[19]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位

第五个触摸点:

Read_buf[20]	Read_buf[21]	Read_buf[22]	Read_buf[23]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位

第六个触摸点:

Read_buf[24]	Read_buf[25]	Read_buf[26]	Read_buf[27]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位

第七个触摸点:

Read_buf[28]	Read_buf[29]	Read_buf[30]	Read_buf[31]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位

第八个触摸点:

Read_buf[32]	Read_buf[33]	Read_buf[34]	Read_buf[35]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位

第九个触摸点:

Read_buf[36]	Read_buf[37]	Read_buf[38]	Read_buf[39]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位



芯片驱动工作流程说明书

第十个触摸点:

Read_buf[40]	Read_buf[41]	Read_buf[42]	Read_buf[43]
y 的低 8 位	y 的高 8 位	x 的低 8 位	低 4 位是 x 的高 4 位

6. 版本修订记录

版本号	修订记录
V1.0_20121003	初版
V1.1_20130723	芯片流程更新

7. 联系

上海浦东新区盛夏路 560 号 2 幢 1003 思立微电子 201203

Tel: 021-20221991 Fax: 021-20221996

Email: minor_deng@sileadinc.com