# PT8P2106 触摸库使用手册

1. 概述 ..... 3
2. 适用型号 ..... 3
3. 触摸库架构 ..... 3
4. 触摸库资源占用情况 ..... 3
4.1 ROM/RAM 资源占用 ..... 3
4.2 堆栈占用 ..... 4
4.3 IO 端口占用 ..... 4
5. 触摸库接口说明 ..... 4
5.1 汇编库接口 ..... 4
5.1.1 文件组成 ..... 4
5.1.2 函数名与调用方式 ..... 4
5.1.3 输入数据 ..... 4
5.1.4 输出数据 ..... 4
5.2 C 库接口 ..... 4
5.2.1 文件组成 ..... 4
5.2.2 函数名与调用方式 ..... 5
5.2.3 输入数据 ..... 5
5.2.4 输出数据 ..... 5
6. 阈值设置 ..... 5
7. 二次触摸功能 ..... 5
7.1 可配置参数 ..... 5
8. 低功耗功能 ..... 6
8.1 可配置参数 ..... 6
8.2 低功耗配置流程 ..... 6
9. 按键超时处理 ..... 6
10. 调试输出功能 ..... 6
10.1 可配置参数 ..... 6
10.2 调试串口输出波特率配置对照表 ..... 7
11. 其它可配置参数 ..... 7
12. 开发注意事项 ..... 7
12.1 中断处理 ..... 7
12.2 WDT 说明 ..... 8
13. 应用开发流程示例 ..... 8
14. 版本历史记录 ..... 9

# 1. 概述

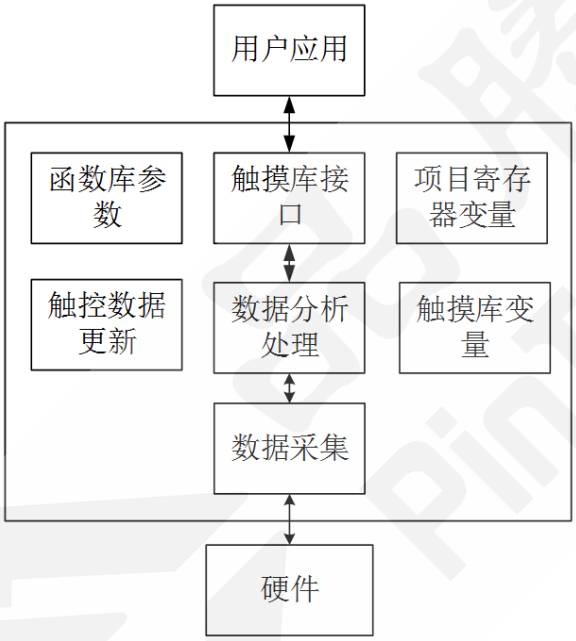
本文档详细介绍 PT8P2106/ PT8P2108 系列触摸函数库的资源占用、功能配置、对外接口与开发流程，为开发者提供触摸功能开发、高度的标准化参考。

# 2. 适用型号

本触摸库支持芯片型号：PT8P2106、PT8P2108、PT8P2108B。

# 3. 触摸库架构

触摸库采用模化函数架构，内置触摸信号采集、算法滤波、低功耗控制、调试输出等功能模块，通过标准化对外接口向上层应用提供稳定的触摸检测能力.




# 4. 触摸库资源占用情况

本章说明触摸库运行时 RAM、ROM、堆栈、中断及 IO 资源占用，用于工程资源评估.

## 4.1 ROM/RAM 资源占用

| 资源类型 | 基本功能 | 低功耗功能 | 二次触摸功能 | REF_EN 开启 | 调试输出开启 |
|---|---|---|---|---|---|
| ROM | 525 | -60 | +80 | +80 | +50 |
| RAM | 19+N*7 | -- | +N*4 | +7 | +4 |

表 1 RAM/ROM 使用情况

注：

1, 汇编模式可用 RAM 地址计算公式：
PT8P2106/PT8P2108: 0x80 ~ (0xEF - 19 + N*7 - (可配置功能 RAM 增量));
PT8P2108B: 0x38 ~ (0x7F - 19 + N*7 - (可配置功能 RAM 增量));
2, N = 已使能的触摸通道数量;
3, 实际 ROM 占用可能存在微小偏差，取决于实际工程配置;
4, C 语言模式下 RAM 空间由链接器自动分配，无需手动计算配置.

### 4.2 堆栈占用
触摸库内部仅一级函数调用逻辑，运行时堆栈开销极低。

### 4.3 IO 端口占用
触摸通道 IO 可按需配置，最大支持 4 路独立触摸通道，实际占用 IO 数量由用户配置决定。

## 5. 触摸库接口说明
触摸库提供汇编、C 语言两套接口，下文描述文件组成、函数定义、输入输出参数及调用方式。

### 5.1 汇编库接口

#### 5.1.1 文件组成

汇编触摸库包含 3 个核心文件：
1) TOUCH_SET.inc：用户触摸参数配置文件，配置阈值、功能、通道；
2) TOUCH.inc：触摸库 RAM 与底层参数定义，禁止用户修改；
3) xxx.lib：触摸库核心算法库。

编译顺序：先包含 TOUCH_SET.inc，再引入 TOUCH.inc。

#### 5.1.2 函数名与调用方式

核心函数：TOUCH
调用逻辑：主循环中循环调，完成信号采集、算法处理、触摸状态判定、按键状态输出等。
调用示例:

```
CALL TOUCH
```

#### 5.1.3 输入数据

- LP_EN(位变量): 低功耗模式控制位，1=进入低功耗扫描模式；
- TH_KEY_EXPIRE_L(字节变量)：按键超时标记。检测到通道长按键异常时，配置相应位触发通道复位；
- FIRST(位变量)： 触摸库复位控制，清零该位，可强制复位触摸库运行状态。

#### 5.1.4 输出数据

- LP_AL(只读位变量)：低功耗允许状态。 1=触摸函数允许系统进入低功耗扫描；
- KEY_STATE_L(只读字节变量)：触摸按键状态标志。对应 bit 为 1 表示该通道触摸有效。

### 5.2 C 库接口

#### 5.2.1 文件组成

C 触摸库共 5 个核心文件，文件需放置于工程 lib 目录：
1) TOUCH_SET.inc：用户可配置触摸参数定义；
2) TOUCH.inc：RAM 与底层参数定义，禁止用户修改；

3) touch.h：函数声明与宏定义头文件；
4) touch.c：C 语言变量定义文件；
5) xxx.alib：触摸核心算法库文件。

### 5.2.2 函数名与调用方式

函数原型: void touch(void);
函数声明位置：touch.h
调用逻辑：主循环中周期性调用，自动完成信号采集、动算与状态更新。
调用示例:

```
touch();
```

### 5.2.3 输入数据

SET_LP_EN(宏)：调用宏，触摸模块进入低功耗扫描模式；
TOUCH_RESET(宏)：调用宏，复位触摸模块全部运行状态；
key_expire(无符号整型)：按键超时寄存器，对应 bit 置 1 触发对应通道长按异常复位。

### 5.2.4 输出数据

key_flag(无符号整型)：触摸按键有效标志寄存器， 读取各通道触摸按键状态；
LP_AL_STA(宏)：获取触摸库低功耗许可状态，判断系统是否可进入低功耗。

## 6. 阈值设置

触摸阈值是触摸检测的核心判定参数，支持用户自定义配置：

| 参数名 | 默认值 | 说明 |
|---|---|---|
| OPT_TH_THRESHOLD_GP0_CHx | 0x1E78 | 高 8 位：噪声阈值;<br>低 8 位：触摸按键阈值, 推荐取值范围 30 ~ 255 |
| OPT_TH_LP_GP0_WK | 60 | 低功耗唤醒阈值，阈值超小，触摸唤醒灵敏度越高。 |

低功耗唤醒阈值推荐配置：

1. 单通道使能: 常规触摸阈值的 50%；
2. 2~3 通道使能: 最小触摸阈值的 5% ~ 10%；
3. 4 通道使能: 最小值建议取 7。

## 7. 二次触摸功能

二次触摸功能用于改善 PWM 输出、电源走线与触摸盘绕线干扰场景下的触摸延迟问题。开启后可提升复杂布线环境触摸响应速度与稳定性。

### 7.1 可配置参数

| 参数名 | 默认值 | 说明 |
|---|---|---|
| OPT_TOUCH_STATE_TOUCH_AGAIN | 0 | 二次触摸功能使能位，默认关闭 |
| OPT_TH_ESTA_RELEASE_DB | 8 | 二次触摸退出消抖次数 |

| 参数名 | 默认值 | 说明 |
| --- | --- | --- |
| OPT_TH_ESTA_BASE_UP_DB | 4 | 二次触摸基值更新消抖次数.有效取值 0x01-0x07 |

注意事项：开启二次触摸功能后，每个触摸通道额外占用 4 字节 RAM 资源。

## 8. 低功耗功能

触摸库默认开启低功耗功能，支持休眠状态下的低功耗触摸扫描与触摸唤醒，降低整机功耗。

### 8.1 可配置参数

| 参数名 | 默认值 | 说明 |
| --- | --- | --- |
| OPT_TH_SUPPOR_LP_ENABLE | 1 | 低功耗使能开关，默认开启 |
| OPT_TH_LP_GP0_WK | 60 | 低功耗唤醒阈值 |

### 8.2 低功耗配置流程

1. 应用读取 LP_AL 状态，确认触摸库允许进入低功耗(LP_AL = 1)；
2. LP_AL=1：启动低功耗计时；LP_AL=0：清空计时；
3. 计时超时后，配置看门狗时间，置位 LP_EN=1；
4. 触摸库检测到 LP_EN = 1，自动切换低功耗扫描模式；
5. 触摸唤醒后, 触摸库自动清零 LP_EN，退出低功耗模式.

## 9. 按键超时处理

按键超时用于防止触摸通道异常卡死、持续误判按键有效，保障长期稳定运行，建议默认开启。

处理流程：

1. 主程序检测按键有效时启动计时；按键释放清空计时；
2. 计时器超时且按键持续有效，置位 TH_KEY_EXPIRE_L 对应通道标志位;
3. 下一次调用触摸函数时，自动复位该异常通道状态。

注意：

- ✓ TH_KEY_EXPIRE_L 的有效赋值必须和通道选择配置 OPT_TOUCH_CHANNEL_SEL_L 匹配 例: OPT_TOUCH_CHANNEL_SEL_L=0x06, 则 TH_KEY_EXPIRE_L 有效赋值为 0x02,0x04,0x06。
- ✓ 不使用该功能时，禁止操作 TH_KEY_EXPIRE_L 寄存器。

## 10. 调试输出功能

触摸库支持串口调试输出，可打印触摸通道原采样数据、阈值、平均值等信息，必须搭配 PT Debug Tool 解析, 普通串口工具无法解析该私有协议。

### 10.1 可配置参数

| 参数名 | 默认值 | 说明 |
| --- | --- | --- |
| OPT_DBG_UART_TX_PIN | P02 | 调试串口(TX)输出引脚，需触摸库外部配置为输出脚 |
| OPT_DBG_AVERAGE_DATA_OUT | 0 | 触摸通道平均值输出使能，默认关闭 |

| 参数名 | 默认值 | 说明 |
|---|---|---|
| OPT_DBG_BASE_DATA_OUT | 0 | 触摸通道基值输出使能，默认关闭 |
| OPT_DBG_SCAN_DATA_OUT | 0 | 触摸通道扫描值输出使能，默认关闭 |
| OPT_DBG_THRESHOLD_OUT | 0 | 触摸通道阈值输出使能，默认关闭 |
| OPT_DBG_UART_BAUDRATE_DLY | 31 | 串口调试输出波特率配置参数 |

### 10.2 调试串口输出波特率配置对照表

OPT_DBG_UART_BAUDRATE_DLY 需要匹配系统主频：

| CPU 主频 | 9600 | 19200 | 38400 | 76800 | 115200 |
|---|---|---|---|---|---|
| 8M | --- | 134 | 65 | 30 | 19 |
| 4M | 134 | 65 | 31 | 13 | --- |
| 2M | 64 | 30 | 13 | 4 | --- |
| 1M | 30 | 13 | 4 | --- | --- |

注：CPUCLK 参考规格书 Clock 与 Option 章节描述

## 11. 其它可配置参数

包含通道选择、消抖次数、EMI/EMC 优化、时钟配置、扫描配置等参数:

| 参数名 | 默认值 | 说明 |
|---|---|---|
| OPT_TOUCH_CHANNEL_SEL_L | 1111B | 触摸通道使能配置，每一比特对应一个通道，配置范围 0000B ~ 1111B。 |
| OPT_KEY_IN_DEBONCE | 0x02 | 按键有效消抖次数 |
| OPT_KEY_OUT_DEBONCE | 0x02 | 按键释放消抖次数 |
| OPT_TOUCH_EMI_RE_OPTIMIZE | 0 | RE 电磁干扰优化开关 |
| OPT_TOUCH_EMS_CS_EFT_OPTIMIZE | 0 | CS/EFT 电磁兼容优化开关 |
| OPT_TOUCH_CDCCK_SEL | 000B | CDC 扫描时钟选择 |
| OPT_TOUCH_CDCTMS_TIME | 120 | CDC 充电时间配置，单位：μs |
| OPT_TOUCH_CDCLDO_EN | 1 | CDC LDO 使能开关 |
| OPT_TH_CDCLDO_2V_EN | 0 | CDC LDO 电压选择: 0 = 2.3V; 1 = 2V |
| OPT_TH_FREQ_CHANGE_EN | 0 | 触摸扫描变频功能使能开关 |
| OPT_ONE_CYCLE_SCAN_TIME | 3 | 扫描次数: 0=1 次扫描, 1=2 次扫描.....; |

## 12. 开发注意事项

### 12.1 中断处理

1, 中断处理函数必须手动清除中断标志位 THIF，否则会引发中断异常；

2, 中断内使用间接寻址操作时，进入中断先备份相关寄存器，退出中断前还原寄存器

### 12.2 WDT 说明

触摸库内部自带看门狗清零指令(CLRWDT), 用户程序无需重复喂狗。

## 13. 应用开发流程示例

- 基础配置：配置触摸通道、触摸阈值、消抖次数、低功耗相关参数；
- 主循环调用：主循环周期性调用触摸库核心函数；
- 低功耗逻辑：读取低功耗许可状态，满足条件则进入休眠；唤醒后恢复正常扫描；
- 异常处理：检测超长按键超时，触摸异常时执行通道复位；
- 状态输出：读取按键标志位，执行相应的程序逻辑。

详细工程实现请参考官方配置 Demo 程序。

## 14. 版本历史记录

| 版本号 | 修改记录 | 发布日期 |
|---|---|---|
| V1.0 | 初始版本，完成全功能手册编写 | 2026-07-10 |
| V1.1 | 1, 修改 ROM/RAM 资源占用描述错误<br>2, 修改描述文字 | 2026-09-15 |
| V1.2 | 1, 修改 RAM 资源占用章节 | 2026-11-9 |
| V1.2 | 1, 修改 RAM 资源计算公式 | 2026-11-9 |