

2011.07

SC05A

5按键带自校正功能的容性触摸感应器

1. 概览

1.1 概述

SC05A 是带自校正的容性触摸感应器，可以检测 5 个感应盘是否被触摸。它可以通过任何非导电介质（如玻璃和塑料）来感应电容变化。这种电容感应的开关可以应用在很多电子产品上，提高产品的附加值。

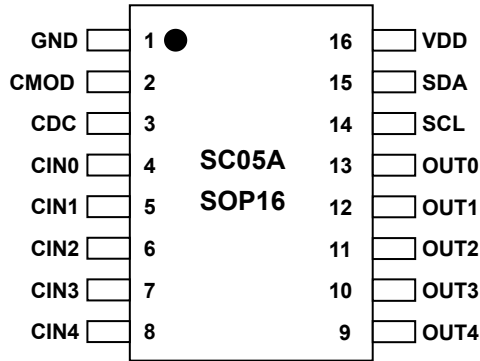
1.2 特征

- ◇ 5 个完全独立的触摸感应按键
- ◇ 保持自动校正，无需外部干预
- ◇ 按键输出经过完全消抖处理
- ◇ 多接口 – I²C 串行接口 / 并行一对一输出
- ◇ 所有按键共用一个灵敏度电容
- ◇ 感应线长度不同不会导致灵敏度不同
- ◇ 2.5V ~ 6.0V 工作电压
- ◇ 符合 RoHS 指令的环保 SOP16 封装

1.3 应用

- ◇ 替代机械开关
- ◇ 家庭应用(电视机, 显示器键盘)
- ◇ 玩具和互动游戏的人机接口
- ◇ 门禁按键
- ◇ 灯控开关
- ◇ 密封键盘面板

1.4 封装



封装简图

1.5 管脚列表

管脚编号	名称	类型	功能	不使用时
1	GND	Pwr	电源地	-
2	CMOD	I/O	接电荷收集电容	-
3	CDC	I/O	接灵敏度电容	-
4	CIN0	I/O	感应按键0检测输入	悬空
5	CIN1	I/O	感应按键1检测输入	悬空
6	CIN2	I/O	感应按键2检测输入	悬空
7	CIN3	I/O	感应按键3检测输入	悬空
8	CIN4	I/O	感应按键4检测输入	悬空
9	OUT4	OD	感应按键4输出	悬空
10	OUT3	OD	感应按键3输出	悬空
11	OUT2	OD	感应按键2输出	悬空
12	OUT1	OD	感应按键1输出	悬空
13	OUT0	OD	感应按键0输出	悬空
14	SCL	I	I ² C 时钟输入	连接GND或VDD
15	SDA	I/O	I ² C 数据输入输出	连接GND或VDD 或者悬空
16	VDD	Pwr	电源正极	-

管脚类型

I	CMOS 输入
I/O	CMOS 输入/输出
OD	NMOS 开漏输出
Pwr	电源 / 地

1.6 管脚说明

VDD, GND

电源正负输入端。

CMOD

电荷收集电容输入端，接固定值的电容，和灵敏度无关。

CDC

接灵敏度电容，电容范围是最小15pf，最大100pf。根据使用环境选择合适的电容值，数值越小，灵敏度越高。

CIN0~CIN4

接感应盘，是感应电容的输入检测端口。

OUT0~OUT4

并行一对一输出端口，分别对应CIN0~CIN4。端口内部结构为NMOS开漏输出，输出高阻或低电平，有效电平是低电平。

SCL, SDA

SCL 是I²C时钟输入端口。SDA是I²C数据输入输出端口。 SDA 端口有内部弱上拉。

2. 芯片功能

2.1 初始化时间

上电复位后，芯片需要300ms进行初始化，计算感应管脚的环境电容，然后才能正常工作。

2.2 灵敏度

灵敏度由CDC端口接的电容值决定。数值越小，灵敏度越高。

2.3 自校正

根据外部环境温度和湿度等的漂移，芯片会一直调整每个按键的电容基准参考值。从检测到按键开始，芯片会停止校正一段时间，这段时间大约15~50秒。然后芯片会继续自校正，也就是说检测按键有效的时间不会超过15~50秒。

2.4 触摸反应时间

每个通道大约每隔12.5ms采样一次。经过按键消抖处理以后，检测到按键按下的反应时间大概是68毫秒，检测按键离开的反应时间大概是44毫秒。所以检测按键的最快频率大概是每秒9次。

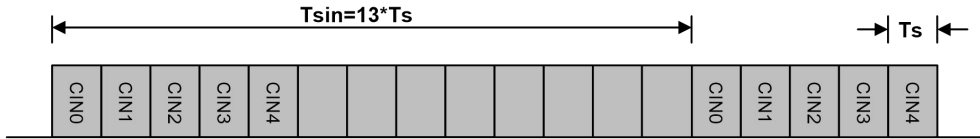
2.5 输出逻辑

输出端口OUT0~OUT4，有两种状态，高阻或低电平。无按键时为高阻，检测到按键时为低电平。

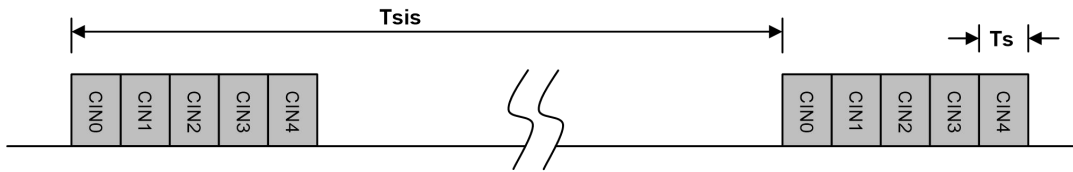
2.6 睡眠模式

如果在一段时间内（Tslp）没有检测到按键并且SDA端口一直保持高电平，芯片会自动进入省电模式。只要让SDA保持高电平时间不超过Tslp，芯片就不会进入睡眠模式。在睡眠模式中，按键的采样间隔会变长，电

流消耗 (I_{dd}) 会减小。如果检测到按键，芯片会马上离开睡眠模式，进入正常模式。



正常模式下采样周期图示



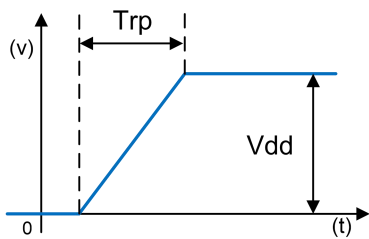
睡眠模式下采样周期图示

- Ts : 单个按键采样周期
- Tsin : 正常模式采样间隔
- Tsis : 睡眠模式采样间隔

Ts 大约是固定的950us左右。

正常模式下，采样间隔Tsin 是固定的大约12.5毫秒。

睡眠模式下，采样间隔Tsis和电流消耗I_{dd}是与V_{dd}和Trp (电源上电时间)有关的。详见下表：



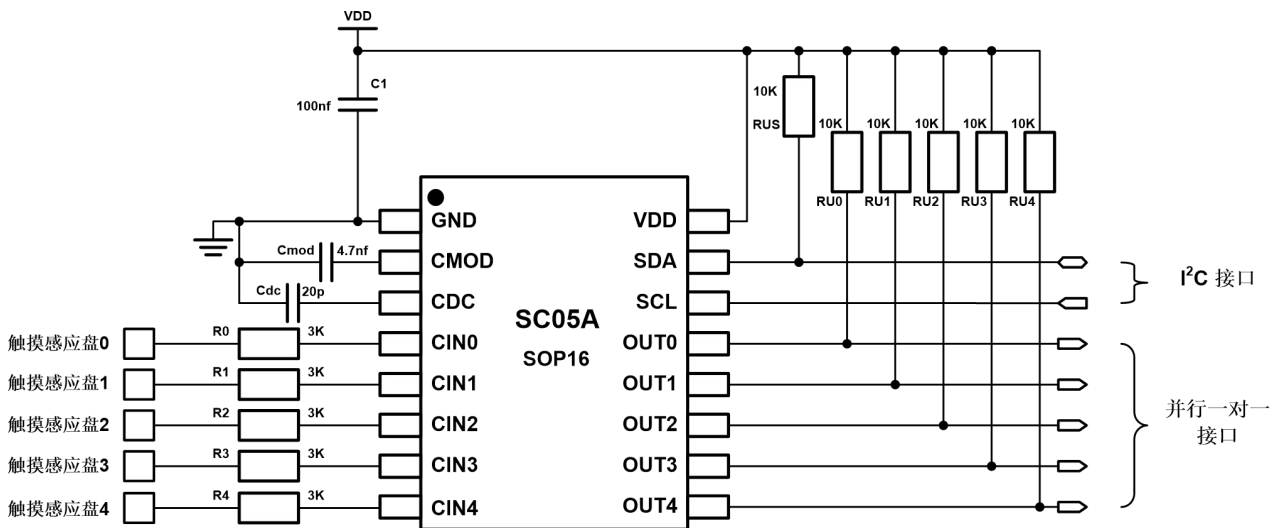
VDD 上电曲线

条件*	Vdd=5v					Vdd=3v				
	Trp < 10us	Trp = 100us	Trp = 1ms	Trp = 10ms	Trp > 100ms	Trp < 10us	Trp = 100us	Trp = 1ms	Trp = 10ms	Trp > 100ms
典型值										
Tsis (ms)	270	252	210	92	67	270	260	245	160	135
I _{dd} (ua)	81	86	104	238	326	39	40	43	65	77
Tslp (s)	86.4	80.6	67.2	29.4	21.4	86.4	83.2	78.4	51.2	43.2

*表中数据是在睡眠模式下测得

3. 应用

3.1 应用电路

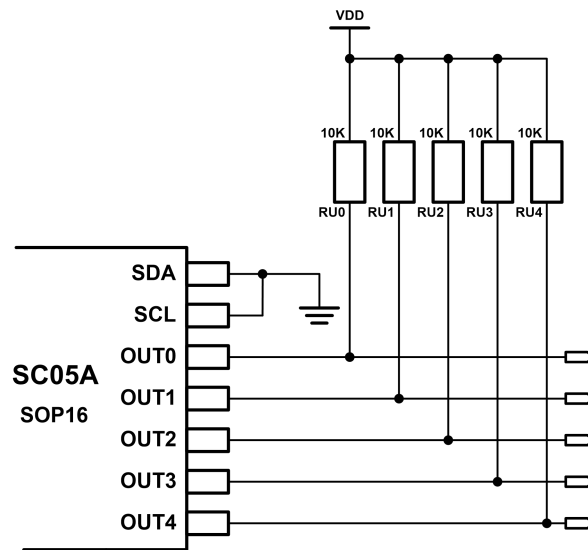


注意:

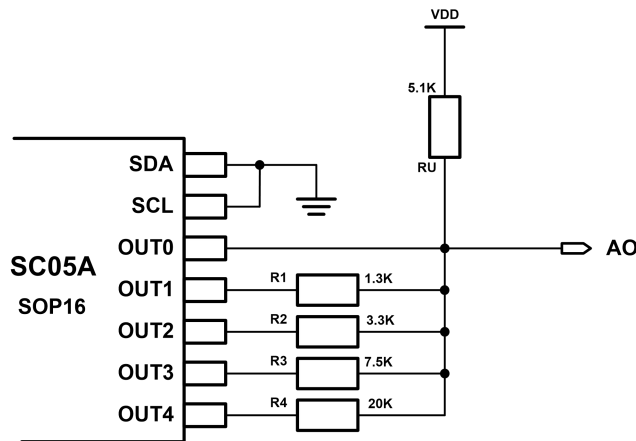
1. Cmod是电荷收集电容，取值范围是1nf~10nf。建议使用4.7nf。
2. Cdc 是灵敏度设置电容，取值范围是最小15pf，最大100pf，电容值越小灵敏度越高。
3. 并行一对一输出和I²C接口同时有效。

3.2 和单片机控制器的接口方式

3.2.1 并行一对一输出

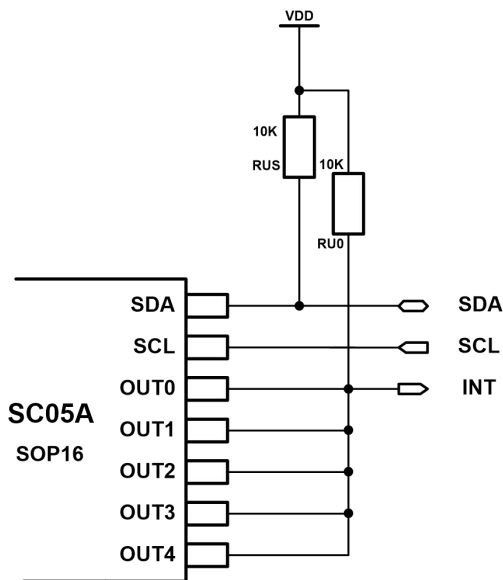


3.2.2 模拟电压输出

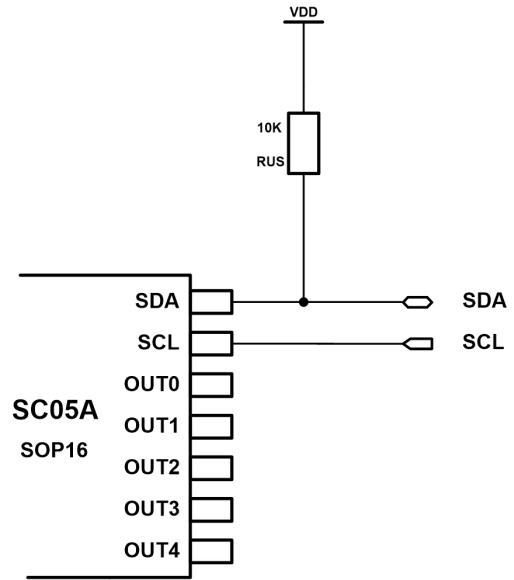


按键情况	OUT0	OUT1	OUT2	OUT3	OUT4	AO 输出电压
触摸感应盘 0(Cin0)	低电平	高阻	高阻	高阻	高阻	0v
触摸感应盘 1(Cin1)	高阻	低电平	高阻	高阻	高阻	0.20*VDD
触摸感应盘 2(Cin2)	高阻	高阻	低电平	高阻	高阻	0.40*VDD
触摸感应盘 3(Cin3)	高阻	高阻	高阻	低电平	高阻	0.61*VDD
触摸感应盘 4(Cin4)	高阻	高阻	高阻	高阻	低电平	0.80*VDD
无触摸感应	高阻	高阻	高阻	高阻	高阻	VDD

3.2.3 I²C 接口



I²C 中断方式



I²C 查询方式

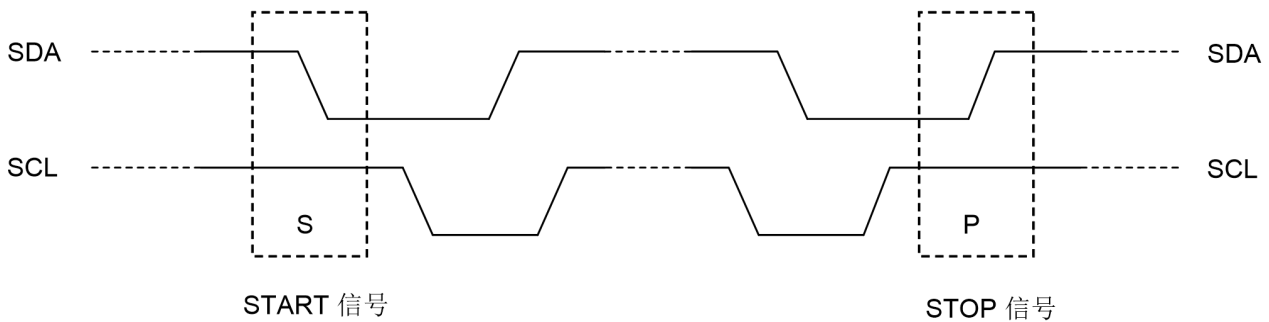
1. Start 和 Stop 信号

Start 信号(S)

当 SCL 是高电平时，SDA 由高到低变化，表示开始传输数据。

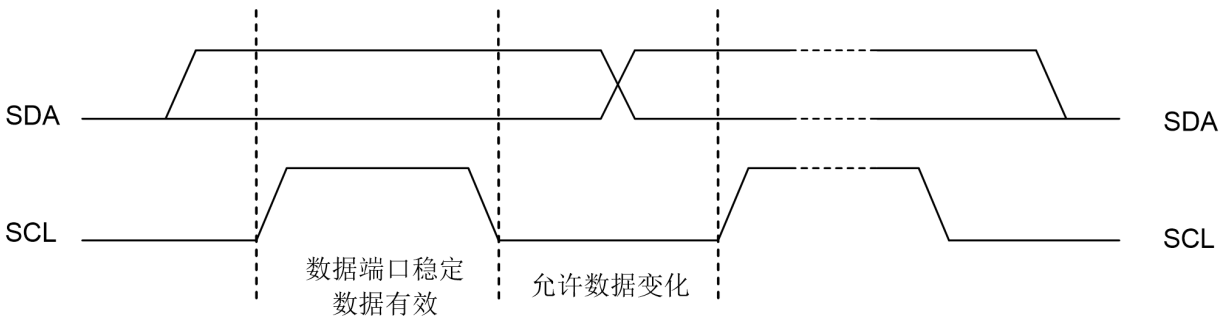
Stop 信号(P)

当 SCL 是高电平时，SDA 由低到高变化，表示结束数据传输。



2. 数据有效

在 SCL 为高电平期间，SDA 必须保持稳定的电平。SDA 线上的高低电平变化只能在 SCL 为低电平期间。

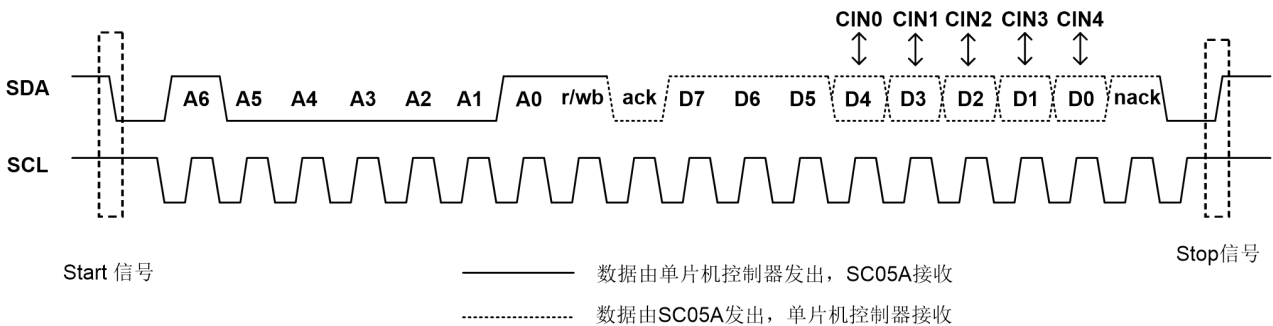


3. 字节格式

字节由 8 位数据和一个应答信号组成

4. SC05A 使用简化的 I2C 协议

- 1) 标准 I²C 器件有器件地址和寄存器地址。SC05A 只有器件地址。
- 2) SC05A 只接收读命令。
- 3) SC05A 的器件地址是 41H(A[6:0]=1000001B)。下图是一次完整的通信用途。D7~D5 是固定高电平，D4~D0 分别对应 CIN0~CIN4 是否有按键触摸。例如，按键 CIN0 被触摸，D4 将是低电平，如果 CIN0 没有被触摸，D4 将是高电平。



4. 详细参数

4.1 额定值 *

工作温度	-40 ~ +85°C
存储温度.....	-50 ~ +150°C
最大Vdd电压.....	-0.3 ~ +6.0V
管脚最大直流输出电流.....	±10mA
管脚容限电压.....	-0.3V ~ (Vdd + 0.3) Volts

* 注意：超出上述值可能导致芯片永久损坏

4.2 电气特性

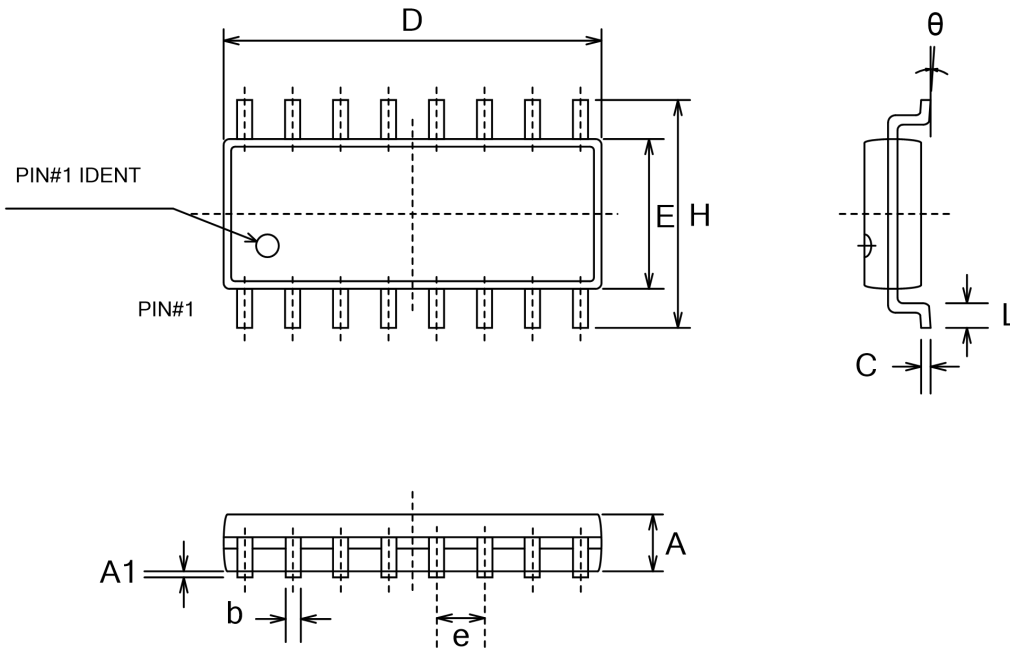
T_A = 25°C

参数	符号	测试条件	最小值	典型值	最大值	单位
工作电压	Vdd		2.5		6.5	V
工作电流 ¹	Idd	VDD=5.0V		1.74		mA
		VDD=3.0V		0.84		mA
芯片上电初始化时间	Tini			300		ms
感应管脚电容范围	Cin				2.5*Cdc ²	
灵敏度电容	Cdc		15pf		100pf	
输出阻抗 (NMOS开漏)	Zo	delta Cin > 0.2pF		50		Ohm
		delta Cin < 0.2pF		100M		
输出灌电流	I _{sk}	VDD=5V			10.0	mA
最小可检测电容	delta_Cin	CDC=15pf		0.2		pF
I ² C 最大波特率	F _{br}	PullUp Res = 10K		400K		Bit/S
采样间隔时间	Tsin	Normal mode		12.5		ms

¹ 正常工作模式下

² 如果感应管脚寄生电容超过2.5倍的Cdc电容，芯片不能正常工作（绝大多数情况无需考虑这个限制）

4.3 封装尺寸图 (SOP-16)



Symbol	Dimensions In Millimeters			Dimensions In Inches		
	Min	Nom	Max	Min	Nom	Max
A	1.30	1.50	1.70	0.051	0.059	0.067
A1	0.06	0.16	0.26	0.002	0.006	0.010
b	0.30	0.40	0.55	0.012	0.016	0.022
C	0.15	0.25	0.35	0.006	0.010	0.014
D	9.70	10.00	10.30	0.382	0.394	0.406
E	3.75	3.95	4.15	.0148	0.156	0.163
e	--	1.27	--	--	0.050	--
H	5.70	6.00	6.30	0.224	0.236	0.248
L	0.45	0.65	0.85	0.018	0.026	0.033
θ	0°	--	8°	0°	--	8°

附录:

通过I²C接口读取SC05A的C语言演示程序

```

// 使用 AT89S52 单片机
#define SDA          P1_5
#define SCL          P1_4
#define ERR          P1_3    // 指示通信过程有问题
#define CON_ADDR    0x83    // {A[6:0] + RWB} = 83H

unsigned char ReadKey(void)
{
    unsigned char  bitnum,temp,addr,key;
    bit            bit_temp;
    addr=CON_ADDR;
    key=0xff;

    EA=0;                // 屏蔽全局中断
    SDA=0;               // 拉低 SDA 端口送出 START 信号
    for(temp=0;temp<4;temp++){ //延时

                                //发送 8 位地址字节 (A[6:0]+RWB)
    for(bitnum=0;bitnum<8;bitnum++)
    {
        SCL=0;
        temp=addr&0x80;
        if(temp==0x80)
            SDA=1;
        else
            SDA=0;
        addr=addr<<1;
        for(temp=0;temp<4;temp++){ //延时
            SCL=1;
            for(temp=0;temp<4;temp++){ //延时
            }
        SDA=1;                //释放 SDA 端口,将 SDA 设置为输入端口

        SCL=0;
        for(temp=0;temp<4;temp++){ //延时
        SCL=1;
        for(temp=0;temp<4;temp++){ //延时
        bit_temp=SDA;
        if(bit_temp)                //读 ack 回应
            ERR=0;                //ack 信号没有读到, 指示通信有误

                                //读 8 为按键数据字节(D[7:0])
        for(bitnum=0;bitnum<8;bitnum++)
        {
            SCL=0;
            for(temp=0;temp<4;temp++){ //延时
            SCL=1;
            for(temp=0;temp<4;temp++){ //延时
            bit_temp=SDA;
    
```

```
    if(bit_temp)
    {
        key=key<<1;
        key=key|0x01;
    }
    else
    {
        key=key<<1;
    }
}
SCL=0;
SDA=1;
for(temp=0;temp<4;temp++){ //延时
SCL=1;
for(temp=0;temp<4;temp++){ //延时
SCL=0;
SDA=0; //发送 NACK 信号
for(temp=0;temp<4;temp++){ //延时
SCL=1;
for(temp=0;temp<4;temp++){ //延时
SDA=1; //释放 SDA 端口, 将 SDA 设置为输入端口
key=key^0xff;
EA=1; //全局中断使能
return(key); //数据位为 1, 说明相应按键被触摸。例如, 返回值为 0x05 说明按键
//CIN2 和 CIN4 被触摸。
}
```