

配置多个电容触摸传感按钮

作者: Keith Curtis
Microchip Technology Inc.
Tom Perme
Microchip Technology Inc.

介绍

本应用笔记将说明在使用超过 4 个电容触摸传感按钮时，如何扫描和检测按钮是否按下。本应用笔记针对 PIC16F616 系列、PIC16F690 系列以及 PIC16F887 系列单片机。建议在阅读本应用笔记之前先了解电容触摸传感的基本概念，这在 AN1101 《电容触摸传感简介》中进行了介绍。

本应用笔记将讨论使用 Microchip 单片机来实现多个触摸按钮的三种方法。第一种方法是使用与比较器模块输入相连的片上 4 选 1 模拟多路复用器，创建简单的 4 传感器系统。第二种方法是把第一种途径创建的 4 传感器系统扩展成 10 传感器系统，扩展方法是把原先的 4

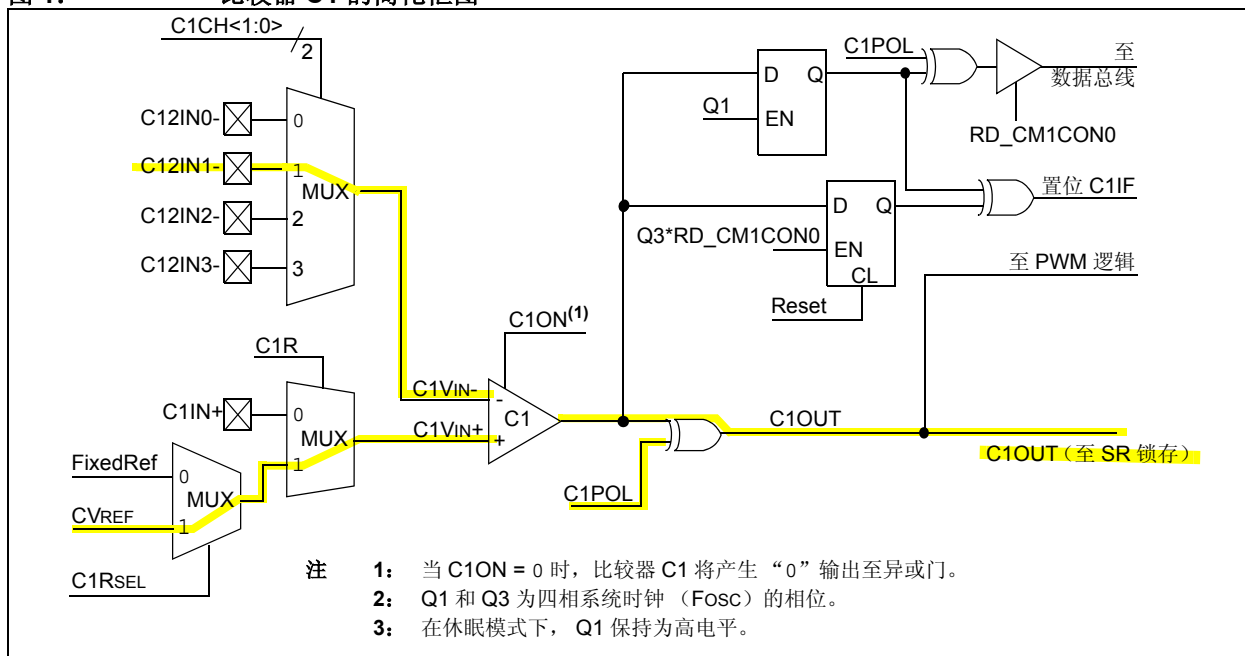
个输入配成对。第三种方法则是创建可扩展的系统，其原理是使用外部模拟多路复用器对添加的传感器进行多路复用。

使用默认数量的按钮

默认情况下，比较器模块能够进行电容触摸传感功能 * 的 PIC® 单片机，可以比较器输入使用内部多路复用器，最多扫描四个按钮。内部 MUX 通过通道选择位来进行控制，即 CM1CON0 的 C1CH<1:0> 和 CM2CON0 的 C2CH<1:0>。两个比较器的通道设置必须相同。

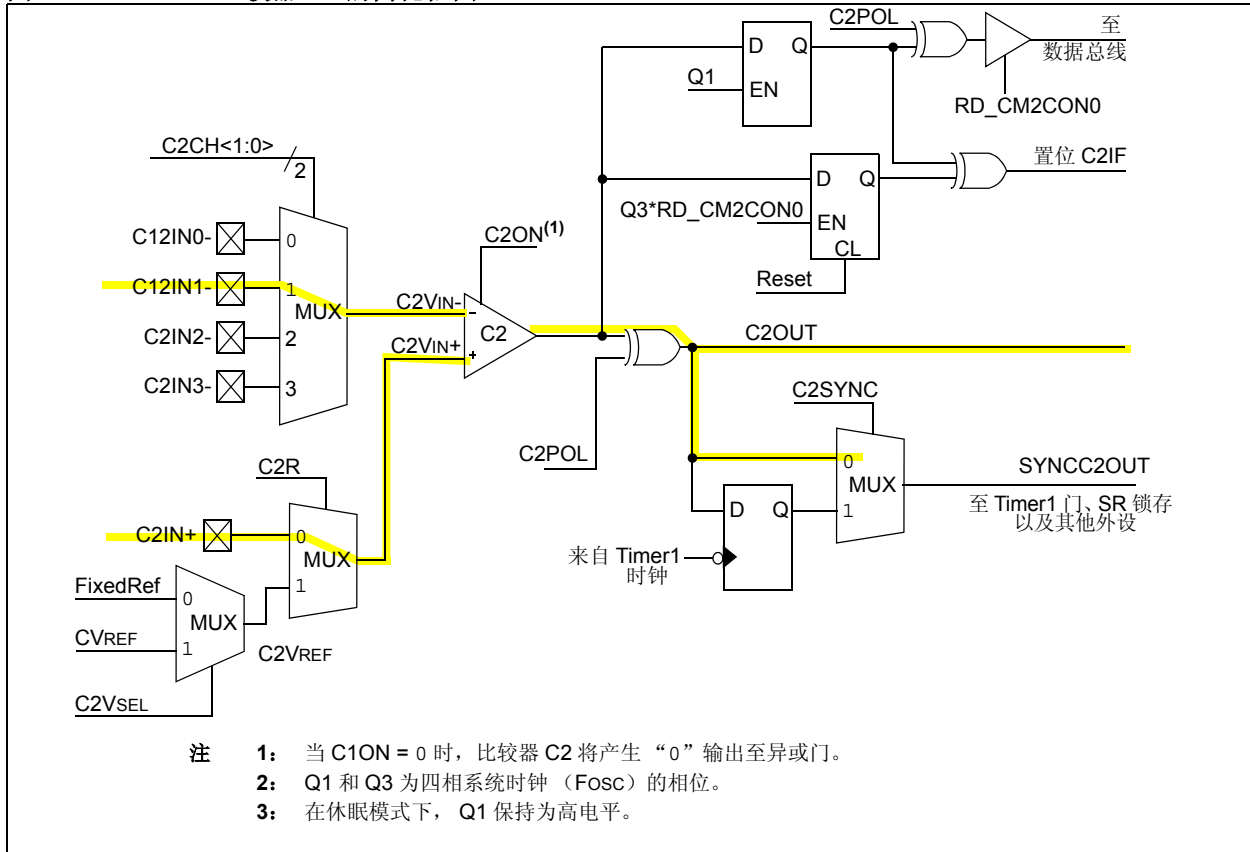
摘自 PIC16F887 系列数据手册的简化框图如图 1 所示。图中，高亮显示部分是电容触摸传感所需的正确通路。在每个内部多路复用器上，选择的通道必须相同。从而当一个按钮切换到下一个时，确保通路相同。选择的通道必须相同是因为基本的传感器振荡电路需要把电容上的电压与上、下限值 C1IN+、C2IN+ 进行比较。如果负输入不同的话，电路将不会振荡，输出就会保持在高或低电平上。

图 1: 比较器 C1 的简化框图



* 上述具有触摸传感功能的比较器模块必须有 SR 锁存选项。

图 2: 比较器 C2 的简化框图



在软件中使用 C 代码处理对四个按钮的扫描是非常有效和方便的。最方便的做法是用数组来处理按钮的测量原始值和平均值，数组的每一个元素对应一个按钮的值。每个按钮最好有各自的 trip 阈值，以便处理各个按钮的不同情形。在平常的 C 程序中，这些变量应定义如下：

```
unsigned int    average [4];
unsigned int    trip [4];
```

注： 为了便于讨论，本文假定使用的是简单的 trip 阈值。应用笔记 AN1103 《电容触摸传感的软件处理》中，介绍了更多有关各种检测按钮按压方法的详细内容。求平均值对于所有的方法而言都是基本的方法。

数组变量 **average** 存放的是每个按钮的平均值，下标从 0 到 3。同样也可以用一个数组来存放最新读数，用于查看或者是辅助设计，但是原始数据并不需要保存，因为它是在每次扫描结束时测量得到的，并由中断服务程序 (Interrupt Service Routine, ISR) 作出按钮是否按下的判断。

在深入讨论细节之前，先了解概况：基本运算是在中断服务程序中完成的。一旦某个按钮准备好进行测量并完成扫描，若产生 Timer0 溢出中断，则调用 ISR。

其他中断也可能会调用 ISR，但这将需要用户进行适当处理，其他中断将会与固定时基 Timer0 中断同时存在。大致的高阶 ISR 构成如下：

例 1: 简单的 ISR

```
void isr {
    // 如果是电容触摸中断
    if (T0IF == 1) {

        // 获取 TMR1 读数
        StopTimers();
        GetMeasurement();

        // 测试是否按下
        if (IsButtonPressed(index))
            SetFlag(index);
        else
            ClearFlag(index);

        // 进行 16 点平均值计算
        PerformAverage(index);

        // 设置下一个传感器
        index = (++index) & 0x03;
        SetComparators(index);
        RestartTimers();
    }
}
```

多个按钮的处理是通过编写注释“设置下一个传感器”下面部分的代码来实现的。要在四个按钮之间切换，需要一个下标变量来记录正在扫描的是哪个按钮，并根据该变量来进行正确的初始设置。使用四个按钮时，下标从 0 到 3，就像定义 `average` 和 `trip` 数组那样。下标变量应定义如下：

```
unsigned char    index;
```

在 `ISR` 最后，递增扫描的 `index` 变量，为下一次扫描做准备。在 `index` 变量递增之后，必须设置比较器通道选择位 `C1CH<1:0>`、`C2CH<1:0>`，并必须重启 `Timer0` 和 `Timer1`。这可以用许多方法实现，但方便的做法是创建一个存放 4 个常量的数组，常量是整个寄存器 `CM1CON0` 和 `CM2CON0` 的设置值，然后使用 `index` 来读取这些值，并把它们载入寄存器中。

例如，假定常量数组定义为 `COMP1` 和 `COMP2`。数组中存放的是寄存器的设置值，然后改变每个通道位，即位 0 和位 1。

```
COMP1 = {0x94, 0x95, 0x96, 0x97}
COMP2 = {0xA0, 0xA1, 0xA2, 0xA3}
```

在 `index` 改变之后（`ISR` 最后）：

例 2： 设置比较器

```
void SetComparators(char index) {
    CM1CON0 = COMP1[index];
    CM2CON0 = COMP2[index];
}
```

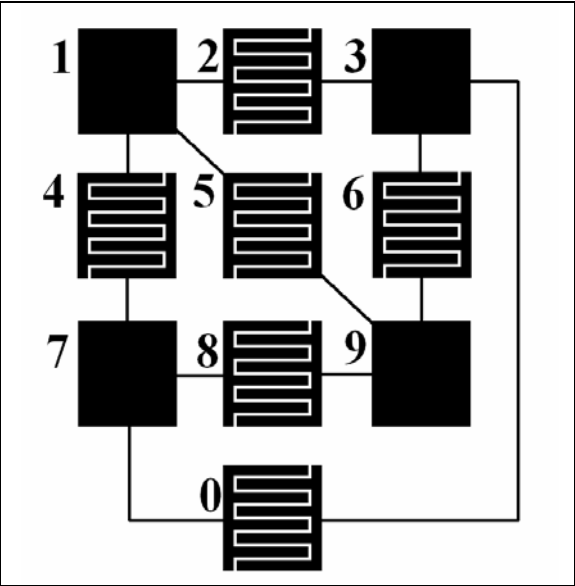
每一次按钮切换，`ISR` 都将进行扫描，递增 `index`，然后为要扫描的下一个按钮做好准备，即设置 `CMxCON0` 寄存器并重启定时器。`index` 递增不得超过 3，然后必须从 3 回到 0，不过这是软件细节了，容易处理。一种计数到 3 然后回 0 的方法是把结果与 3 进行“与（AND）”运算，这将最高 6 位清零，如上面例 1 所示。变量 `index` 递增到 4（0b100），而后与 3（0b011）进行“与”运算，得到结果 0。

现在将按顺序 0, 1, 2, 3, 0, 1, 2, 3 ... 不停地扫描四个按钮，剩下需要完成的工作是 `SetFlag(int index)`、`ClearFlag(int index)` 以及 `PerformAverage(int index)` 函数。设置和清除标志位对某些应用可能就已足够，而有些应用可能要直接采取其他的操作。按钮的操作以 `index` 值为准，因为 `index` 指明了按下的是哪个按钮。16 点平均值也要根据 `index` 来进行。每次按钮切换都应重新计算平均值，平均值应存放在 `average` 数组中 `index` 指向的相应位置。

通过配对按压来进行扩展

使用比较器模块作为触摸按钮接口的主要缺点之一是输入个数受到限制。要支持更多的传感器，一种方法是把现有触摸输入进行配对，构成新的触摸传感器（见图 3）。当用户触压配对传感器时，配对的两个传感器的输入均受到相同作用，根据减少的偏移以及施加在两个输入（而不是一个输入）上的作用，软件将配对触摸与来自单个输入的触摸区别开来。

图 3： 仅有 4 个输入时的多个按钮



由于配对传感器输入结合了已有输入，所以不需要额外的电路，系统计算平均值的存储器开销也不会增加。唯一需要增加的是解码逻辑，用于搜寻单个和配对的按钮按压状态。

表 1 列出了传感器输入个数确定时，能够得到的触摸传感器数量。由于比较器输入有四通道的内部 MUX，最大传感器数量是十个。

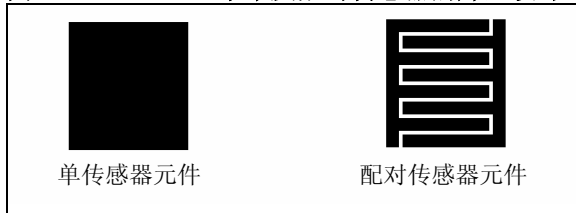
表 1： 最多传感器数量

输入数	传感器数量
2	3
3	6
4	10
5	15
6	21
7	28
N	0.5 (N ² + N)

使用配对传感器系统依然存在局限。一次只能有一个传感器被按下，与完整传感器相比较，配对传感器只能使传感器电路的频率产生一半的偏移量。这就要求在解码程序中有一些附加逻辑，并限制了一些传感器输入的灵敏度。在设计系统时，设计人员应该把这些因素都考虑进去，在灵敏度要求较高的应用中使用完整的传感器，而在能够接受较低灵敏度的应用中使用配对传感器。

配对传感器的传感器模式要求两个传感器的输入相互交错（见图 4）。

图 4： 单个及配对传感器的焊盘设计



需要传感器相互交错是为了在传感器上手触摸位置不确定的情况下，尽可能使两个输入的偏移保持相等。如果可能的话，两个传感器元件的面积应该相等，约为单个传感器面积的 $\frac{1}{2}$ 。尽管这样增加了传感器的大小，但配对传感器的灵敏度将更高。注意，两个配对传感器元件之间的间距也应该尽可能大，这样在配对传感器两边连接的单个传感器被激活时，可以避免在元件之间产生交互。

共用按钮系统的解码逻辑从测试每个频率值开始，把频率值与两个触摸阈值进行比较，一个阈值用于单个传感器的处理，较小的阈值则用于配对传感器。测试结果交给搜索算法，搜索算法先检查配对偏移，然后检查单个偏移。如果发现了配对偏移，就认为按钮已经检测到，不再进行单个偏移测试。如果没有检测到配对偏移，就进行单个偏移测试，并报告任何检测到的按下状态。如果检测到超过配对阈值的偏移超过了两个，就判定出现 Fault（故障）状态，终止解码程序。

通过多路复用器进行扩展

增加按钮数量的另一种方法是使用一个或几个外部模拟多路复用器。当要处理的按钮非常多时，这是个非常有效的办法；它与“配对按压”方法不同，它无需使用任何特殊的技巧。不过，随之而来的代价是更大的 PCB 表面积，而且每个 MUX 都引入电容，而这将降低灵敏度。由于增加了按钮，整个系统的扫描速度也将降低。所以

建议最开始在每个比较器输入通道上只使用一个 MUX，然后每添加一个 MUX 都要编译并测试系统，渐进地进行。由于寄生电容的增加，连接的 MUX 的数量是有限的，因为最终过高的寄生电容将使手指按压引起的电容改变无法被检测到。

处理外部 MUX 与处理内部 MUX 很相似，要选择把哪个比较器输入通道指派给哪两个比较器。现在选择 MUX 线的额外选择通道对于器件而言是外设，所以必须使用 I/O 引脚来选择 MUX 连接的电容焊盘通道，在内部必须设置比较器输入通道选择位，指向相应通道，从而确保来自 MUX 共用线的连接将被连接到正确的比较器输入上。

图 5 的原理图，说明了如何把外部 MUX 连接到 PIC 单片机。基本原理与 AN1101 《电容触摸传感简介》中相同，不过，在连接到比较器输入之前，焊盘先要通过 MUX 连接到共用线。

注： 推荐的 8 通道模拟 MUX 是 74HC4051，推荐的 16 通道 MUX 是 74HC4067。

现在，至关重要的是要注意在 MUX 和接通的按钮上，按钮的 index 是如何分配的以及 MUX 的共用线连接的是哪个比较器的输入。index 就是前面描述的变量，用来识别要扫描的按钮以及在按钮之间进行切换。一种用来设置按钮 index 值的合理方法如下。假定需要 32 个按钮，这需要使用 4 个 8 通道 MUX。在 C12IN0- 上，放置 MUX0；这个多路复用器将用于按钮 0 到 7。在 C12IN1- 上，放置 MUX1，用于按钮 8 至 15。在 C12IN2- 上，放置 MUX2，用于按钮 16-23，类似地在 C12IN3- 上放置 MUX3，用于按钮 24-31。

于是，用来处理这样设置的软件就变得相当简单。如果 index 是 0-7，使能 MUX0，设置通道选择位，使得 index 值 $\in [0, 7]$ 。如果 index 大于 7，index 被 8 除之后的模数就是通道选择位，index 被 8 整除的商，对应的就是要使能的 MUX。例如，假设 index = 21：

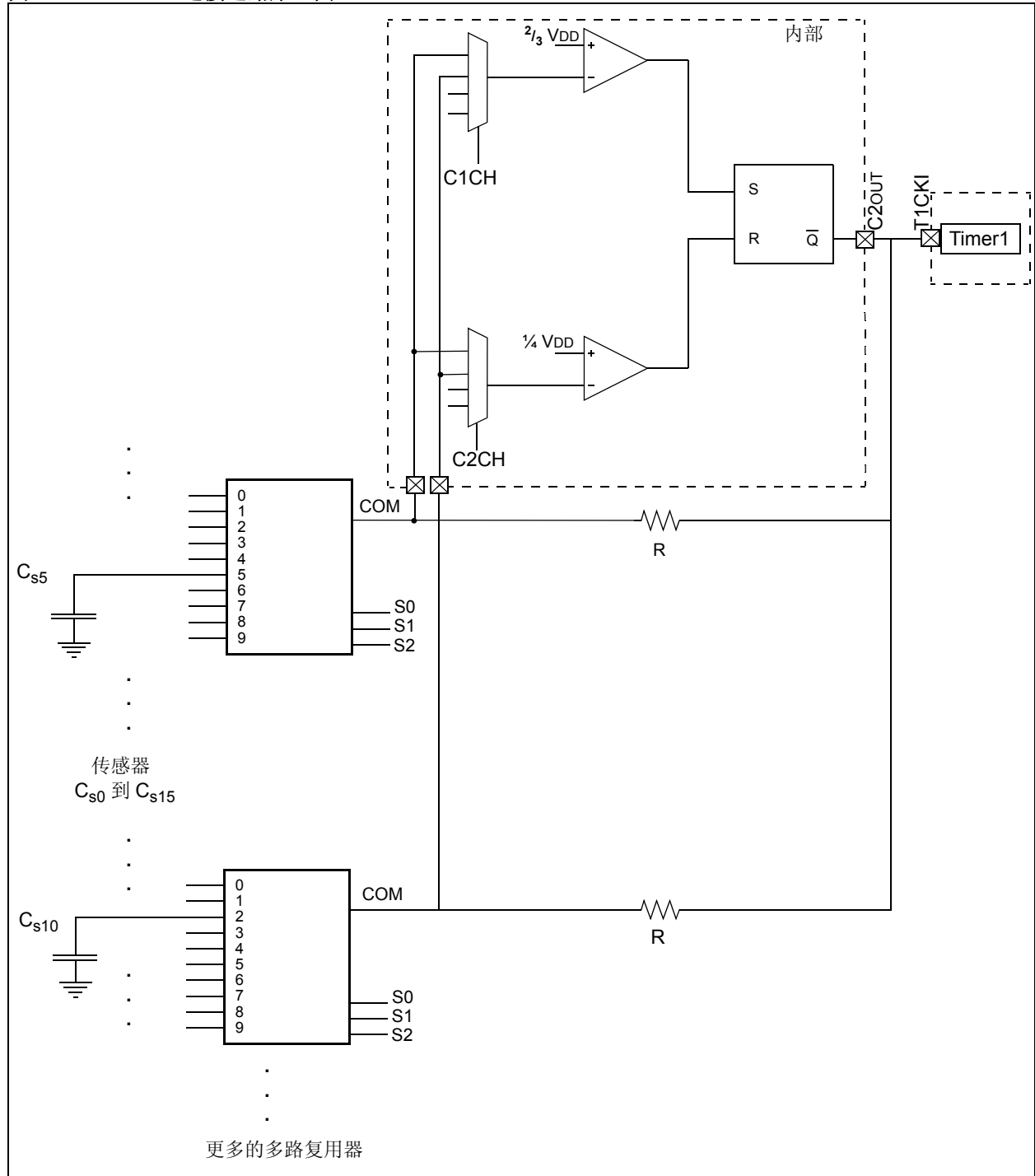
$$\begin{aligned} \text{index} \% 8 &= 21 \% 8 = 5 \\ \text{index} / 8 &= 21 / 8 = 2 \end{aligned}$$

$$\text{MUX 通道选择位} = 5 = (101)_2$$

$$\text{要使能的 MUX} = 2 = \text{MUX2}$$

要验证结果是否正确，就要回顾一下按钮 index 是如何分配的。在 C12IN2- 上，MUX2 对应按钮 16-23，在该 MUX 上按钮 21 的通道是第 6 个通道（其 index 为 5）。

图 5: 连接通路原理图



除了要使能外部 MUX 及其控制线之外，还必须为内部 4 通道 MUX 选择相应的比较器输入。由于在 C12INx- 上 MUXx 的定义如前所述，整数除法 $\text{index}/8$ 也能方便地给出针对 C1CH 和 C2CH 的比较器输入通道位。下面的代码示例说明了在所述配置中，如何扫描 32 个按钮。

例 3: 代码示例

```
void isr {
    // 如果是电容触摸中断
    if (TOIF == 1) {
        // 获取 TMR1 读数
        StopTimers();
        GetMeasurement();

        // 测试是否按下
        if (IsButtonPressed(index))
            SetFlag(index);
        else
            ClearFlag(index);

        // 进行 16 点平均值计算
        PerformAverage(index);

        // 设置下一个传感器
        // *** 与之前不同 ***

        // index 增一或回到 0
        if (index < 31)
            index++;
        else
            index = 0;

        div = index / 8;
        rem = index % 8;

        // 使能正确的 MUX
        // (假定 MUX EN 在 RB<7:4> 线上
        // 是低电平有效)
        switch(div) {
            // 使能 MUX 0、1、2 或 3
            // 禁止全部, 然后设置正确的、
            // 已使能的 MUX, 低电平有效
            case 0: PORTB |= 0xF0; RB4=0; break;
            case 1: PORTB |= 0xF0; RB5=0; break;
            case 2: PORTB |= 0xF0; RB6=0; break;
            case 3: PORTB |= 0xF0; RB7=0; break;
            default: break;
        }

        // 设置 MUX 通道
        // (假定通道线 = RB<2:0>)
        PORTB &= 0xF8;      // 清除 <2:0>
        PORTB |= rem;      // 设置 3 个 LSB

        // 设置比较器
        CM1CON0 = COMP1[div];
        CM2CON0 = COMP2[div];

        RestartTimers();
    }
}
```

方法比较

处理按钮的每一种方法, 都有其优缺点。下表比较了每种方法的一些关键特性。

表 2:

	最大按钮数量	一次按下 2 个按钮	扫描速度	I/O 线
默认	4	YES	++	+
配对按压	10	NO	+	+
多路复用	N*	YES	-	-

* 由于最终受到检测的物理限制, 按钮个数并不能任意增加。

对于需要 4 个或更少按钮的应用而言, 只要使用默认按钮就足够了。通常, 对于只需要一到两个按钮的应用, 使用默认硬件设置是最适宜的。它扫描速度最快, 由于寄生电容更小, 与多路复用输入相比, 远距离感应能力更好。

如果按钮不超过 10 个, 而且不需要同时按下两个按钮的话, 使用配对按压方法较经济。除了增加布线和按钮焊盘之外, 使用配对按压方法并不需要添加外部器件。这种简单的硬件扩展需要较复杂的软件。

多路复用按钮系统的最大优点在于按钮数可以超过 10 个。它还允许同时按下两个按钮。但是扫描速度取决于要扫描的输入个数。随着按钮数量的增加, 扫描速度也将加大, 扫描速度是有极限的, 从而最大按钮数量将受到限制。改变 Timer0 的预分频数将加快或者减慢扫描速度, 这可能会对系统的长时间扫描有所帮助, 但也将对传感过程本身产生影响。如“通过多路复用器进行扩展”一节所叙述的那样, 随着添加的 MUX 的增多, 最终寄生电容将变得非常大, 导致无法检测按压。由于上述若干因素的存在, 使得按钮数量不能任意增加, 不过可使用的按钮数量能够大大超过默认或配对按压方法所允许的按钮个数。

总结

可以使用若干方法来对多个电容触摸传感按钮进行控制。PIC16F616 系列、PIC16F690 系列以及 PIC16F887 系列均可直接处理四个按钮的情况，而且不使用外部器件就可扩充到 10 个按钮。如果需要更多的按钮，可以使用外部多路复用器。

在电容中断服务程序中，主要的处理方法是相同的，但在处理多个按钮时需要正确跟踪记录哪个 **index** 代表的是哪个实际按钮。在实际设计过程中，规划好软件中 **index** 与实际按钮之间的关系，将使得设计的软件部分更易于实现、编写和理解。

其他相关应用笔记是 AN1101 《电容触摸传感简介》、AN1102 《电容触摸传感器布板和物理设计指南》以及 AN1103 《电容触摸传感的软件处理》。

注:

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下，Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。

代码保护功能处于持续发展中。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案 (Digital Millennium Copyright Act)》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

提供本文档的中文版本仅为为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

Microchip 的名称和徽标组合、Microchip 徽标、Accuron、dsPIC、KEELOQ、KEELOQ 徽标、MPLAB、PIC、PICmicro、PICSTART、PRO MATE、rPIC 和 SmartShunt 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

FilterLab、Linear Active Thermistor、MXDEV、MXLAB、SEEVAL、SmartSensor 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、Application Maestro、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、In-Circuit Serial Programming、ICSP、ICEPIC、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、mTouch、PICKit、PICDEM、PICDEM.net、PICtail、PIC³² 徽标、PowerCal、PowerInfo、PowerMate、PowerTool、REAL ICE、rLAB、Select Mode、Total Endurance、UNI/O、WiperLock 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 是 Microchip Technology Inc. 在美国的服务标记。

在此提及的所有其他商标均为各持有公司所有。

© 2008, Microchip Technology Inc. 版权所有。

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2002 认证。公司在 PIC[®] MCU 与 dsPIC[®] DSC、KEELOQ[®] 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外，Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。



MICROCHIP

全球销售及服务中心

美洲

公司总部 **Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277

技术支持:
<http://support.microchip.com>
网址: www.microchip.com

亚特兰大 Atlanta
Duluth, GA

Tel: 678-957-9614
Fax: 678-957-1455

波士顿 Boston
Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago
Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

达拉斯 Dallas
Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit
Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

科科莫 Kokomo
Kokomo, IN
Tel: 1-765-864-8360
Fax: 1-765-864-8387

洛杉矶 Los Angeles
Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣克拉拉 Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

加拿大多伦多 Toronto
Mississauga, Ontario,
Canada
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 **Asia Pacific Office**
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 北京
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

中国 - 成都
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 - 香港特别行政区
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 南京
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青岛
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 沈阳
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深圳
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

中国 - 武汉
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 - 厦门
Tel: 86-592-238-8138
Fax: 86-592-238-8130

中国 - 西安
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 - 珠海
Tel: 86-756-321-0040
Fax: 86-756-321-0049

台湾地区 - 高雄
Tel: 886-7-536-4818
Fax: 886-7-536-4803

台湾地区 - 台北
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

台湾地区 - 新竹
Tel: 886-3-572-9526
Fax: 886-3-572-6459

亚太地区

澳大利亚 Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

印度 India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 Japan - Yokohama
Tel: 81-45-471-6166
Fax: 81-45-471-6122

韩国 Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

韩国 Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

马来西亚 Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

菲律宾 Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 Denmark-Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

法国 France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

西班牙 Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

01/02/08