

## 使用电压频率转换器实现模数转换

作者: Paul Klonowski

### 简介

电压频率转换器(VFC)接受模拟电压或电流信号输入, 输出一个频率与输入值成比例的脉冲串或方波序列。因此, 电压频率转换器可以用作模数转换系统的构建模块, 用于使计数器运行一定的时间(“计数时间”、“选通时间”或“转换时间”), 并读取输出数字字。此数字字将与模拟输入成比例。

模数转换方案使用电压频率转换器有多个好处。首先, 不同于基于二进制加权网络的转换器, 它在所有电源和温度条件下都具有内在单调性。其次, 输入信号被转换为易于传输的串行位流, 因而模拟电路(VFC和模拟信号调理电路)可以靠近信号源放置, 数字电路(计数器、定时门和显示电路)则可以放在任意位置。当需要大量通道时, 这尤其有利; 远程VFC可以用来执行“每通道转换器”数据采集。最后, 数字值是经过大量周期的积累而得到的, 因此对于干扰信号的积分和消除也是过程本身所固有的功能。

模拟信号转换为数字字所需的时间与VFC的最大满量程频率和所需的测量分辨率有关。例如, ADI公司AD650 VFC的满量程频率为1MHz。如果将此器件用于要求16位或1/65,536分辨率的应用中, 则模拟信号转换为16位数字字所需的时间为65.536 ms。18位或1/262,144分辨率将需要略长于0.262秒的计数时间。一般而言, 利用VFC进行模数转换所需的计数时间为:

$$T_{\text{COUNT}} = \frac{N}{FS_{\text{out}}}$$

其中: N为给定分辨率的码数,  $FS_{\text{out}}$ 为VFC满量程输出频率。

虽然基于VFC的模数转换器比逐次逼近型和Flash转换器要慢, 但在速度上与积分模数转换器相当。因此, 基于VFC的模数转换器非常适合温度和应变计测量等低频应用。VFC在这类应用中所能提供的分辨率, 可以抵消获取与模拟输入值相对应的数字字所需计数时间相对较长这一劣势。

本应用笔记的主旨是介绍电压频率转换器用作模数转换构建模块的不同方法。因此, 本文的重点是讨论VFC的输出接口。有关处理各类输入(温度、应变计和光电二极管信号)的详细信息, 建议读者参考AD650、AD651和AD654数据手册以及AD654应用笔记; ADI公司文献中心(电话: (617) 329-4700)可以提供这些资料。上述资料还包括信号多路复用方案、隔离电路和个别器件的详情。

### 脉冲计数

利用电压频率转换器执行模数转换的一种方法是让一个单芯片微电脑计数一定期间内出现的脉冲数。此期间内计数的总脉冲数即与VFC的输入电压成比例。例如, 如果1V满量程输入在VFC中产生一个100kHz信号, 且计数周期为100ms, 则总满量程数为10,000。然后依据与该最大值的比例便可确定输入电压, 例如脉冲数5,000对应于0.5V输入电压。

图1显示ADI公司AD654 VFC的输出连到Intel 8051微电脑的计数器输入T1。AD654是一款低成本、单电源、单芯片VFC, 满量程频率最高可达500kHz。8051属于Intel MCS-51系列8位微电脑, 该系列不同成员之间的区别主要在于内存容量。下文中, “8051”泛指MCS-51系列的所有成员。

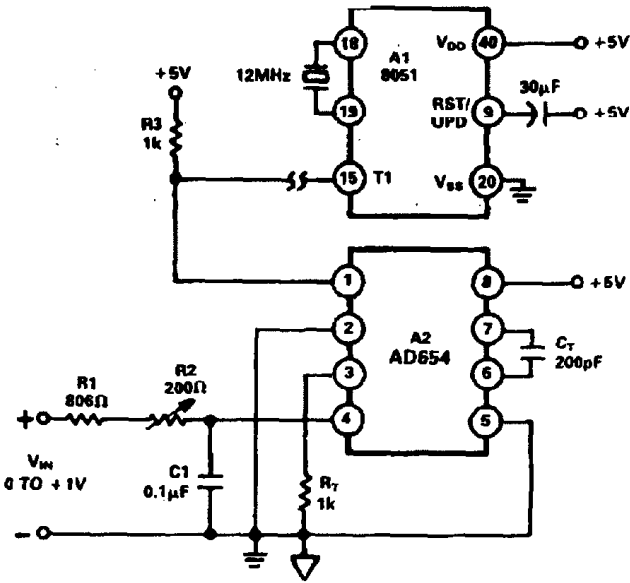


图1. AD654脉冲计数器

图1中，AD654的模拟输入是一个0至+1V信号。所选定时电阻 $R_T$ 和定时电容 $C_T$ 使得引脚4上的此0至+1V信号产生0至500kHz输出频率。上拉电阻 $R_3$ 确保AD654输出达到8051的T1(引脚15)所要求的逻辑电平。

8051具有两个片内16位定时器/事件计数器(8052和8032则有三个)。可以对这些计数器(定时器0和定时器1)进行独立编程，以用作16位时间间隔或事件计数器。定时器0和定时器1的使用由两个8位寄存器TMOD(定时器模式)和TCON(定时器控制)决定。TMOD寄存器如下面图2所示：

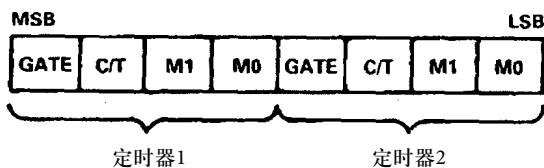


图2. 8051 TMOD寄存器

M1和M0用于选择各定时器的模式。模式01将定时器配置为16位时间间隔或事件计数器。C/T为定时器或计数器选择器，清零则选择定时器工作方式。本应用中，定时器0配置为定时器(用来提供固定时间间隔)，定时器1配置为计数器(用来计脉冲个数)。下文中，这两个定时器分别称为定时器0和计数器1。运行时，定时器0以一定的速率递增，该速率等于外部时钟速率除以12。如果使用12MHz晶振，则定时器0每微秒加1。GATE为选通控制。如果此位清零，则只要TCON寄存器中的TRx控制位置1，就会使能定时器/计数器X。TRx位通过软件控制。如果GATE位置1，则只要TRx位置1，并且引脚(引脚12或13，分别针对定时器0或1)上的信号电平为高，就会使能定时器/计数器X。因此，当GATE位清零时，该定时器仅由软件控制；当GATE位置1时，该定时器由软件和硬件组合控制。本应用

中，GATE位清零；但在下一个应用中，GATE位则是置1。

表I所列为软件程序PLSECNT，用于计数50ms内T1(计数器1输入)上出现的下降沿数目。计数器1清零后，值15539载入定时器0。由于定时器0为16位定时器，因此最大可能计数为65535。在定时器0中断使能的情况下，计数65536将导致程序跳转至定时器0中断服务程序的起始地址(OBH)。定时器0从15539开始计数，每微秒加1(基于12MHz时钟)，因此跳转至中断服务程序之前将有49,997次计数或49.997ms时间。与50ms相比，存在3µs时间差，它由中断响应的速度决定。使用12MHz晶振时，中断响应延时范围是从3µs至7µs。在这50ms计数期间内，控制权归于主程序。因此，当计数器1计数50ms时，8051并未被占用。到达中断服务后，计数器1和定时器0停止，计数器1的内容移入RAM，用户可以在方便的时候予以存取。然后，控制权回归子程序所隶属的主程序。当最大频率为500kHz且计数窗口为50ms时，计数器1的最大值为25,000，这可以提供14位以上的分辨率。相对于此1V满量程参考点的比例计算可以在软件中执行。

	ORG	00H	
	AJMP	MAIN	
PLSECNT	ORG	60H	;脉冲计数器程序
	MOV	TMOD,#5BH	;将计数器0和计数器置于模式1
	MOV	TL1,#00H	;初始化计数器寄存器
	MOV	TH1,#00H	
	MOV	TL0,#0B3H	;将15536+载入定时器
	MOV	TH0,#3CH	;将在50ms+3µs延迟后溢出
	SETB	PT0	;定时器优先中断
	SETB	ET0	;使能定时器中断
	SETB	EA	;使能全局中断
	SETB	TR0	;启动定时器
	SETB	TR1	;启动计数器
	RET		;返回主程序
	ORG	OBH	;定时器中断子程序
	CLR	TR1	;停止计数器
	CLR	TR0	;停止定时器
	AJMP	COUNT	
COUNT	MOV	50H,TL1	;将计数器内容移入RAM
	MOV	51H,TH1	
	RETI		;从中断返回
	ORG	100H	
MAIN	-	-	;PLSECNT子程序所隶属的主程序

表I. 8051脉冲计数程序

## 周期计时

利用电压频率转换器(VFC)和微电脑执行模数转换的另一种方法是让微电脑确定VFC输出频率的周期。例如, 25kHz输出频率的周期为40 $\mu$ s。如果将一个每微秒加1的定时器选通此信号, 则将产生总数为40的计数。250Hz输出频率的周期为4ms, 选通此周期信号的同一定时器将产生总数为4000的计数。

与脉冲计数相比, 周期计时的优势之一是计数窗口取决于VFC的输出频率; 许多情况下, 周期计时的计数窗口短于脉冲计数的计数窗口。这在转换多个通道的系统中尤为重要。在上面的脉冲计数器应用中, 无论输出频率是50kHz还是50Hz, 计数窗口均为50ms。对于周期计时, 计数窗口则为输出频率的倒数。因此, 50kHz信号将具有20 $\mu$ s的计数窗口, 50Hz信号则具有20ms的计数窗口。事实上, 只有当输出频率达到20Hz时, 周期定时的计数窗口才等于脉冲计数的计数窗口50ms。

图3显示利用ADI公司电压频率转换器AD650, 通过周期计时方式执行模数转换所需的电路。AD650的最大满量程频率为1 MHz, 此频率时的最大非线性误差为0.1%。图3中, AD650按如下要求配置: 0至+10V输入产生0至50kHz输出频率。AD650的输出由脉冲组成, 因此使用SN7474 D触发器将这些脉冲转换为一个方波。图3中草绘的SN7474引脚3和引脚5波形显示, 引脚5上出现的高电平或低电平输出的宽度与AD650输出频率的一个周期相同。还应注意, 当SN7474的引脚1保持低电平时, 引脚5也保持低电平。

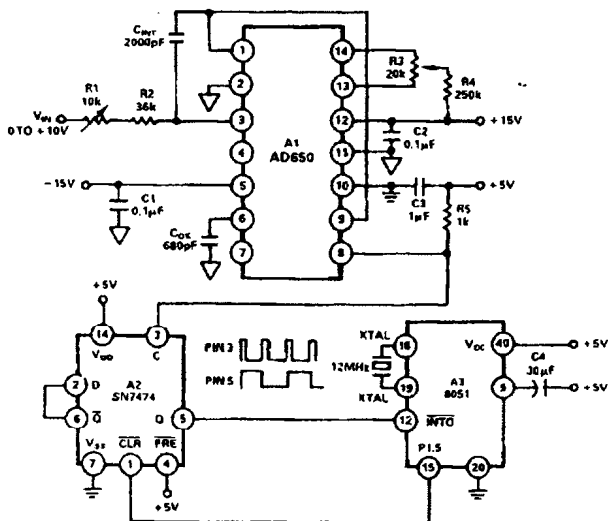


图3. AD650周期计时

如前所述, 8051的 $\overline{\text{INT0}}$ 引脚(引脚12)是定时器0选通引脚。(参见脉冲计数器部分)当TMOD寄存器中的GATE位置1时, 只有当引脚12的 $\overline{\text{INT0}}$ 为高且TCON寄存器中的TR0已通过软件置1时, 定时器0才会运行。因此, 将SN7474的Q输出连到8051的 $\overline{\text{INT0}}$ 引脚可以确保定时器运行AD650频率的一个周期。

可能会发生这样一个问题, 即软件在 $\overline{\text{INT0}}$ 引脚的一个高电平中间将TR0位置1。此时, 定时器0将只运行一个周期的一部分时间, 而不是一个完整周期。将8051端口1位5 (P1.5)引脚连到SN7474  $\overline{\text{CLR}}$ 引脚, 可以防范这一问题。当 $\overline{\text{CLR}}$ 为低且PRE为高时, Q为低。如果 $\overline{\text{CLR}}$ 和 $\overline{\text{PRE}}$ 均为高, 则每当时钟(C)引脚上出现正边沿时, Q就会改变状态。因此, 将P1.5设为低, 并通过软件将TR0置1, 然后将P1.5设为高, 可以确保定时器0运行一个完整周期。

表II所示为软件子程序PCNT, 用于在AD650的一个输出频率周期内, 使定时器0每微秒加1。注意有两个中断服务程序, 一个用于 $\overline{\text{INT0}}$ , 一个用于定时器0。对 $\overline{\text{INT0}}$ 服务程序的访问发生在 $\overline{\text{INT0}}$ 引脚(引脚12)上出现负边沿之后, 表示一个周期已结束。然后定时器停止, 其内容载入RAM。用户可以在方便的时候访问该内容。

	ORG	00H	
	AJMP	MAIN	
PCNT	ORG	90H	;周期计数器子程序
	MOV	TMOD,#05H	;将定时器0置于模式1。使能 $\overline{\text{INT0}}$ 引脚
	CLR	P1.5	;将 $\overline{\text{INT0}}$ 引脚初始设置为低
	SETB	IT0	;指定边沿触发中断
	MOV	TL0,#00H	;初始化定时器
	MOV	TH0#00H	
	SETB	EX0	;使能 $\overline{\text{INT0}}$ 中断
	SETB	ET0	;使能定时器0中断
	SETB	EA	;使能所有中断
	SETB	TH0	;启动定时器
	SETB	P1.5	;使能选通 $\overline{\text{INT0}}$ 引脚
	RET		;返回主程序
	ORG	03H	;INT0子程序服务程序;
	CLR	TR0	;停止定时器
	CLR	EA	;禁用中断
	AJMP	COUNT	;跳转至Count
	ORG	0BH	;定时器0子程序服务程序
	CLR	TR0	;停止定时器
	CLR	EA	;禁用中断
	AJMP	OFLW	;跳转至OFLW
	ORG	40H	
OFLW	MOV	60H,#FF	;将溢出值载入RAM
	MOV	61H,#FF	
	CLR	P1.5	;将 $\overline{\text{INT0}}$ 引脚设为低
	RET1		;从子程序返回
	ORG	50H	
COUNT	MOV	60H,TH0	;将计数器内容载入RAM
	MOV	61H,TL0	
	CLR	P1.5	;将 $\overline{\text{INT0}}$ 引脚设为低
	RET1		;从子程序返回
	ORG	100H	
MAIN	-	-	;子程序所隶属的主程序

定时器0服务程序用来将计数窗口限制在约65.5ms。对此服务程序的访问发生在定时器0达到65536时，即AD650输入电压约为3.05mV，或输出频率约为15.26Hz时。然后，此服务程序将溢出值65535载入RAM。完成两个中断子程序且初始化PCNT子程序之后，控制器回归子程序所隶属的主程序。因此，在周期计时期间，8051并未被占用。

本应用的一个可能误差源是抖动，即输出频率周期的变化范围。周期的变化会导致从一个周期到下一个周期中计数的脉冲数不同。在软件中求取多个周期计数的平均值，并利用此平均值进行计算，可以大大降低这种误差的幅度。

### 完整的16位分辨率模数转换系统

一个完整的16位分辨率模数转换系统如图4所示，它使用ADI公司AD651作为其VFC。AD651是一款2MHz满量程输出同步电压频率转换器，利用外部时钟产生满量程频率，而不是依赖无源元件的稳定性。因此，AD651的线性和稳定性远远优于现有的其它单芯片VFC。此系统的其它重要元件包括单芯片计数器-解码器-LED驱动器Intersil 7208、4MHz TTL振荡器和2个4020B二进制计数器。

AD651配置为0至+10V输入、2MHz输出模式。上拉电阻R1用来将AD651输出频率直接馈送至7208计数器-解码

器-驱动器的计数器输入引脚。4020B为14级二进制纹波计数器，具有时钟和主复位输入(引脚10和11)，第一级和后11级提供缓冲输出。4020B用来产生7208所用的计数窗口。将4MHz TTL时钟除以 $2^{18}$ (由两个 $2^9$ 分频级级联而成)，得到一个15.2588Hz信号。当引脚13为低时，7208将计数AD651输出脉冲的负边沿。对于15.2588Hz信号，计数窗口为32.768ms。

15.26Hz信号除了连到计数使能引脚之外，还连到双通道单稳态多谐振荡器74LS221。

图5显示了单稳态74LS221的作用。7208的计数窗口关闭后(即计数使能输入变高)，必须锁存数据，然后进行解码，以便驱动LED。锁存此数据后，为了在下一个计数窗口中获得准确计数，计数器必须清零。74LS221的作用即是提供所需的STORE输入和STORE输入脉冲。虽然7208只需要大于50 $\mu$ s的脉冲宽度，但所选 $R_3 \cdot C_2$ 和 $R_4 \cdot C_3$ 可提供大约500 $\mu$ s的脉冲宽度。

7208自动处理LED的解码和驱动，所需的其它元件只有电阻 $R_5$ 、 $R_6$ 和电容 $C_4$ 。这些非关键无源元件控制LED的显示复用速率，其选择依据是制造商的指南，可提供50Hz至200Hz范围内的复用率。

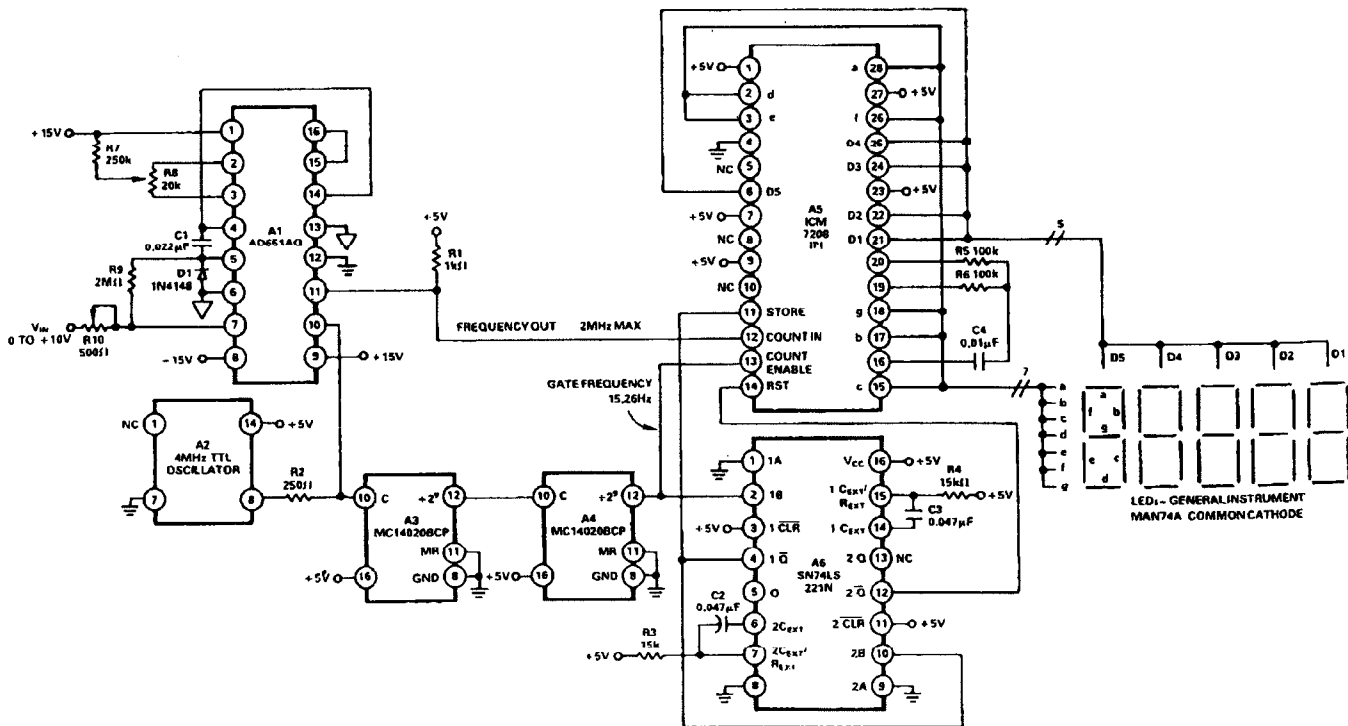


图4. 完整的16位模数转换系统

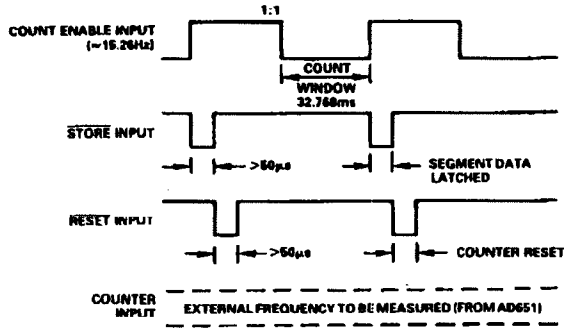


图5. ICM7208时序图

为实现最佳性能，固定选通时间间隔(或“计数窗口”)应利用AD651 SVFC时钟输入的倍数产生，本应用即是如此。以这种方式计数可消除由于时钟(抖动、时间或温度漂移等)引起的误差，因为测量的是时钟与输出频率的比值。

模数转换的分辨率当然是由时钟频率和选通时间决定。例如，如果需要12位分辨率，而时钟频率为1MHz(导致AD651 FS频率为500kHz)，则选通时间为：

$$\left(\frac{FS\ Freq}{N}\right)^{-1} = \left(\frac{1\ Clock\ Freq}{2\ N}\right)^{-1} = \left(\frac{1MHz}{2(4096)}\right)^{-1}$$

$$= \frac{8192}{1 \times 10^6} sec = 8.192ms: \text{其中} N \text{是给定分辨率对应的总码数。}$$

表III给出了各种分辨率时AD651时钟频率与选通时间之间的关系。本应用要求16位分辨率，使用4MHz时钟，因此所需选通时间为32.77ms。

**消除已知干扰信号**

模数转换系统的误差源之一是干扰信号耦合至待转换的模

拟信号中。例如，不需要的电力线能量耦合经常以正弦波形式出现在要转换的直流电平上，导致数字输出错误。这种干扰正弦波的频率是已知的(50Hz或60Hz)，因此使选通时间等于该正弦波周期的倍数，就可以消除拾取所造成的误差。可以截取邻近变压器的干扰信号的复制品，将其馈入一个锁相环，如图6所示的框图。此环路提供两个输出信号：一个高频时钟(干扰信号的高次谐波)和一个选通时钟(干扰信号的低次谐波)。

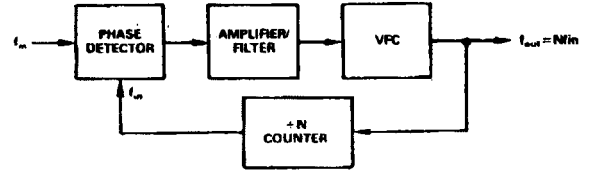


图6. 锁相环

利用f<sub>OUT</sub>为AD651时钟源，并利用N分频计数器的f<sub>IN</sub>为计数窗口源，可以实现1/2 N分之一的分辨率，其中N为计数器的“±N”。如果计数窗口由电平触发，而不是由边沿触发(像7208一样)，则分辨率为1/4N。

图7是图6的硬件实现图。图7中的MC4044内置框图所示的鉴相器和放大器/滤波器两个器件。干扰信号为60Hz，被转换为TTL电平信号，以馈入MC4044。选择适当的元件R1、R2和C1，以便处理50Hz或60Hz干扰信号。来自MC4044的误差电压馈入AD654，后者配置为0至+1V输入、0至500kHz输出模式。由于双通道4位二进制计数器74LS393配置为提供8192的“±N”，因此AD654的输出频率为491520Hz，此信号用作AD651的时钟。第二计数器的输出为60Hz，不仅反馈至MC4044，而且用作频率计数器选通信号。

分辨率	N	时钟	转换或选通时间	典型线性度	备注
12位	4096	81.92kHz	100ms	0.002%	50,60,400HzNMR
12位	4096	2MHz	4.096ms	0.01%	
12位	4096	4MHz	2.048ms	0.02%	
4位数	10000	200kHz	100ms	0.002%	50,60,400HzNMR
14位	16384	327.68kHz	100ms	0.002%	50,60,400HzNMR
14位	16384	1.966MHz	16.66ms	0.01%	60HzNMR
14位	16384	1.638MHz	20ms	0.01%	50HzNMR
4 1/2 Digits	20000	400kHz	100ms	0.002%	50,60,400HzNMR
16位	65536	655.36kHz	200ms	0.002%	50,60,400HzNMR
16位	65536	4MHz	32.77ms	0.02%	

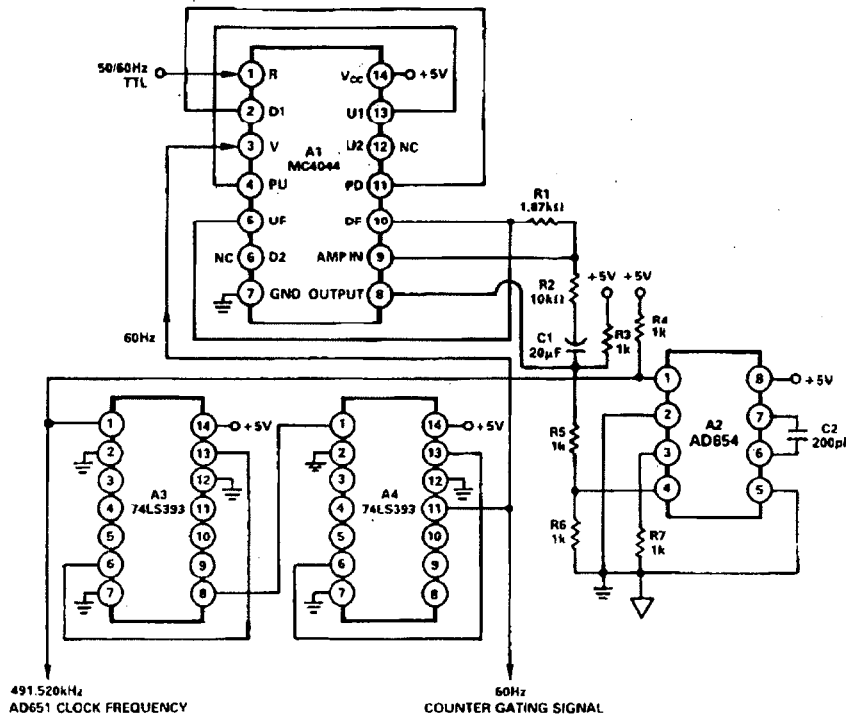


图7. 消除已知干扰信号

在此系统中，利用A4的引脚10、9或8为频率计数器选通信号，可以获得更高的分辨率。引脚11则必须反馈至MC4044。从A3的引脚1到A4的引脚11， $N = 8192$ 或 $2^{13}$ ，可提供12位分辨率。如果利用引脚10、9或8提供选通频率，则分辨率分别为13位、14位或15位。

#### 基于MC6801-AD650的模数转换

某些应用中，可能需要利用不含片内定时器/计数器或没有可用片内定时器/计数器的微处理器执行模数转换。使用一些附加硬件，仍然可以在一定的期间内计数VFC的输出脉冲，并将结果存储在微处理器的RAM中，用户可以在方便的时候进行访问。

图8给出了利用AD650 VFC和MC6801微处理器执行模数转换所需的电路，其中假设片内计数器已为其它功能所专用。MC6801是一款8位单芯片微电脑，具有2048字节ROM、128字节RAM、一个串行通信接口和一个三功能可编程定时器。AD650 VFC配置为0至+10V输入、0至1MHz输出模式。

计数AD650输出脉冲所需的附加硬件包括：两个带输出寄存器的8位二进制计数器74590，一个14级二进制计数器4020B，一个带预设和清零功能的双通道D触发器7474，以及一个16进制反相器74LS04。

4020B和7474提供定时信号，告知计数器何时开始和停止计数。4020B的输入端连到MC6801的E引脚(引脚40)。E引脚上的信号为MC6801外部晶振频率除以4，本例中为1.2288MHz。4020B将1.2288MHz信号除以 $2^{14}$ ，得到一个75Hz信号，馈送至7474的引脚3。7474进一步将此信号除以4，得到一个18.75Hz信号。根据引脚1和10的信号电平，此信号可能会出现在引脚9上。如果引脚1和10为低电平，引脚9将保持TTL高电平。如果引脚1和10为高电平，18.75Hz方波将出现在引脚9上。当引脚9为高电平时，74590计数器禁用，不会执行计数。如果引脚9的输出为方波，计数器将在周期的低电平期间使能，并计数26.67ms。请注意，计数器使能与否由端口1的位4 (P14)控制。如果该位为低，则不会发生计数。还应注意，引脚9通过一个反相器连到MC6801的外部中断请求( $\overline{IRQ1}$ )引脚。当引脚9从低电平变为高电平时，将产生两个结果。第一个结果是禁用计数器，因为A2的CCKEN与引脚9相连。第二个结果是低电平至高电平转换反向，在 $\overline{IRQ1}$ 线上产生中断请求。然后，MC6801清除P14，防止在74590计数器值被读取之前再次发生计数。

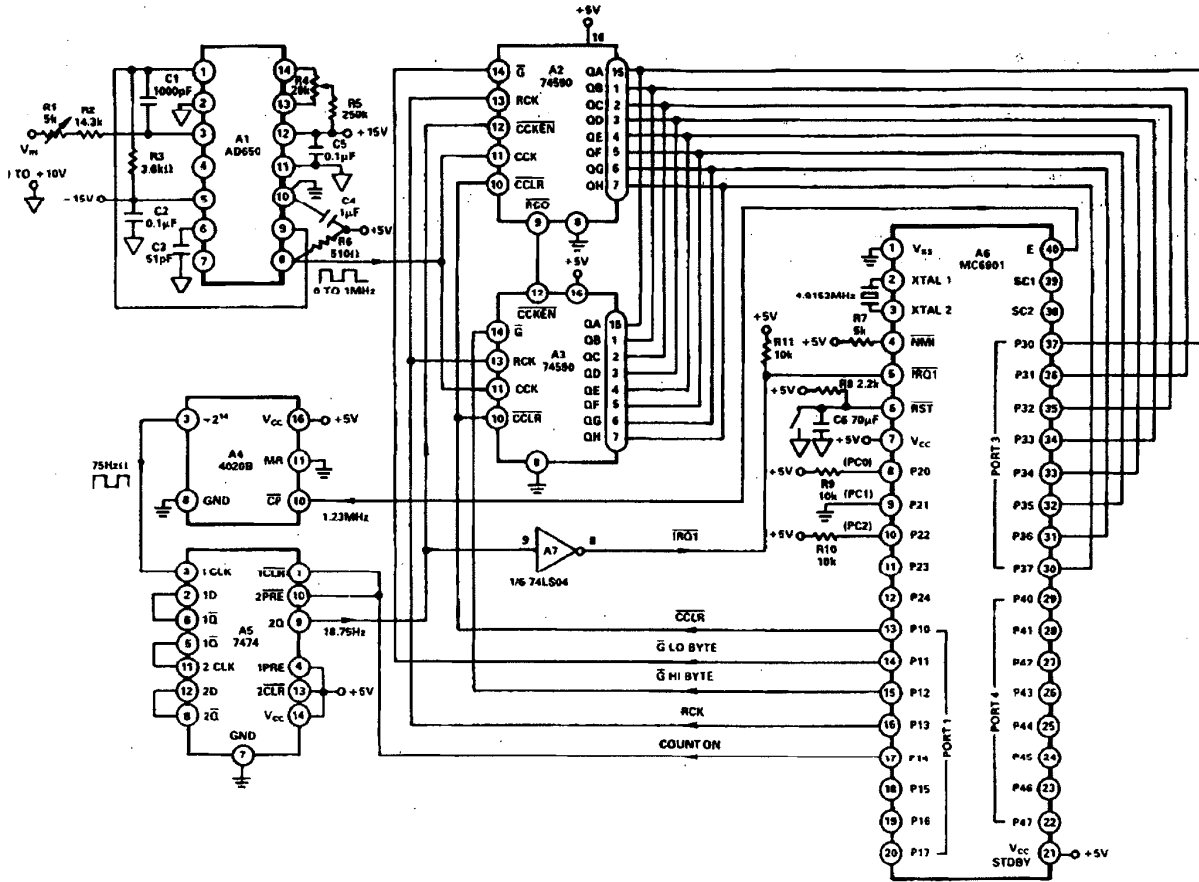


图8.AD650-MC6801模数转换系统

所有计数事件都由端口1不同位的信号电平控制。将不同的值写入端口1，可以清零、使能、禁用、锁存或读取74590计数器。控制这些功能的值如表IV所示。

端口1配置						
事件	P4	P3	P2	P1	P0	Hex
计数器清零	0	0	1	1	0	06
使能计数器	1	0	1	1	1	17
禁用计数器	0	0	1	1	1	07
锁存数据	0	1	1	1	1	0F
输出高字节	0	1	1	0	1	0D
输出低字节	0	1	0	1	1	0B

表IV. 端口1事件控制值

表V给出了控制AD650脉冲计数过程的软件程序。此程序设置堆栈指针，禁用中断，清零74590计数器，经过一定延迟时间后使能中断(这将在稍后讨论)。SN7474引脚9的低电平至高电平转换触发中断请求之后，程序计数器跳转至中断程序。此程序禁用所有后续中断，关闭74590计数器，并且读取低字节和高字节。然后，堆栈条件代码寄存器的中断位置1，程序计数器从中断程序返回。

```

0E ooco BEGIN      ORG 0100          计数程序
0F                LDS #SC0          设置堆栈指针
86 06              SEI                禁用中断
97 02              LDAA #S06          计数器清零
86 17              LDAA #S17          打开计数器
97 02              STAA $02
86 2F              LDAA #S2F          插入13.3ms以上延迟
C6 7F              LDBB #S7F          以防中断错误
5A FD              AGN CNT
2E FD              DECB CNT
4A FD              DECA CNT
2E FD              BGT AGN
0E                CLI                允许中断
39                RTS                从子程序返回
0080              ORG FFF8          LDCNT
                  AIRQ1 FDB          定义中断程序
                  ORG 0080          起始点
0F                SEI                中断程序
86 07              LDAA #S07          禁用中断
97 02              STAA $02          关闭计数器
86 0F              LDAA #S0F          锁存计数器中的数据
97 02              STAA $02
66 0D              LDAA #S0D          输出计数器低字节
97 02              STAA $02
90 06              LDAA $06          从端口3读取字节
97 A7              STAA $A7          存储于位置00A7
B6 0B              LDAA #S0B          输出计数器高字节
97 02              STAA $02
96 06              LDAA $06          从端口3读取字节
97 A6              STAA $A6          存储于位置00A6
85 07              LDAA #S07          关闭计数器
97 02              STAA $02
86 10              LDAA #S10
9A BA              ORAA SBA          堆栈条件代码寄存器的
97 BA              STAA SBA          中断位置1
3B                RTI                从中断返回
    
```

表V. MC6801脉冲计数程序

软件程序中有两个地方需要细加讨论：使能MC6801中断之前的时间延迟和堆栈条件代码寄存器的中断位置1。

图9显示了将引脚1和10设为高电平以指示74590计数器开始计数时出现的波形。

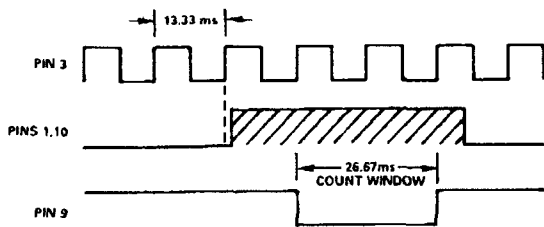


图9. SN7474波形

如前所述，当引脚1和10为高电平时，引脚9只会输出一个18.75Hz方波。引脚1和10变为高电平后，引脚9不会改变状态，直到引脚3上再次出现正边沿。图9显示了将会发生的最差情况：引脚3上出现趋正边沿后仅几纳秒，引脚1和10就变为高电平。引脚13.33ms。另外，引脚9通过反相器馈至IRFQ1引脚，从而在13.33ms等待状态期间将信号电平设

为低。IRFQ1对电平敏感，因此等待状态期间的中断将导致程序跳转至中断程序，而此时26.67ms计数窗口尚未开启。在允许中断之前插入13.33ms以上的延迟，就可以在计数窗口期间使能中断(当引脚9为低电平时，这样IRFQ1引脚为高电平)，从而确保中断程序是在26.67ms计数窗口关闭之后进行访问。

堆栈条件代码寄存器的中断位置1也很重要。软件执行CLI指令后，条件代码寄存器(CCR)的位4清零，这将使能IRQ1中断。一旦检测到中断请求(IRQ1引脚为低电平)，就会将CCR以其当前状态压入堆栈。总共会将7个字节压入堆栈，最后一个字节为CCR。堆栈指针设在位置00C0，因此CCR的位置在其下7个字节，或位置00BA。用10H与位置00BA的内容求“或”将禁用中断，从而防止程序计数器在刚刚跳出来之后又跳转至中断程序，即使检测到IRQ1引脚为低电平。如果该位未置1，则程序计数器跳出中断程序，看到中断使能位清零，并且IRFQ1引脚为低电平之后，将会再次跳回中断程序。