ANALOG DEVICES

AN-765
应用笔记

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106 • Tel: 781/329-4700 • Fax: 781/461-3113 • www.analog.com

ADuC702x系列的短脉冲持续时间测量

作者：Aude Richard

简介

ADuC702x具有四个外部中断，这些中断只能配置为电平触发且低电平有效模式。因此，利用这些外部中断测量短脉冲需要某种外部胶合逻辑组合。

本应用笔记描述一种利用定时器1和PLA来测量短脉冲持续时间(数毫秒)的方法。这种技术不需要任何外部数字逻辑。

原理

PLA(可编程逻辑阵列)可以看作是胶合逻辑，因此无需简单的外部逻辑。它由16个单元组成，每个单元都包含一个双输入的查找表，通过配置可以实现任何基于单输入或双输入的逻辑功能。

PLA可以路由至内部中断系统；在中断控制器中，PLA有两个专用中断位。

定时器1是一个通用定时器，并且具有捕获事件模式等额外特性。定时器1捕获寄存器(T1CAP)可以被选定的中断源触发。与在进入ISR时定时器被触发相比，此功能可用来更精确地确定事件的开始时间。本应用笔记中即使用了此捕获事件功能。

硬件考虑

可以使用任何能用作PLA输入的GPIO来测量脉冲。本方法使用两个单元来利用两个中断源，如图1所示。

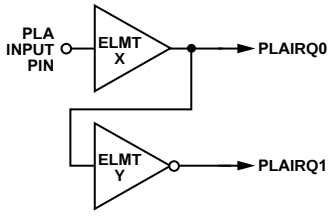

图1.PLA配置

脉冲通过单元x。当单元x的输入变为高电平时，其输出便会触发PLAIRQ0。

单元x的输出反馈至单元y，后者被配置为“非”门。当输入信号变回低电平时，单元y的输出便会触发PLAIRQ1。

软件

所有处理均通过中断服务程序(ISR)完成，如图2所示。

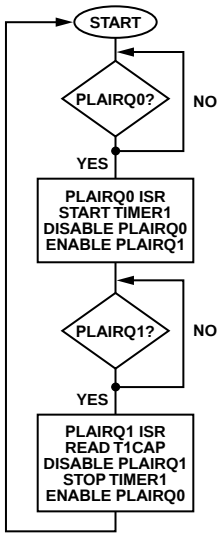

图2.流程图

定时器1在PLAIRQ0 ISR中启动。PLAIRQ0禁用，以确保无法重新进入ISR，并且PLAIRQ1使能。

在PLAIRQ1 ISR中，读取定时器1自动捕获并存储在T1CAP中的值。PLAIRQ1禁用，以确保无法重新进入ISR，并且PLAIRQ0重新使能，允许进行新的测量。定时器1也需要停止和复位。

参见文后所附的源代码。

限制和精度

PLAIRQ0中断会延迟定时器1的启动。ADuC702x的中断延迟时间为5到50个处理器周期；在本例中(采用连续45 MHz处理器时钟)，延迟时间稍稍超过1.1 μs。因此，本方法应只适用于测量毫秒范围或更长持续时间的脉冲。

```
#include<aduc7020.h>

long pulse;

void My_IRQ_Function(void);         // IRQ Function Prototype

int main (void) {

  IRQ = My_IRQ_Function;            // Specify Interrupt Service Routine

  PLAELM0 = 0x0035;                 // pass
  PLAELM1 = 0x0047;                 // not
  PLAIRQ = 0x1110;                  //

  IRQEN = 0x080000;                 // enable PLA IRQ0

  while (1){
  }
}

/*********************************************************************/
/*                                                                   */
/*     Interrupt Service Routine                                     */
/*                                                                   */
/*********************************************************************/

void My_IRQ_Function()
{
  if ((IRQSTA & PLA_IRQ0_BIT) == 0x00080000)       // PLAIRQ0
  {
    T1CON = 0x32180;                // start Timer1. capture PLAIRQ1
    IRQCLR = 0x80000;               // disable PLA IRQ0
    IRQEN = 0x00100000;             // enable PLA IRQ1
  }
  if ((IRQSTA & PLA_IRQ1_BIT) == 0x00100000)       // PLAIRQ1
  {
    pulse = T1CAP;                  // read the capture event
    IRQCLR = 0x00100000;            // disable PLA IRQ1
    IRQEN = 0x80000;                // enable PLA IRQ0
    T1LD = 0x00;                    // to reset timer1
    T1CON = 0xC0;                   // reset timer1
    T1CON = 0;                      // stop timer1
  }
  return ;
}
```


AN05319–0–2/05(0)