

Revisiones	Fecha	Comentarios
0	13/8/03	

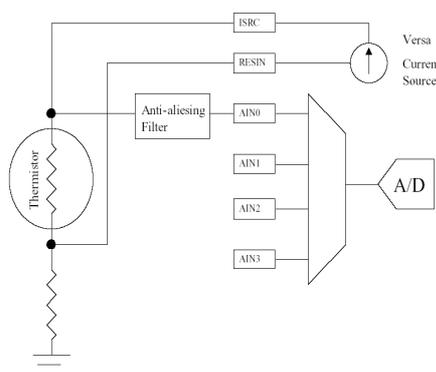
Desarrollamos una aplicación de filtrado de entradas analógicas aprovechando la MAC presente en los VERSA1 de Goal Semiconductor. La nota se basa en la medición de la señal proveniente de un termistor, vinculado al microcontrolador mediante un cable cuya longitud y tendido lo hacen propenso a captar ruidos. Para la polarización del termistor, utilizaremos la fuente de corriente controlada.

Descripción

En muchas aplicaciones, el ruido es un grave problema a considerar, especialmente si las señales son de poca amplitud o el sistema está ubicado en un ambiente de alto ruido electromagnético. Un ejemplo podría ser un sistema de medición de temperatura ambiente mediante un termistor, conectado al sistema por un par de metros de cable que pasa cerca de equipos de alta potencia. En este caso, es probable que el cable permita una gran inducción de la señal alterna de línea. La forma tradicional de resolver el problema es blindar los cables e intercalar filtros analógicos discretos. Esta última opción puede llegar a incrementar demasiado la superficie de placa necesaria, e introducir errores proporcionales a la tolerancia de los componentes; los cuales serán fijos y no permitirán adaptarse a cambios en la señal.

Otra alternativa, consiste en tomar muestras de la señal a una velocidad algo mayor, y filtrarla con alguna técnica conocida. En este caso, para ilustrar la aplicación de la MAC incluida en la serie VERSA1, emplearemos un filtro FIR. No daremos demasiados detalles sobre el cálculo y funcionamiento del filtro FIR, el lector puede consultar la extensa bibliografía sobre el tema en los textos de DSP si así lo desea.

Hardware



Conectamos el termistor como figura en el diagrama a la izquierda; el resistor entre RESIN y masa cumple la función de programar la corriente de la fuente controlada. La corriente constante a través del resistor producirá una tensión proporcional a la resistencia del mismo, la cual, a su vez, variará con la temperatura. El filtro antialias es un simple pasabajos con frecuencia de corte menor a la mitad de la frecuencia de muestreo.

A los fines del desarrollo de esta nota, dado que el centro de la misma es la implementación del filtro digital, se ha utilizado un hardware suplementario destinado a probar y documentar el funcionamiento del mismo.

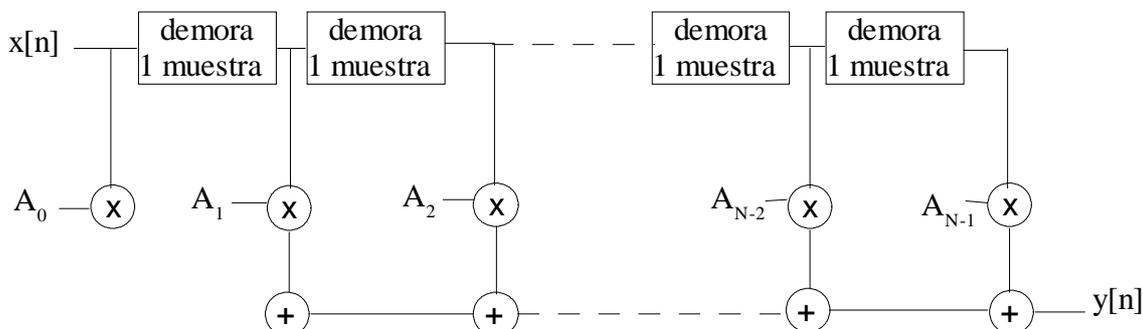
Software

La función transferencia en el plano Z que caracteriza un sistema FIR (Finite Impulse Response) es la siguiente: $H[Z] = \sum_{k=0}^{N-1} A_k \cdot Z^{-k}$, donde A_k son coeficientes y N es la cantidad de taps del filtro.

La ecuación diferencia que implementa esta transferencia es la siguiente: $y[n] = \sum_{k=0}^{N-1} A_k \cdot x[n-k]$, donde $y[n]$ es la secuencia de salida, $x[n]$ la secuencia de entrada, A_k son coeficientes, $x[n-k]$ corresponde al

valor de la secuencia $x[n]$ un instante k muestras anterior al momento actual, y N es la cantidad de taps del filtro, longitud de la respuesta al impulso en muestras.

Si bien existen varias formas de implementarlo, la forma más simple y explícita es la siguiente:



Como puede observarse, esto sugiere la necesidad de emplear gran cantidad de sumas y multiplicaciones con números del formato que la precisión de la señal requiera, y realizarlo a gran velocidad, dado que deben realizarse todas estas operaciones antes de que la próxima muestra esté disponible. Así mismo, vemos también que una MAC (Multiply and ACcumulate) resulta de suma utilidad para realizar esta tarea, dado que nos simplifica notoriamente la tarea de multiplicar (sólo debemos cargar un par de registros) y también la de acumular (no necesitamos propagar carries).

Table 1 – Impulse coefficients of the 16 taps Low-pass Filter

Filter Coef	Relative Value	Filter Coef	Relative Value
1	0.008743286	9	0.114517211
2	0.018020629	10	0.106201171
3	0.032867431	11	0.091094970
4	0.051467895	12	0.071823120
5	0.071823120	13	0.051467895
6	0.091094970	14	0.032867431
7	0.106201171	15	0.018020629
8	0.114517211	16	0.008743286

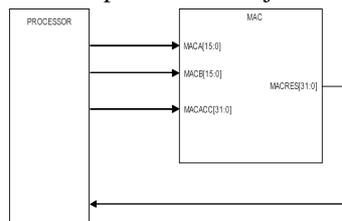
Del análisis del ruido en la entrada, supongamos que deducimos que debemos implementar un filtro pasabajos con una frecuencia de corte de 60Hz¹ y una atenuación fuera de la banda pasante de 40dB mínimo, con un ripple máximo tolerable de 0.01dB en banda pasante. Supongamos también que tomamos muestras a una frecuencia de 480Hz, es decir, 480 muestras por segundo; el cálculo de un filtro de estas características nos conduce a la realización de un filtro digital FIR de 16 taps con coeficientes positivos, como puede verse en la tabla a la izquierda.

Dado que se trata de todos números positivos, podemos utilizar el formato 0.16, es decir, números con coma fija, situada a la izquierda del MSb. Esto equivale a dividir los coeficientes por 2¹⁶, y genera la tabla que figura a la derecha, que cuantiza los coeficientes a 16 bits en formato 0.16. Esto nos permitirá, más adelante, simplificar notablemente los cálculos, dado que la multiplicación de dos números fraccionarios en formato 0.16 es un número fraccionario en formato 0.32, simplemente descartamos la palabra menos significativa y mantenemos la precisión de 16 bits. Si en cambio usáramos el formato 1.15, la multiplicación daría como resultado un número en formato 2.30 y sería necesario realizar desplazamientos (shifts) para conservar la precisión. Aquí es cuando resulta útil disponer de un barrel shifter, como es el caso del VERSA Mix.

Table 2 – FIR Filter coefficients

Filter Coef	Scaled coeffs Value (hex)	Filter Coef	Scaled coeffs Value (hex)
1	023Dh	9	1D51h
2	049Dh	10	1B30h
3	086Ah	11	1752h
4	0D2Dh	12	1263h
5	1263h	13	0D2Dh
6	1752h	14	086Ah
7	1B30h	15	049Dh
8	1D51h	16	023Dh

Veamos como podemos implementar una simple rutina para realizar el filtrado. Los coeficientes son copiados a RAM para poder utilizar los punteros (direccionamiento indirecto). En vez de utilizar un buffer circular, el autor ha preferido trabajar con posiciones fijas, realizando un desplazamiento de todas las muestras luego del cálculo. Esto no representa mayor carga dado que se trata de un buffer de 16 muestras.



La MAC se ve desde el procesador como un conjunto de SFRs, su ecuación de funcionamiento es:

$$MACRESULT = MACACC + (MACA \times MACB)$$

MACA y MACB son registros de 16 bits, MACACC y MACRESULT son de 32 bits. Todos están mapeados como múltiples SFR de 8 bits.

1 La nota de aplicación original fue realizada para 60Hz, frecuencia de red de América del Norte

CAN-009, Filtrado de entradas analógicas con VERSA1

Para mayor claridad mostramos sólo el núcleo del cálculo del FIR. El resto del código es algo más complejo, dado que el autor ha incluido código para observar el funcionamiento.

```
FIRCOMPUTE:    MOV     R0,#NPOINTSBASEADRS ;Dirección BASE de las muestras
               MOV     R1,#COEFBASEADRS ;Dirección BASE de los coeficientes
               MOV     R7,#NPOINTS      ;Número de taps

               CLR     A
               MOV     MACA1,A          ;Inicializa MACA = 0
               MOV     MACA0,A

               MOV     MACACC3,A        ;Inicializa MACACC = 0
               MOV     MACACC2,A
               MOV     MACACC1,A
               MOV     MACACC0,A        ;Combinados: MACRESULT = 00000000h

               ;** Calcula Y[n]

COMPUTELOOP:   MOV     MACACC3,MACRES3   ;Copia resultado previo en el acumulador
               MOV     MACACC2,MACRES2   ;MACACC
               MOV     MACACC1,MACRES1
               MOV     MACACC0,MACRES0

               MOV     MACA1,@R0         ; x[n-k]
               INC     R0
               MOV     MACA0,@R0
               INC     R0

               MOV     MACB1,@R1        ; Ak
               INC     R1
               MOV     MACB0,@R1
               INC     R1

               DJNZ    R7,COMPUTELOOP    ; Loop de N taps

               ;** Desplaza las muestras 1 lugar y deja lugar para la siguiente

SHIFTPAST:     MOV     R7,#(NPOINTS-1)*2 ;Número de desplazamientos: 2(N-1)
               MOV     R0,#(NPOINTSBASEADRS - 1 + 2*(NPOINTS-1))
               ;Fuente
               MOV     R1,#(NPOINTSBASEADRS + 1 + 2*(NPOINTS-1))
               ;Destino

SHIFTLLOOP:   MOV     A,@R0             ;Desplaza
               MOV     @R1,A
               DEC     R0
               DEC     R1
               DJNZ    R7,SHIFTLLOOP

               RET
```

El resultado de la última multiplicación está en MACRES (32-bits). Como dijéramos anteriormente, al ser números en formato 0.16, el resultado de la multiplicación está en formato 0.32. Tomamos como resultado de la señal filtrada la palabra más significativa, descartando los 16 “decimales” menos significativos, es decir, un número en 0.16 en MACRES3|MACRES2.

Como verificación de lo expuesto, en la hoja siguiente pueden apreciarse:

- La respuesta teórica esperada de este filtro (Figura 3)
- Oscilogramas de entrada y salida del filtro (Figura 4): se conectó un generador de funciones a la entrada y un conversor D/A SPI generó la tensión de salida.
- Respuesta en frecuencia obtenida con barredor (Figura 5)

El listado completo del código assembler está disponible para su consulta.

Figure 3 – Expected FIR Filter response

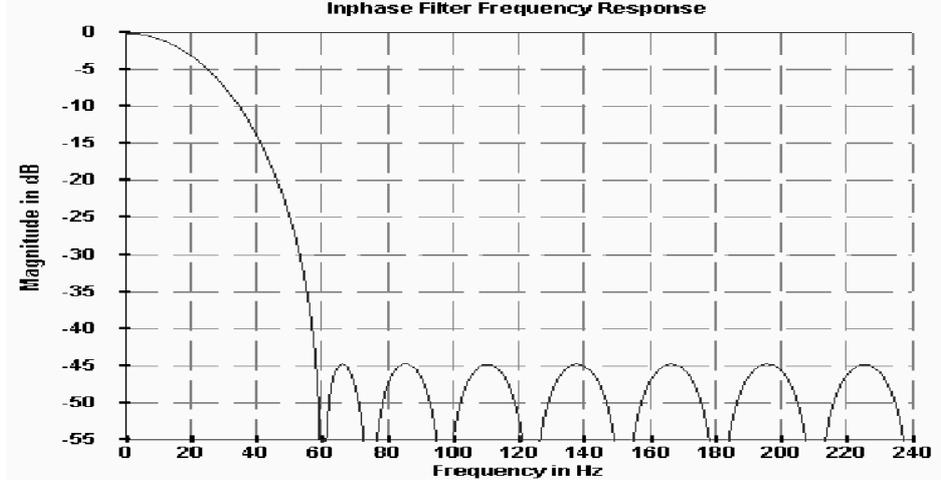


Figure –4 – Input / Output of the FIR filter at 60Hz

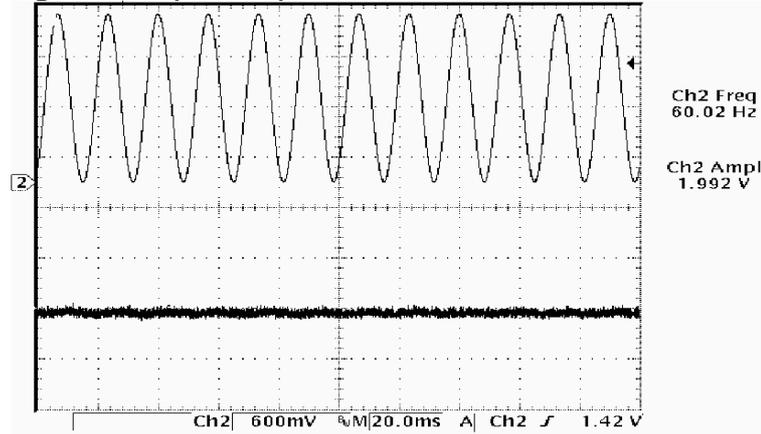


Figure 5 – Practical FIR Filter Response

