



---

## 50/60Hz Rejection 的解決方案

## 目錄

1. 簡介 .....	4
2. 原理說明 .....	5
2.1. 軟體SINC <sup>4</sup> 的原理.....	5
2.2. 軟體SINC <sup>3</sup> 的原理.....	6
3. 50/60HZ的NORMAL MODE REJECTION在軟體平均(SINC <sup>1</sup> )、軟體SINC <sup>3</sup> 、軟體SINC <sup>4</sup> 的差異 .....	7
3.1. SINC <sup>1</sup> 、SINC <sup>3</sup> 與SINC <sup>4</sup> 在不同輸入頻率下的Normal Mode Rejection比較表.....	7
3.2. 平均(SINC <sup>1</sup> )、SINC <sup>3</sup> 與SINC <sup>4</sup> 在不同輸入頻率下的FFT比較.....	7
4. SINC <sup>4</sup> 軟體撰寫.....	10
5. SINC <sup>3</sup> 軟體撰寫 .....	13
6. DEMO CODE .....	15

注意：

- 1、本說明書中的內容，隨著產品的改進，有可能不經過預告而更改。請客戶及時到本公司網站下載更新 <http://www.hycontek.com>
- 2、本規格書中的圖形、應用電路等，因第三方工業所有權引發的問題，本公司不承擔其責任。
- 3、本產品在單獨應用的情況下，本公司保證它的性能、典型應用和功能符合說明書中的條件。當使用在客戶的產品或設備中，以上條件我們不作保證，建議客戶做充分的評估和測試。
- 4、請注意輸入電壓、輸出電壓、負載電流的使用條件，使 IC 內的功耗不超過封裝的容許功耗。對於客戶在超出說明書中規定額定值使用產品，即使是瞬間的使用，由此所造成的損失，本公司不承擔任何責任。
- 5、本產品雖內置防靜電保護電路，但請不要施加超過保護電路性能的過大靜電。
- 6、本規格書中的產品，未經書面許可，不可使用在要求高可靠性的電路中。例如健康醫療器械、防災器械、車輛器械、車載器械及航空器械等對人體產生影響的器械或裝置，不得作為其部件使用。
- 7、本公司一直致力於提高產品的品質和可靠度，但所有的半導體產品都有一定的失效概率，這些失效概率可能會導致一些人身事故、火災事故等。當設計產品時，請充分留意冗餘設計，採用安全指標，這樣可以避免事故的發生。
- 8、本規格書中內容，未經本公司許可，嚴禁用於其他目的之轉載或複製。

## 1. 簡介

對於使用 AC 電源的高精度量測產品，由於其電源的 50/60 Hz 雜訊，等效會有 50/60 Hz 的雜訊加到訊號端被量測出來，如果測量系統的 50/60 Hz Normal Mode Rejection 不佳，將導致測量結果的不穩定。所以一般產品要解決這樣問題除了加強電源線路的濾波能力以外，使用軟體做數字濾波器，也是常用的方法。

HY11P 系列的 ADC 其內建的數位濾波器為 2 階的梳狀濾波器(SINC<sup>2</sup> COMB Filter)，以使用內部 RC 震盪器為其工作時脈及 OSR=32768 情形下(約 8 SPS output rate)其 50/60 Hz Normal Mode Rejection 可能只有 50~60 dB，即使再做 8 次軟體平均也只能到 80dB。相較於 TI 之 ADS1230 使用 4 階的梳狀濾波器(SINC<sup>4</sup> COMB Filter)，其在 10 SPS output rate 下可以做到 100dB 而言，HY11P 系列之 ADC 的 50/60 Hz Normal Mode Rejection 明顯有較大的差異。

本文提供軟體的數字濾波器方法，讓同樣使用 HY11P 系列的 ADC 可以做到 100 dB 50/60 Hz Normal Mode Rejection，解決客戶使用 AC 電源雜訊的問題。

## 2. 原理說明

下圖為 10 sps output rate 之 4 階梳狀濾波的頻譜，COMB Filter 對於 10、20、30、40、50、60Hz... 等的訊號會被濾除，從 50~60Hz 間其至少達 95dB 以上訊號衰減。但如果使用 2 階梳狀濾波雖然同樣可以將 50/60 Hz 濾除，但其 50~60Hz 間只有 50dB 的訊號衰減。

由於 HY11P 可以改變 ADC OSR 設定提高 ADC 的 output rate 所以我們可以在使用軟體做高階梳狀濾波器。來提升 ADC 的 normal mode Rejection。

以下我們將以軟體實現  $SINC^3$ / $SINC^4$  梳狀濾波器，並與原先硬件  $SINC^2$  比較其 50/60Hz normal mode Rejection 能力。我們在以相同約 1 s 輸出穩定時間來比較( $SINC^3$  需 3 筆資料穩定， $SINC^4$  需 4 筆資料穩定)，所以原硬件  $SINC^2$  OSR=32768 的輸出我們還做 8 筆資料平均( $SINC^1$ )。

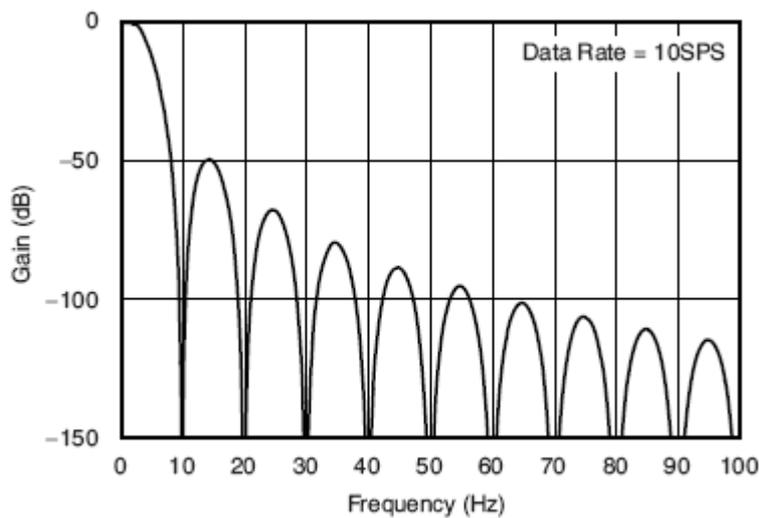


圖 2-1

### 2.1. 軟體 $SINC^4$ 的原理

$SINC^4$  是由 4 個加法器與 4 個減法器所組成，可視為 4 個積分器與 4 個微分器。

由於積分器需累加平均 32 次 ( $2^5$ ) 所以 1 個積分器需多出 5 個 bits Register，而 4 個積分器就須多出 20 bits Register，而 ADC 有 24 bit 輸出，因此每一個積分器就需要 44 bits 的 Register，運算 32 筆後取 Bit 20~43 輸出作微分處理，而每一個微分器只需 24bits Register，因此做一個軟體 4 階的 COMB Filter 最少需要 272 bits(34 bytes)的 Register。

由於 4 階的 COMB Filter 經由 4 次運算最後得出的結果，因此輸入訊號延遲 4 筆輸出，因此如果 ADC Output Rate 為 122Hz，經 32 筆平均後約為 3.8Hz，但訊號延遲 4 筆，所以訊號輸入到穩定輸出需 1.05 秒。

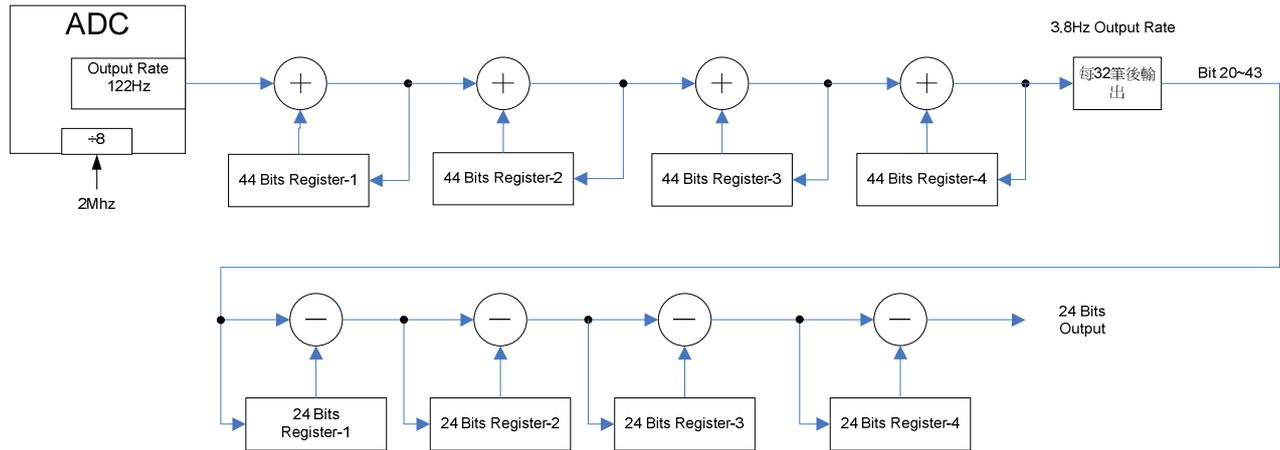


圖 2-2 軟體 4 階 COMB Filter 示意圖

## 2.2. 軟體 SINC<sup>3</sup> 的原理

SINC<sup>3</sup> 是由 3 個加法器與 3 個減法器所組成，可視為 3 個積分器與 3 個微分器。

由於積分器需累加平均 32 次 ( $2^5$ ) 所以 1 個積分器需多出 5 個 bits Register，而 3 個積分器就須多出 15 bits Register，而 ADC 有 24 bit 輸出，因此每一個積分器就需要 39 bits 的 Register，運算 32 筆後取 Bit 15~38 輸出作微分處理，而每一個微分器只需 24bits Register，因此做一個軟體 3 階的 COMB Filter 最少需要 189 bits(23.6 bytes)的 Register。

由於 3 階的 COMB Filter 經由 3 次運算最後得出的結果，因此輸入訊號延遲 3 筆輸出，因此如果 ADC Output Rate 為 122Hz，經 32 筆平均後為 3.8Hz，但訊號延遲 3 筆，所以訊號輸入到穩定輸出需 0.79 秒。

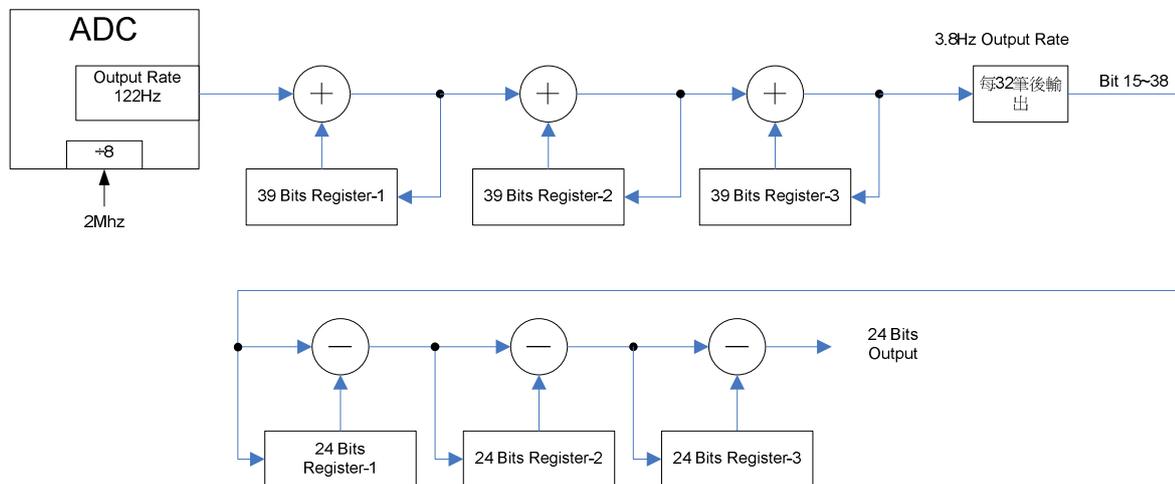


圖 2-3 軟體 3 階 COMB Filter 示意圖

### 3. 50/60Hz 的 Normal Mode Rejection 在軟體平均(SINC<sup>1</sup>)、軟體 SINC<sup>3</sup>、軟體 SINC<sup>4</sup> 的差異

由於考慮內部 HY11P 系列內部 RC 震盪頻率飄移。所以我們測試頻率從 40~70Hz 比較不同方法的 normal mode Rejection 能力。

#### 3.1. SINC<sup>1</sup>、SINC<sup>3</sup> 與 SINC<sup>4</sup> 在不同輸入頻率下的 Normal Mode Rejection 比較表

PGA Gain = 1 , ADC Gain = 8

VDD=3.3V , VDDA = 2.93V

Input Signal = 50mV<sub>RMS</sub> @ 40~70Hz , DC offset 70.7mV

表格 3-1

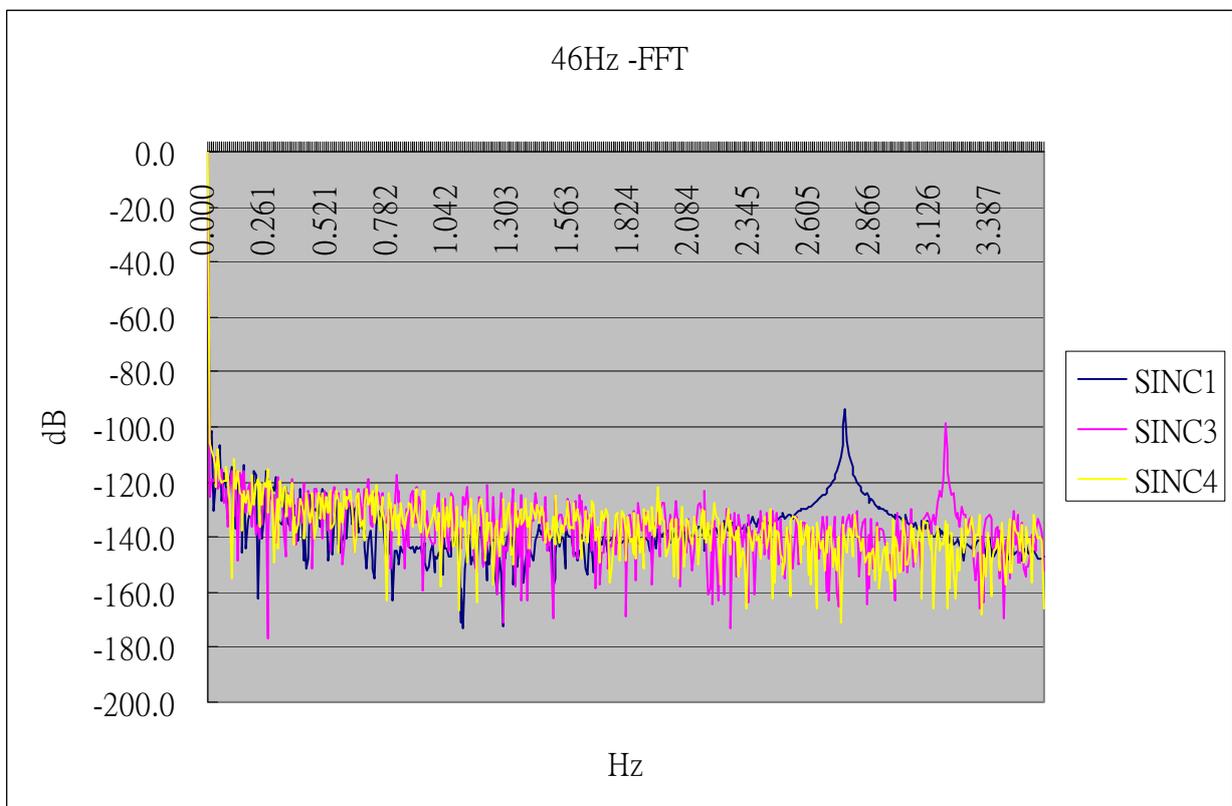
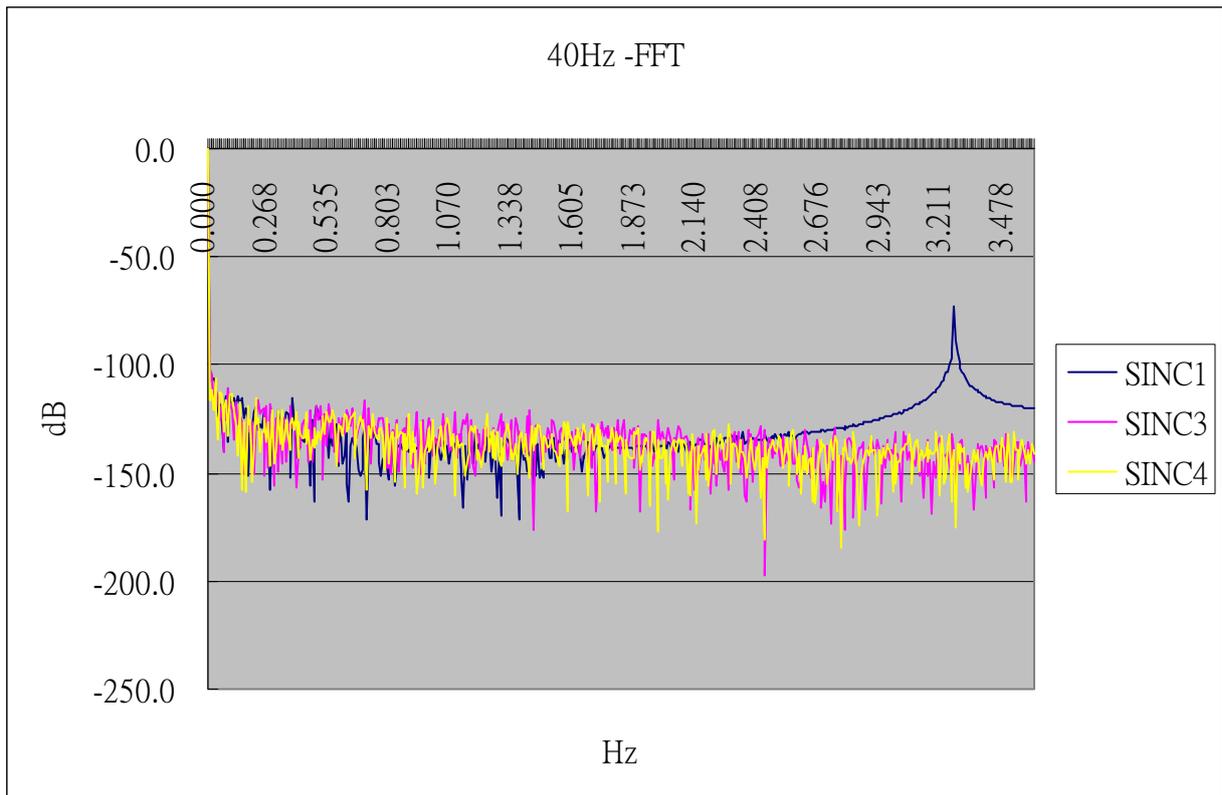
Hz \ dB	40	41	42	44	45	46	47	48	49	50	51	52	54	55
SINC <sup>1</sup>	-72.55	-84.87	-78.7	-97.54	-99.76	-93.41	-82.46	-80.55	-85.85	-95.4	-95.02	-85.37	-89.41	-113.4
SINC <sup>3</sup>	-115.8	-109.3	-99.22	-113.5	-106.8	-98.9	-113.8	-115.1	-118	-113.8	-116.5	-115.3	-108.1	-118
SINC <sup>4</sup>	-120.5	-117.1	-117.6	-118.9	-117.3	-114.9	-117.4	-118.7	-118.2	-117.3	-116.4	-120.3	-116.6	-118.4

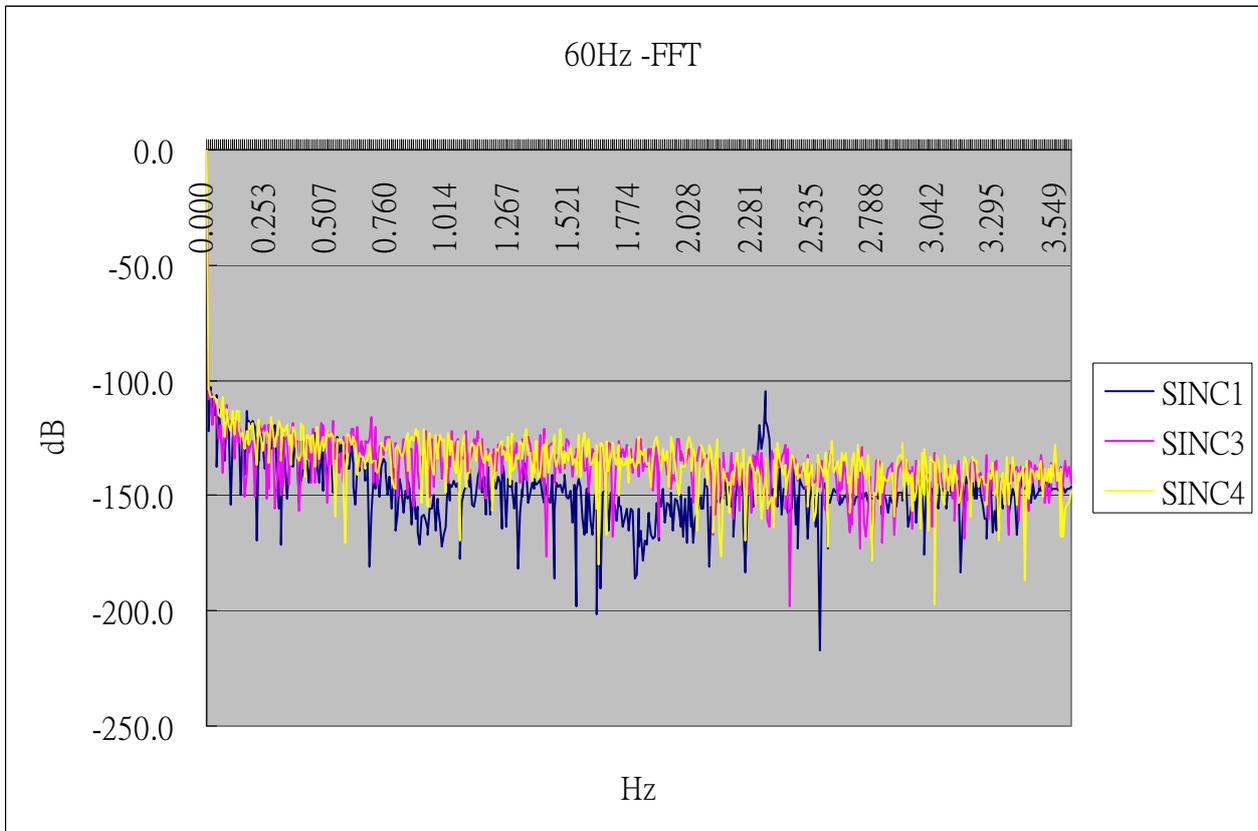
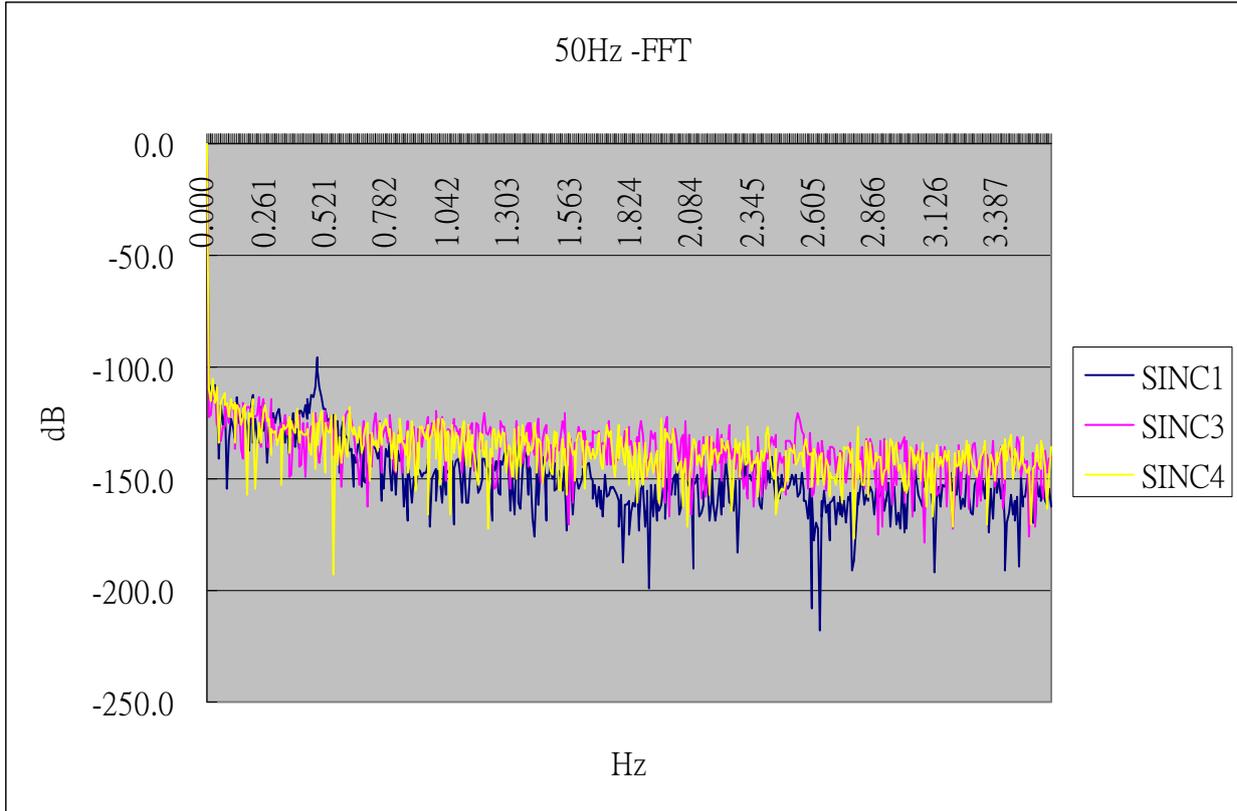
Hz \ dB	56	57	58	59	60	61	62	63	64	66	68	69	70
SINC <sup>1</sup>	-106.7	-97.01	-95.64	-85.44	-105	-84.92	-85.06	-114.8	-115	-100.4	-96.17	-82.87	-102
SINC <sup>3</sup>	-111.1	-108.5	-117.1	-116.5	-105.1	-109.8	-113.8	-114.3	-105	-117.1	-115.9	-116.5	-118.3
SINC <sup>4</sup>	-118.1	-118.8	-116.4	-116	-115.7	-115.8	-117.8	-117.5	-114.4	-115.8	-115.9	-117.9	-117.9

表格 3-1，在輸入 40~70Hz 的 Common mode 訊號下比較平均(SINC<sup>1</sup>)、軟體 SINC<sup>3</sup>、軟體 SINC<sup>4</sup> 的 Rejection 效果；SINC<sup>3</sup> 最少有 98dB 以上並且在一些輸入頻率上會有突刺點，在 SINC<sup>4</sup> 就完全看不到突刺點而且最少在 114dB 以上，而 SINC<sup>1</sup> 的 Rejection 效果只能達到 72dB。

#### 3.2. 平均(SINC<sup>1</sup>)、SINC<sup>3</sup> 與 SINC<sup>4</sup> 在不同輸入頻率下的 FFT 比較

雖然我們直接可以在 Time domain 下看到單純使用平均的方式，對於不同頻率的輸入訊號會有不同的滾動程度，而 SINC<sup>3</sup> 與 SINC<sup>4</sup> 的方式幾乎不會有任何的滾動。但實際更細微的差異我們取 1024 筆資料做 FFT，可以在頻譜上觀察到量化的結果。下面幾個圖表是在不同頻率下的 FFT 的結果。我們可以看到 SINC<sup>4</sup> 的方法在 40~70 Hz 間的訊號幾乎不會有任何影響，而使用 SINC<sup>3</sup> 的方法在某些頻率下(如 46Hz)，還是無法完全濾掉，至於單純平均的方法效果最差。





## 4. SINC<sup>4</sup> 軟體撰寫

```

=====
; ADC output 約 122 Hz
; 輸出 32 筆做 SINC4 平均 約 3.8hz 輸出
; 積分器 bit 0 ~ bit 43
; 微分器 bit 20 ~ bit 43
; IZAL、IZAM、IZAH、IZAU、IZAY、IZAX = 積分暫存器 1
; IZBL、IZBM、IZBH、IZBU、IBAY、IBAX = 積分暫存器 2
; IZCL、IZCM、IZCH、IZCU、ICAY、ICAX = 積分暫存器 3
; IZDL、IZDM、IZDH、IZDU、IDAY、IDAX = 積分暫存器 4
; DZAL、DZAM、DZAH' 微分暫存器 1
; DZBL、DZBM、DZBH' 微分暫存器 2
; DZCL、DZCM、DZCH' 微分暫存器 3
; DZDL、DZDM、DZDH' 微分暫存器 4
=====
    btss    INTF1,ADCIF,ACCE
    jmp     Inter_3_out
    bcf     INTF1,ADCIF,ACCE

    lbsr    IZAL
    mvf    ADC0RL,W,ACCE      ;積分器 1 + ADC --> 積分器 1
    addf   IZAL,F,BANK
    mvf    ADC0RM,W,ACCE
    addc   IZAM,F,BANK
    mvf    ADC0RH,W,ACCE
    addc   IZAH,F,BANK
    mvl    0
    btsz   ADC0RH,7,ACCE
    mvl    0FFh
    addc   IZAU,F,BANK
    addc   IZAY,F,BANK
    addc   IZAX,F,BANK

    mvf    IZAL,W,BANK      ;積分器 1 + 積分器 2 --> 積分器 2
    addf   IZBL,F,BANK
    mvf    IZAM,W,BANK
    addc   IZBM,F,BANK
    mvf    IZAH,W,BANK
    addc   IZBH,F,BANK
    mvf    IZAU,W,BANK
    addc   IZBU,F,BANK
    mvf    IZAY,W,BANK
    addc   IZBY,F,BANK
    mvf    IZAX,W,BANK
    addc   IZBX,F,BANK

    mvf    IZBL,W,BANK      ;積分器 2 + 積分器 3 --> 積分器 3
    addf   IZCL,F,BANK
    mvf    IZBM,W,BANK
    addc   IZCM,F,BANK
    mvf    IZBH,W,BANK
    addc   IZCH,F,BANK
    mvf    IZBU,W,BANK
    addc   IZCU,F,BANK
    mvf    IZBY,W,BANK
    addc   IZCY,F,BANK
    mvf    IZBX,W,BANK
    addc   IZCX,F,BANK

    mvf    IZCL,W,BANK      ;積分器 3 + 積分器 4 --> 積分器 4
    addf   IZDL,F,BANK
    mvf    IZCM,W,BANK
    addc   IZDM,F,BANK
    mvf    IZCH,W,BANK
    addc   IZDH,F,BANK
    mvf    IZCU,W,BANK
    addc   IZDU,F,BANK
    mvf    IZCY,W,BANK
    addc   IZDY,F,BANK

```

```

mvf    IZCX,W,BANK
addc   IZDX,F,BANK

inf    ADCCnter,W,ACCE
mvl    31                ; 累加 32 次
cpsl   ADCCnter,ACCE
jmp    INTER_DVIN
inf    ADCCnter,F,ACCE
jmp    Inter_3_out
;=====
INTER_DVIN:
mvl    0C0h
andf   ADCCnter,F,ACCE
inf    WECNT,F,ACCE
INTER_DVOU:
mvff   IZDL,BIOUL       ;积分器 4 --> Buffer
mvff   IZDM,BIOUM
mvff   IZDH,BIOUH
mvff   IZDU,BIOUU
mvff   IZDY,BIOUY
mvff   IZDX,BIOUX

bsf    ADCFg,b_ADCint,ACCE
bsf    Flag,b_ADCPCOK,ACCE
lbsr   DZAL
mvl    4                ;取 Buffer 的 Bit 20 ~ 44
LINTER_DVOULOOP:
bcf    Status,C,ACCE
rrfc   BIOUX,F,BANK
rrfc   BIOUY,F,BANK
rrfc   BIOUU,F,BANK
rrfc   BIOUH,F,BANK
rrfc   BIOUM,F,BANK
rrfc   BIOUL,F,BANK
dcsz   WREG,F,ACCE
jmp    LINTER_DVOULOOP

mvf    DZAL,W,BANK      ;Buffer - 微分器 1 --> BufferA
subf   BIOUH,W,BANK
mvf    BADCOL,F,BANK
mvf    DZAM,W,BANK
subc   BIOUU,W,BANK
mvf    BADCOM,F,BANK
mvf    DZAH,W,BANK
subc   BIOUY,W,BANK
mvf    BADCOH,F,BANK
mvff   BIOUH,DZAL      ; Buffer --> 微分器 1
mvff   BIOUU,DZAM
mvff   BIOUY,DZAH

mvf    DZBL,W,BANK     ; BufferA - 微分器 2 --> BufferB
subf   BADCOL,W,BANK
mvf    BBBADNEWL,F,BANK
mvf    DZBM,W,BANK
subc   BADCOM,W,BANK
mvf    BBBADNEWM,F,BANK
mvf    DZBH,W,BANK
subc   BADCOH,W,BANK
mvf    BBBADNEWH,F,BANK
mvff   BADCOL,DZBL     ; BufferA --> 微分器 2
mvff   BADCOM,DZBM
mvff   BADCOH,DZBH

mvf    DZCL,W,BANK     ; BufferB - 微分器 2 --> BufferA
subf   BBBADNEWL,W,BANK
mvf    BADCOL,F,BANK
mvf    DZCM,W,BANK
subc   BBBADNEWM,W,BANK
mvf    BADCOM,F,BANK
mvf    DZCH,W,BANK
subc   BBBADNEWH,W,BANK

```

# 50/60Hz Rejection 的解決方案

```
mvf   BADCOH,F,BANK
mvff  BBBADNEWL,DZCL      ; BufferB --> 微分器 3
mvff  BBBADNEWM,DZCM
mvff  BBBADNEWH,DZCH
```

```
mvf   DZDL,W,BANK        ; BufferA - 微分器 4 --> BufferB
subf  BADCOL,W,BANK
mvf   BBBADNEWL,F,BANK
mvf   DZDM,W,BANK
subc  BADCOM,W,BANK
mvf   BBBADNEWM,F,BANK
mvf   DZDH,W,BANK
subc  BADCOH,W,BANK
mvf   BBBADNEWH,F,BANK
mvff  BADCOL,DZDL        ; BufferA --> 微分器 4
mvff  BADCOM,DZDM
mvff  BADCOH,DZDH
```

```
mvff  BBBADNEWH,BADNEWH ; 最後 BufferA 輸出到 ADC Output
mvff  BBBADNEWM,BADNEWM
mvff  BBBADNEWL,BADNEWL
```

```
;=====
Inter_3_out:
```

## 5. SINC<sup>3</sup> 軟體撰寫

```

;=====
; ADC output 約 122 Hz
; 輸出 32 筆做 SINC3平均 約 3.8hz 輸出
; 積分器 bit 0 ~ bit 38
; 微分器 bit 15 ~ bit 38
; IZAL、IZAM、IZAH、IZAU、IZAY → 積分暫存器 1
; IZBL、IZBM、IZBH、IZBU、IBAY → 積分暫存器 2
; IZCL、IZCM、IZCH、IZCU、ICAY → 積分暫存器 3
; DZAL、DZAM、DZAH → 微分暫存器 1
; DZBL、DZBM、DZBH → 微分暫存器 2
; DZCL、DZCM、DZCH → 微分暫存器 3
;=====
        btss    INTF1,ADCIF,ACCE
        jmp     Inter_3_out
        bcf     INTF1,ADCIF,ACCE

        lbrs   IZAL
        mvf    ADC0RL,W,ACCE      ;積分器 1 + ADC --> 積分器 1
        addf   IZAL,F,BANK
        mvf    ADC0RM,W,ACCE
        addc   IZAM,F,BANK
        mvf    ADC0RH,W,ACCE
        addc   IZAH,F,BANK
        mvl    0
        btsz   ADC0RH,7,ACCE
        mvl    0FFh
        addc   IZAU,F,BANK
        addc   IZAY,F,BANK

        mvf    IZAL,W,BANK        ;積分器 1 + 積分器 2 --> 積分器 2
        addf   IZBL,F,BANK
        mvf    IZAM,W,BANK
        addc   IZBM,F,BANK
        mvf    IZAH,W,BANK
        addc   IZBH,F,BANK
        mvf    IZAU,W,BANK
        addc   IZBU,F,BANK
        mvf    IZAY,W,BANK
        addc   IZBY,F,BANK

        mvf    IZBL,W,BANK        ;積分器 2 + 積分器 3 --> 積分器 3
        addf   IZCL,F,BANK
        mvf    IZBM,W,BANK
        addc   IZCM,F,BANK
        mvf    IZBH,W,BANK
        addc   IZCH,F,BANK
        mvf    IZBU,W,BANK
        addc   IZCU,F,BANK
        mvf    IZBY,W,BANK
        addc   IZCY,F,BANK

        inf    ADCCnter,W,ACCE    ; 累加 32 次
        mvl    31
        cpsl   ADCCnter,ACCE
        jmp    INTER_DVIN
        inf    ADCCnter,F,ACCE
        jmp    Inter_3_out
;=====
INTER_DVIN:
        mvl    0C0h
        andf   ADCCnter,F,ACCE
INTER_DVOU:
        mvff   IZCL,BIOUL        ;積分器 3 --> Buffer
        mvff   IZCM,BIOUM
        mvff   IZCH,BIOUH
        mvff   IZCU,BIOUU
        mvff   IZCY,BIOUY
    
```

```

bsf   ADCFg,b_ADCint,ACCE
bsf   Flag,b_ADCPCOK,ACCE
lbsr  DZAL
mvl   7                               ;取 Buffer 的 Bit 15 ~ 38
LINTER_DVOULOOP:
bcf   Status,C,ACCE
rrfc  BIOUY,F,BANK
rrfc  BIOUU,F,BANK
rrfc  BIOUH,F,BANK
rrfc  BIOUM,F,BANK
rrfc  BIOUL,F,BANK
dcsz  WREG,F,ACCE
jmp   LINTER_DVOULOOP

mfv   DZAL,W,BANK                     ; Buffer - 微分器 1 --> BufferA
subf  BIOUM,W,BANK
mfv   BADCOL,F,BANK
mfv   DZAM,W,BANK
subc  BIOUH,W,BANK
mfv   BADCOM,F,BANK
mfv   DZAH,W,BANK
subc  BIOUU,W,BANK
mfv   BADCOH,F,BANK
mvff  BIOUM,DZAL                       ; Buffer --> 微分器 1
mvff  BIOUH,DZAM
mvff  BIOUU,DZAH

mfv   DZBL,W,BANK                     ; BufferA - 微分器 2 --> BufferB
subf  BADCOL,W,BANK
mfv   BBBADNEWL,F,BANK
mfv   DZBM,W,BANK
subc  BADCOM,W,BANK
mfv   BBBADNEWM,F,BANK
mfv   DZBH,W,BANK
subc  BADCOH,W,BANK
mfv   BBBADNEWH,F,BANK
mvff  BADCOL,DZB                       ; BufferA --> 微分器 2
mvff  BADCOM,DZBM
mvff  BADCOH,DZBH

mfv   DZCL,W,BANK                     ; BufferB - 微分器 2 --> BufferA
subf  BBBADNEWL,W,BANK
mfv   BADCOL,F,BANK
mfv   DZCM,W,BANK
subc  BBBADNEWM,W,BANK
mfv   BADCOM,F,BANK
mfv   DZCH,W,BANK
subc  BBBADNEWH,W,BANK
mfv   BADCOH,F,BANK
mvff  BBBADNEWL,DZCL                   ; BufferB --> 微分器 3
mvff  BBBADNEWM,DZCM
mvff  BBBADNEWH,DZCH

mvff  BADCOH,BADNEWH                   ; 最後 BufferB 輸出到 ADC Output
mvff  BADCOM,BADNEWM
mvff  BADCOL,BADNEWL
Inter_3_out:

```

## 6. Demo Code

晶片選擇: HY11P14  
功能選擇: 僅提供內碼顯示功能



HY11P14-SINC3.rar



HY11P14-SINC4.rar