

# Implementing a 10-Bit Sigma-Delta Analog-to-Digital Converter Using the HC9S08Rx MCU Family Analog Comparator

By: Jefferson Daniel de Barros Soldera  
Marcus Espindola  
Alfredo Olmos  
Brazilian Semiconductor Technology Center — BSTC/SPS

---

## Introduction

Analog-to-digital (AD) converters, based on the Sigma-Delta ( $\Sigma\Delta$ ) modulation trade resolution in time for resolution in amplitude by combining:

- High sampling rates
- Negative feedback
- Digital filtering

First order  $\Sigma\Delta$  AD comparators are the simplest, most robust, and have a stable modulator architecture. They are especially insensitive to circuit imperfections and component mismatch since they rely on a simple integrator and a comparator embedded within a feedback loop; whereas, the digital filter can be easily implemented with a comb filter.

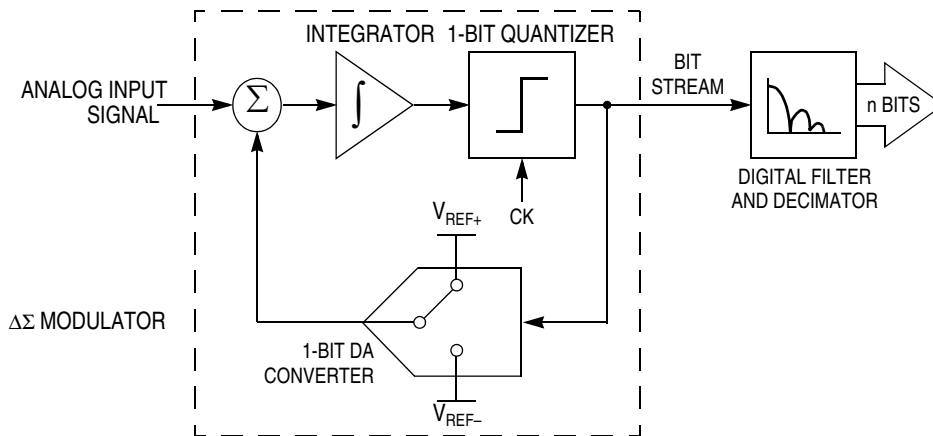
This application note describes how to implement a 10-bit first-order continuous-time  $\Sigma\Delta$  AD converter using a member of the HC9S08Rx MCU Family of devices including the analog comparator feature. In spite of this MCU family not having a built-in AD converter the analog comparator, internal bandgap reference voltage and timers can be used to convert an analog input signal to the digital domain with 10-bits of resolution. Some members of the HC9S08Rx MCU Family have a rail-to-rail comparator module and their internal bandgap reference can be selected as the comparator reference voltage. The HC9S08Rx analog comparator allows developing a very low-cost first-order continuous-time  $\Sigma\Delta$  modulator by adding a few external components; whereas, the MCU itself emulates the digital decimeter filter in software using its internal timers. High resolution can be attained by properly adjusting the input bandwidth and the oversampling ratio.

**$\Sigma\Delta$  Modulation at a Glance**

Figure 1 illustrates the basic architecture of a  $\Sigma\Delta$  AD converter comprising:

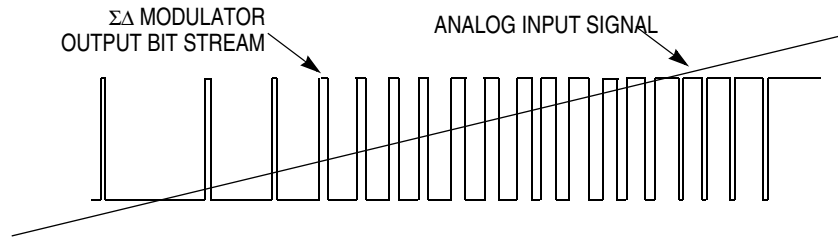
- An analog integrator
- A quantizer or comparator enclosed in a feedback loop via a single-bit digital-to-analog (DA) converter
- A digital low-pass filter.

The input signal is fed to the comparator through the integrator and the quantized output is fed back and subtracted from the input. This feedback forces the average value of the quantized signal to track the average input. Any difference between them accumulates in the integrator and eventually corrects itself. Jointly with the integrator, the feedback loop attenuates the quantization noise at low frequencies while shaping the noise power to the high frequency range. Since the signal is sampled at a frequency greater than the Nyquist rate, high frequency noise can be removed without affecting the signal band by means of a digital low-pass filter operating at the output of the  $\Sigma\Delta$  modulator.



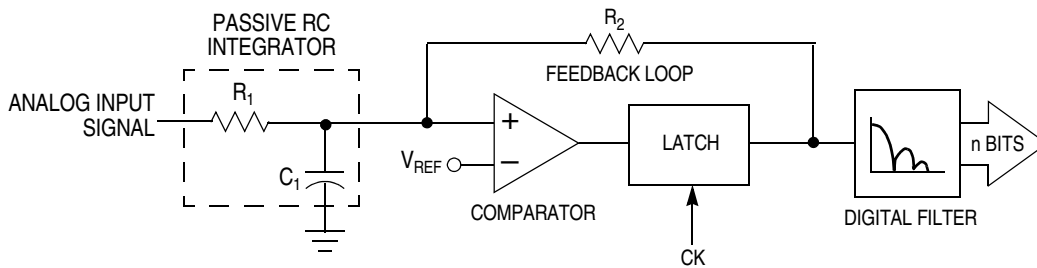
**Figure 1.  $\Sigma\Delta$  AD Converter**

Figure 2 exhibits the response of the modulator to a ramp input signal. Note the quantized output oscillates in such a manner that its local average equals the average input. The digital data coming from the modulator is a single bit stream of 1's and 0's and the density of digital 1's is proportional to the average value of the input signal. The high frequency data from the modulator is then low-pass filtered in order to remove any shaped noise and to decimate to the Nyquist rate. This decimation results in an n-bits binary output format.



**Figure 2. Response of the  $\Sigma\Delta$  Modulator to a Ramp Input Signal**

$\Sigma\Delta$  modulators have been widely implemented in the discrete-time domain using switched-capacitor circuits. However, continuous-time  $\Sigma\Delta$  modulators have become very popular since their performance is similar to the discrete-time version. Figure 3 shows a first order continuous-time  $\Sigma\Delta$  AD converter where the  $R_1$ - $C_1$  network acts as a passive integrator; whereas, the resistor  $R_2$  works as a 1-bit DA converter closing the feedback loop. The  $R_1$ - $C_1$  circuit also performs low-pass anti-alias filtering on the AD converter input.



**Figure 3. First-Order Continuous-Time  $\Sigma\Delta$  AD Converter**

Defining the oversampling ratio as:

$$M = f_{CK}/f_N \tag{Equation 1}$$

where:

- $f_{CK}$  is the clock sampling frequency
- $f_N$  is the Nyquist rate for a signal bandwidth of  $f_{in}$  ( $f_N = 2 f_{in}$ )

The signal-to-noise ratio (SNR) of the first order  $\Sigma\Delta$  modulator is given by:

$$SNR = 30 \log M - 3.41 \text{ dB} \tag{Equation 2}$$

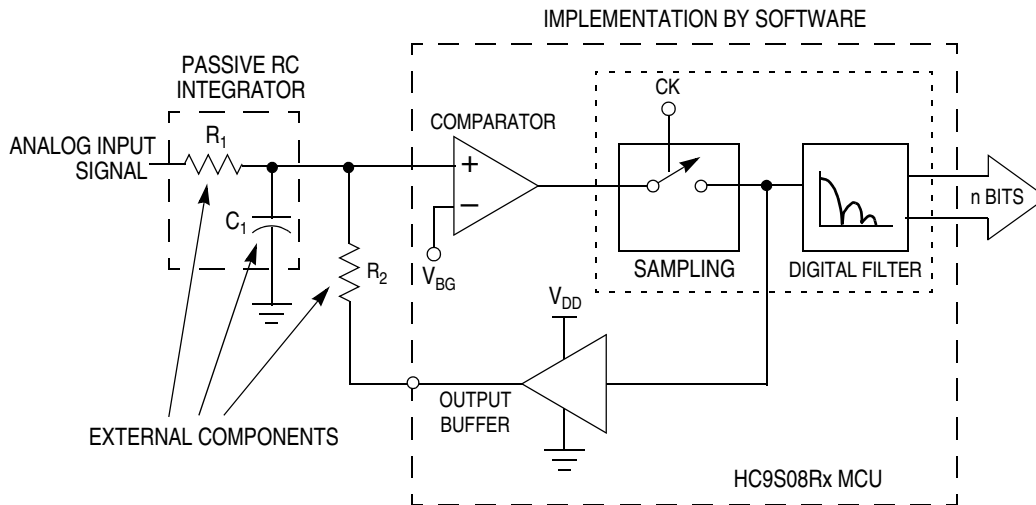
From equation 2, providing the oversampling ratio is larger than approximately 130, the circuit in Figure 3 allows attaining a 10-bit resolution. Since for DC or slowly varying input signals the noise power has large peaks degrading the  $\Sigma\Delta$  modulator SNR, a high frequency dither signal uncorrelated with the input should be injected. It has the effect of redistributing the energy present in the noise peaks over the entire amplitude range maintaining the expected resolution given by equation 2.

**Implementing a 10-Bit Sigma-Delta Analog-to-Digital Converter Using the HC9S08Rx MCU Family Analog Comparator, Rev. 0.1**

## Implementation with HC9S08Rx MCU Family

The HC9S08Rx MCU with the analog comparator function can be configured as a first-order continuous-time  $\Sigma\Delta$  AD converter as depicted in Figure 4. This is done by adding:

- Two external resistors
- One capacitor
- Properly programming the internal timers to filter and decimate the comparator digitized output.



**Figure 4. First-Order Continuous-Time  $\Sigma\Delta$  AD Converter Implementation with The HC9S08Rx MCU Including the Analog Comparator Feature**

The  $R_1$ - $C_1$  integrator is similar to the  $\Sigma\Delta$  modulator in Figure 3. The 1-bit DA conversion is achieved by driving the resistor  $R_2$  with an output buffer. The resistors must be adjusted to allow full range operation avoiding the integrator saturation. A suitable capacitor value must be adjusted regarding the sampling frequency and the input signal bandwidth. Once again, the  $R_1$ - $C_1$  network provides low-pass filtering with the option to include an additional anti-alias analog filter.

One internal HC9S08Rx MCU timer accumulates the number of 1's (or 0's) at the output of the comparator to compute the average of the analog input signal. A second timer controls the number of accumulator counts. Each accumulator increment is carefully controlled by software and it defines the  $\Sigma\Delta$  modulator sampling frequency and then the oversampling ratio  $M$ . To achieve a 10-bit output word length, the accumulator must sample 1024 times the input signal. Therefore, the conversion time increases when a more accurate solution is required.

---

## First Order Continuous-Time $\Sigma\Delta$ AD Converter Characteristics

$\Sigma\Delta$  AD converter characteristics include:

- Low cost implementation with few external components
- Low complexity especially in the analog domain since the HC9S08Rx MCU itself provides the digital processing capacity needed
- Small amount of memory required
- Input range controlled by external resistors
- Low matching level in between external components
- Performance immunity face-to-comparator hysteresis
- Modulator highly robust and inherently stable
- Accurate internal bandgap reference voltage
- Converter inherently linear and inherently monotonic
- Very simple anti-alias low-pass filter feasible to embed with the RC passive integrator
- Sample and hold, and trimming circuitry not required
- Architecture not particularly appropriate for DC inputs but for busy signals
- DC input applications require radiometric measurements and dithering
- 61.2 dB of SNR equivalent to 10-bits resolution
- 65.7 dB of dynamic range (DR) at 2.4 V of power supply

---

## HC9S08Rx Features and Benefits

Standard features of the HCS08 Family

- Extended battery life technology
  - Multiple power management modes including 20 nA power down @ 2 V
  - Optional auto wake-up with internal timer and internal oscillator typically 700 nA @ 2 V
  - 1.8–3.6 V operation
- High-performance when needed
  - 125 ns minimum instruction cycle time down to 1.8 V @ 8 MHz bus
- Innovative on-chip trigger/trace debug capability
  - Single wire background debug mode
  - Trace capability with 9 trigger modes and 3 breakpoints

Features of the MC9S08RC/RD/RE/RG MCU Family

- 8 MHz internal bus frequency
- On-chip in-circuit programmable FLASH memory with block protection and security option
- On-chip random-access memory (RAM)
- Low power oscillator capable of operating from crystal or resonator from 1 to 16 MHz

## Devices in the MC9S08RC/RD/RE/RG Series

- On-chip analog comparator with internal reference (ACMP)
  - Full rail-to-rail supply operation
  - Option to compare to a fixed internal bandgap reference voltage
- Serial communications interface module (SCI)
- Serial peripheral interface module (SPI)
- 2-channel, 16-bit timer/pulse width modulator (TPM) module with selectable input capture, output compare, and edge-aligned or center-aligned PWM capability on each channel.
- Keyboard interrupt ports (KBI1 and KBI2) providing 12 keyboard interrupts
  - Eight with falling-edge/low-level plus four with selectable polarity
- Carrier modulator timer (CMT) with dedicated infrared output (IRO) pin
  - Drives IRO pin for remote control communications
  - Can be disconnected from IRO pin and used as output compare timer
  - IRO output pin has high-current sink capability
- Eight high-current pins (limited by maximum package dissipation)
- Software selectable pull-ups on ports when used as input. Selection is on an individual port bit basis.
- During output mode, pull-ups are disengaged.
- 39 general-purpose input/output (I/O) pins, depending on package selection
- Four packages available:
  - 28-pin plastic dual in-line package (PDIP)
  - 28-pin small outline integrated circuit (SOIC)
  - 32-pin low-profile quad flat package (LQFP)
  - 44-pin low-profile quad flat package (LQFP)

---

## Devices in the MC9S08RC/RD/RE/RG Series

Table 1 lists all the devices available in the MC9S08RC/RD/RE/RG series and summarizes the differences in available functions and configuration between them.

**Table 1. MC9S08RC/RD/RE/RG Series Devices**

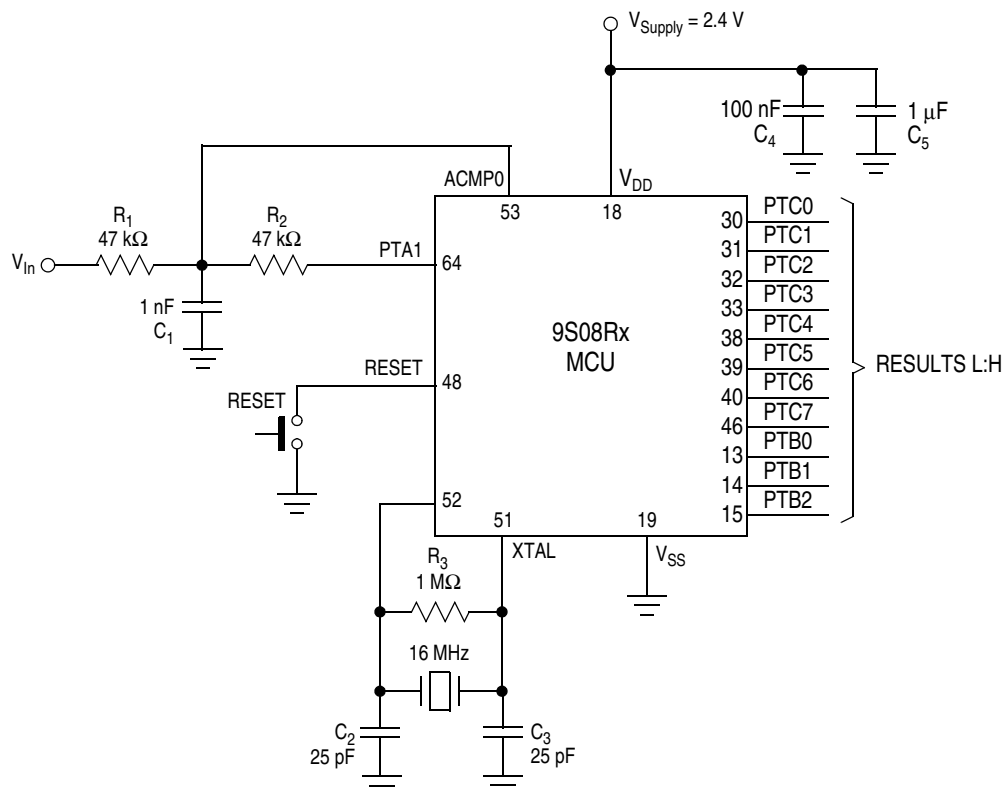
Device	FLASH	RAM <sup>(1)</sup>	ACMP <sup>(2)</sup>	SCI	SPI
9S08RG32/60	32K/60K	2K/2K	Yes	Yes	Yes
9S08RE8/16/32/60	8K/16K/32K/60K	1K/1K/2K/2K	Yes	Yes	No
9S08RD8/16/32/60	8K/16K/32K/60K	1K/1K/2K/2K	No	Yes	No
9S08RC8/16/32/60	8K/16K/32K/60K	1K/1K/2K/2K	Yes	No	No

**NOTES:**

1. Three S08RC/RD/RE16 ROM MCU devices have 512 bytes RAM instead of 1K bytes.
2. Only available in 32- or 44-pin LQFP packages.

## Application Description

Figure 5 shows the schematic diagram of the 10-bit  $\Sigma\Delta$  AD converter using a MC9S08Rx MCU including the analog comparator feature. Assuming the system employs two NiCd batteries of 1.2 V and the full input signal range is to be 2.4 V,  $R_1$  and  $R_2$  must be set to 47 k $\Omega$  to prevent integrator saturation. Since these resistors are not critical, carbon composition resistors or metal film resistors with 5% tolerance can be used.  $C_1$  is a ceramic capacitor. Its value depends on the modulator sampling frequency and oversampling ratio. The modulator sampling frequency is the MCU bus clock frequency divided by the number of cycles required to run the software instructions. As mentioned in [Software Description](#), the modulator sampling frequency can be set to 50 kHz adjusting properly the timer interruption as a function of the 8-MHz bus clock. Therefore, for an oversampling ratio of 130, the input signal bandwidth is limited to be 190 Hz and the integrator capacitor should be below 18 nF.



**Figure 5. First-Order Continuous-Time  $\Sigma\Delta$  AD Converter Circuit Diagram**

Typically, application systems have two separate capacitors across the power pins. In this case there should be a bulk electrolytic capacitor, such as a 1- $\mu$ F tantalum capacitor, to provide bulk charge storage for the overall system and a 0.1- $\mu$ F ceramic bypass capacitor located as close to the MCU power pins as practical to suppress high-frequency noise.

The MCU oscillator is a traditional Pierce oscillator that can accommodate a crystal or ceramic resonator up to 16 MHz. Resistor  $R_3$  should be a low-inductance resistor such as a carbon composition resistor and its tolerance is not critical.  $C_2$  and  $C_3$  should be ceramic capacitors being typically in the 5-pF to 25-pF range and are chosen to match the requirements of a specific crystal or resonator. The crystal

**Implementing a 10-Bit Sigma-Delta Analog-to-Digital Converter Using the HC9S08Rx MCU Family Analog Comparator, Rev. 0.1**

## Application Description

manufacturer typically specifies a load capacitance that is the series combination of  $C_2$  and  $C_3$  that are usually the same size.

Figure 6 illustrates the typical  $\Sigma\Delta$  modulator output when a ramp signal is applied to the input. Note the density of digital 1's is proportional to the average input. If the input is just about the internal bandgap reference voltage the output signal oscillates at half the modulator sampling frequency. That is known as the idle pattern. Figure 7 and Figure 8 exhibit the typical  $\Sigma\Delta$  modulator response to triangle and sinusoidal input signals, respectively.

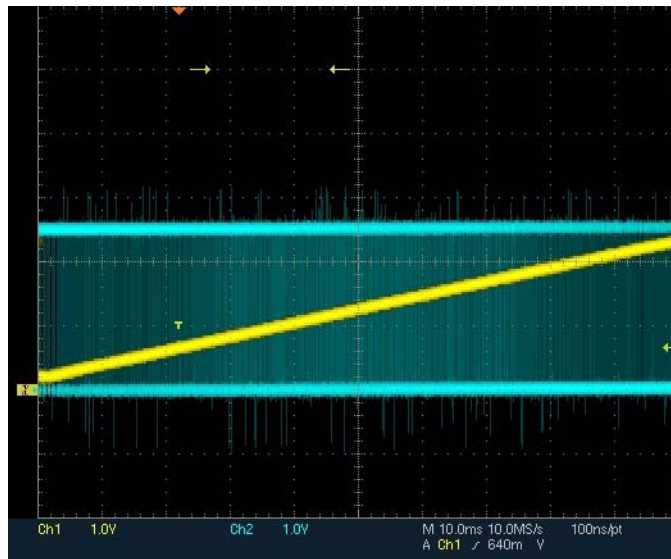


Figure 6. First-Order Continuous-Time  $\Sigma\Delta$  Modulator Response to a Ramp Input Signal

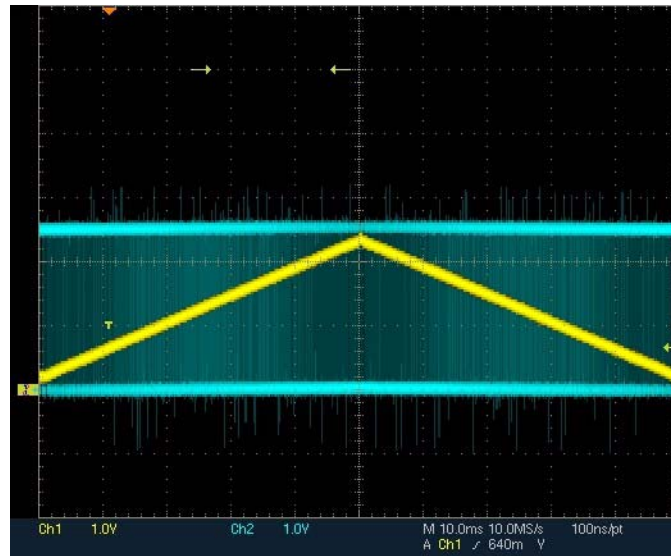
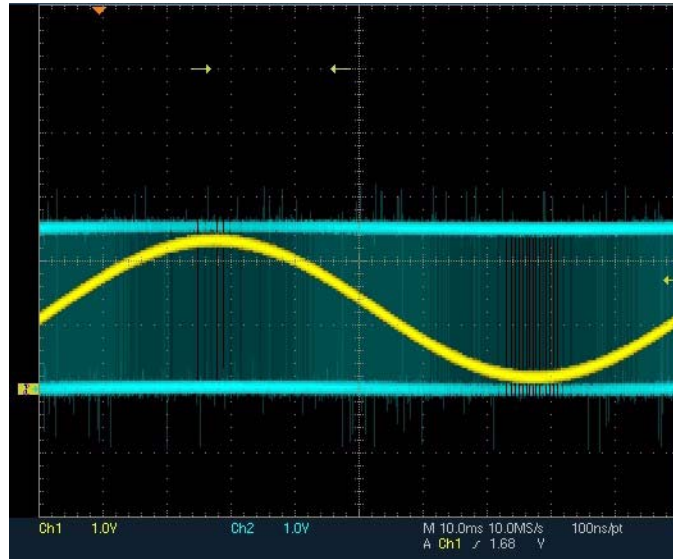


Figure 7.  $\Sigma\Delta$  Modulator Response to a Triangle Input Signal





**Figure 8.  $\Sigma\Delta$  Modulator Response to a Sinusoidal Input Signal**

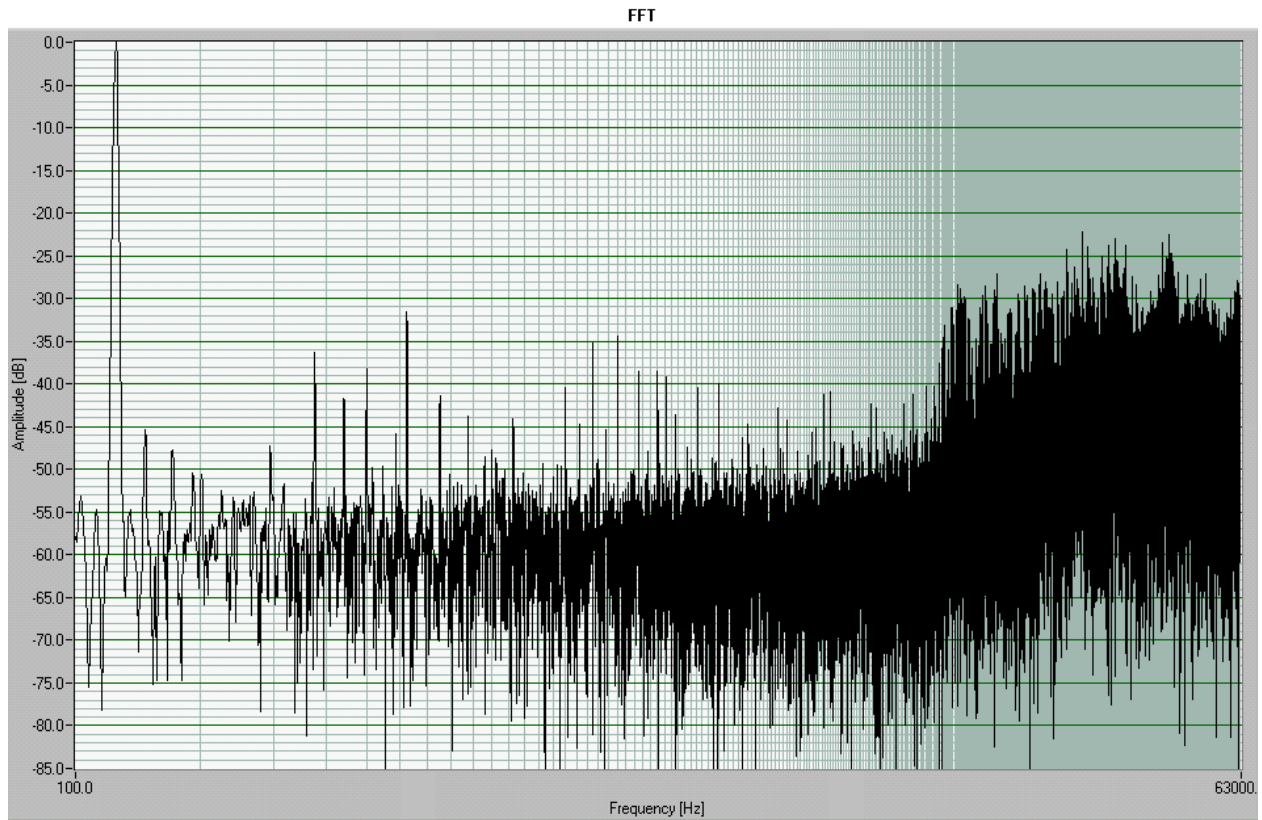
The  $\Sigma\Delta$  modulator evaluation is performed by analyzing the single-bit output signal spectrum through the Fast Fourier Transform (FFT). Input power is normalized to the quantization step defined by the comparator supply voltage whereas the sampling rate is set to 128 kHz with the timer interruption feature. [Figure 9](#) shows the output spectrum for a 125 Hz sinusoidal input signal with  $-8$  dB of normalized power.

#### NOTE

*The noise shaping concentrates the noise power in the high frequency range of the spectrum.*

The spectrum detail in [Figure 10](#) demonstrates a SNR of 55 dB for a  $-8$  dB input power.

[Figure 11](#) shows the measured  $\Sigma\Delta$  modulator SNR versus the input signal amplitude. Extrapolating from the data of [Figure 11](#), the measured dynamic range of the AD converter is found to be 65.7 dB and its maximum SNR is 61.2 dB. For large input signals, the precision of the converter is limited by quantization noise and harmonic distortion as apparent from the output spectrum in [Figure 9](#) and [Figure 10](#). The 10-bit digital output is monotonically linear as expected.



**Figure 9. Measured Output Spectrum of the First-Order Continuous-Time  $\Sigma\Delta$  Modulator**

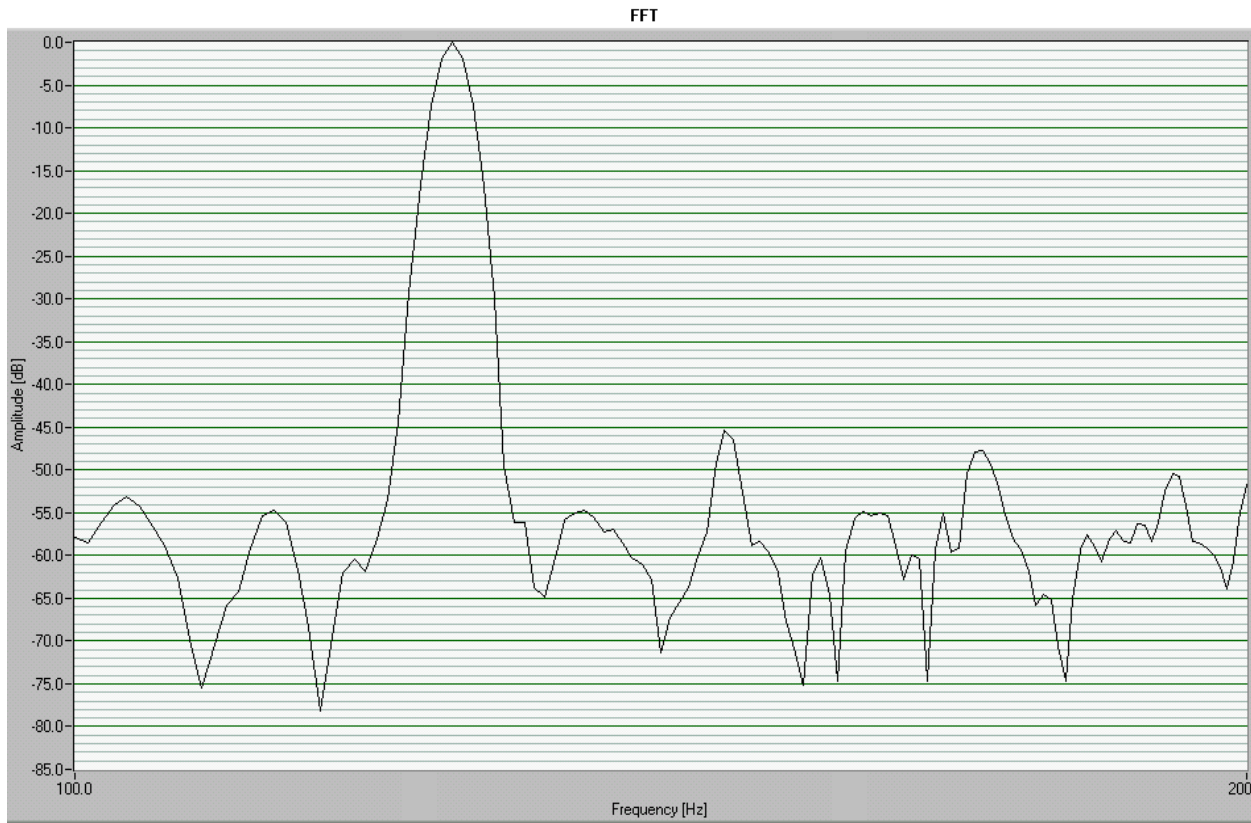


Figure 10. Measured Baseband Output Spectrum with -8 dB Sinusoidal Input Power

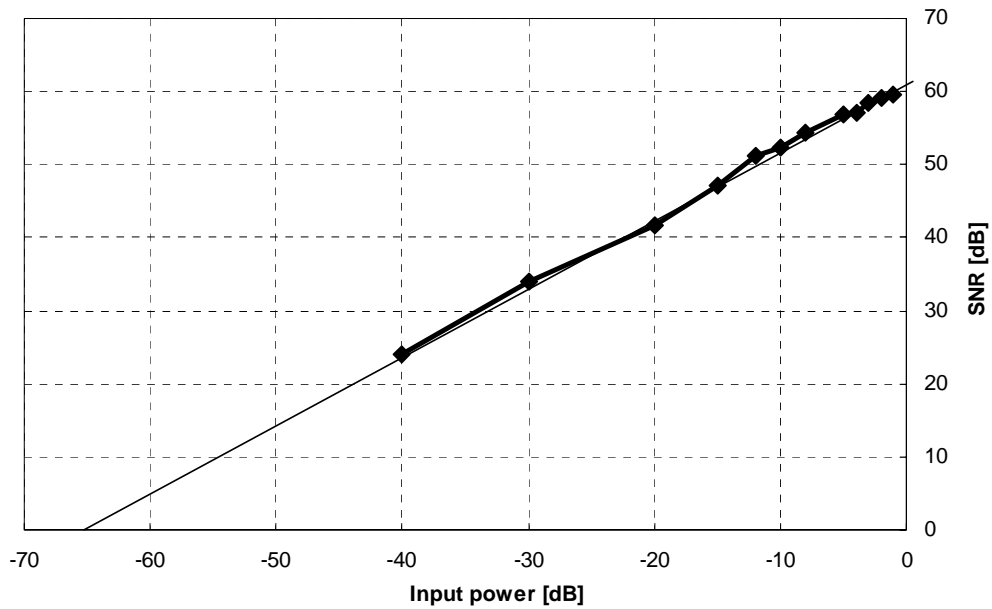
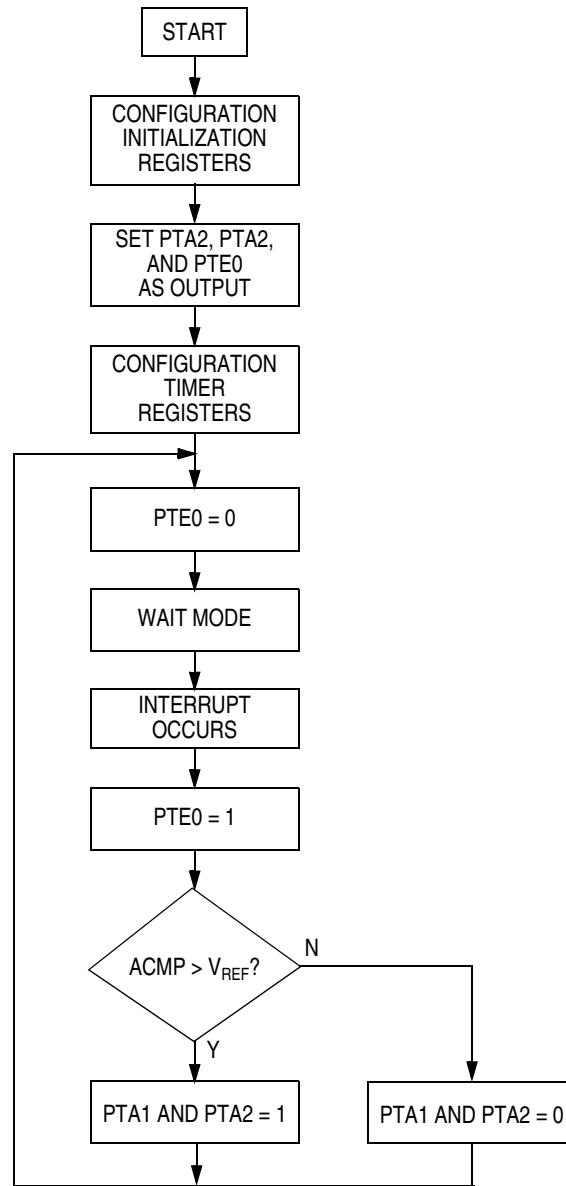


Figure 11. Measured  $\Sigma\Delta$  Modulator SNR for a 50-kHz Sampling Frequency and 125-Hz Input Signal Frequency

## Software Description

Two software options are listed in this application note. The first one is intended to implement a first-order continuous-time  $\Sigma\Delta$  modulator according to the flowchart in [Figure 12](#). The second one corresponds to the complete first-order continuous-time  $\Sigma\Delta$  AD converter in [Figure 5](#) and its flowchart is depicted in [Figure 13](#).



**Figure 12. The First-Order  $\Sigma\Delta$  Modulator Software Flowchart**

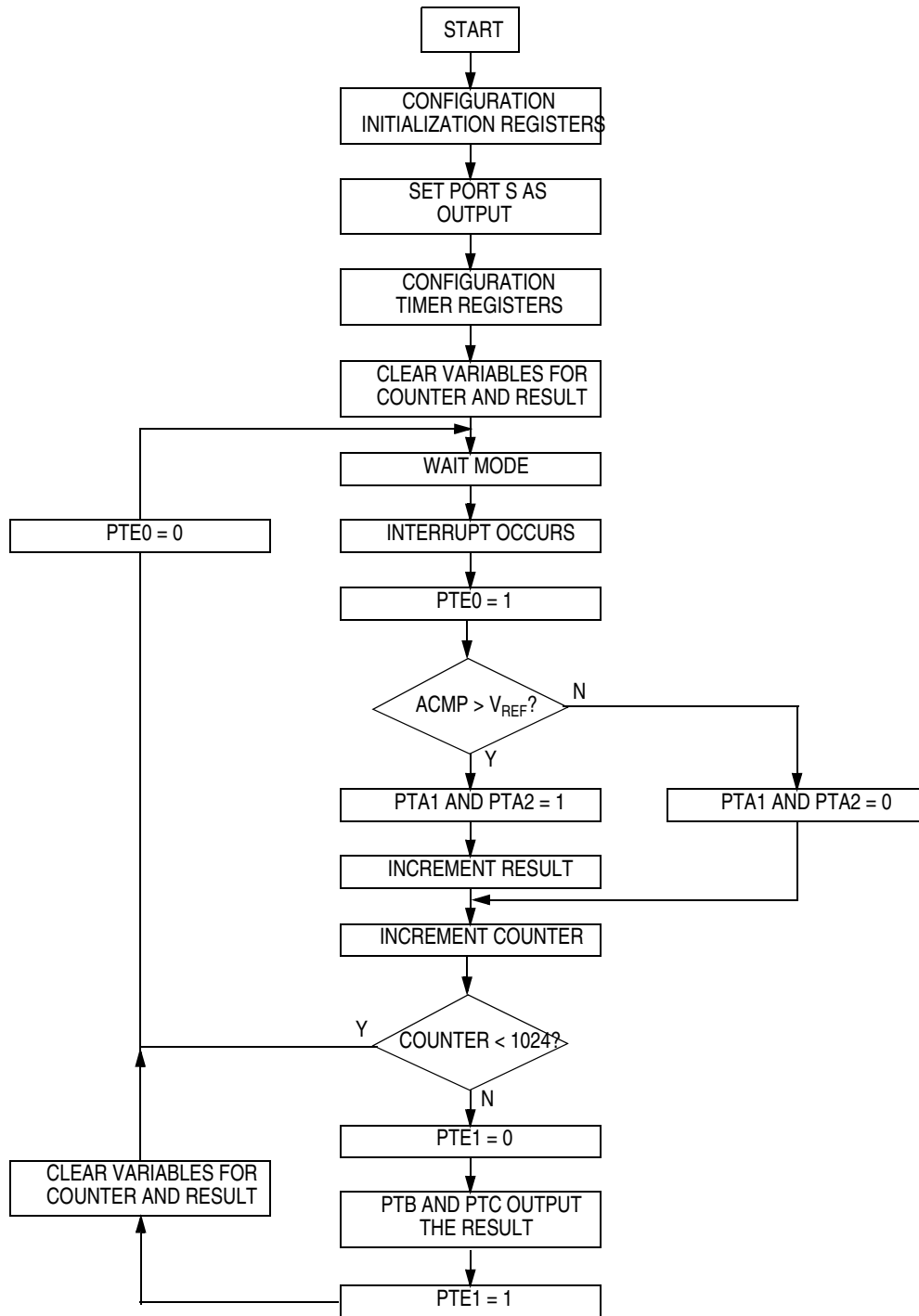


Figure 13. The Complete First-Order  $\Sigma\Delta$  AD Converter Software Flowchart

## Software Description

The continuous-time  $\Sigma\Delta$  modulator software:

- Configures the I/O ports and registers
- Sets the timer interrupt to a 128-kHz frequency
- Clears PTE0 and the MCU stays in wait mode until an interrupt is reached.
- After the timer overflows, PTE0 is set high and the comparator output flag (ACO) is verified.
  - If ACO is set, PTA1 and PTA2 are also set high
  - When ACO is cleared, PTA1 and PTA2 are cleared as well
- Code returns to wait mode and the process stays in a loop

The  $\Sigma\Delta$  AD converter software:

- Comprises both the modulator and the filter codes
- Configures the I/O ports and registers
- Sets the timer interrupt to occur periodically at a 60-kHz frequency
- Clears counter and result variables and the program enters wait mode
- Verifies the modulator routine ACO
- The difference between the codes is:
  - When ACO is set, PTA1 and PTA2 are also set and the result is incremented.
  - If ACO is cleared, PTA1 and PTA2 are cleared as well but the result is not incremented.
- The next step is to increment the counter and check its value.
  - If counter is less than 1024, decimal, the program returns to wait mode.
  - Otherwise, when the counter is at 1024, PTB and PTC ports output the AD conversion result.
- Both counter and results are cleared and the program returns to wait mode

## Software Listings

### SigmaDeltaM.asm

```

;*****
;* Title: SigmaDeltaM.asm (c) Freescale Inc. 2004 All rights reserved.
;*****
;* Author: Marcus Espindola - Freescale SPS/BSTC
;*
;* Description: Implementing a 10-bit Sigma-Delta Analog to Digital Converter Using the
;*              HC9S08Rx MCU Family Analog Comparator - Code for Sigma-Delta Modulator
;*
;* Documentation: 9S08RC16 Data Sheet for register and bit explanations
;*
;* Include Files: 9S08RC16.equ, SigmaDelta.equ
;*
;* Assembler: P&E Microcomputer Systems - CASM for HC08 and
;*              CodeWarrior 3.0
;*
;* Revision History:
;* Rev #      Date      Who      Comments
;* -----
;* 1.0       29-Jan-04   Espindola  Initial data entry
;*****
;* Freescale reserves the right to make changes without further notice to any product
;* herein to improve reliability, function, or design. Freescale does not assume any
;* liability arising out of the application or use of any product, circuit, or software
;* described herein; neither does it convey any license under its patent rights nor the
;* rights of others. Freescale products are not designed, intended, or authorized for
;* use as components in systems intended for surgical implant into the body, or other
;* applications intended to support life, or for any other application in which the
;* failure of the Freescale product could create a situation where personal injury or
;* death may occur. Should Buyer purchase or use Freescale products for any such
;* intended or unauthorized application, Buyer shall indemnify and hold Freescale and
;* its officers, employees, subsidiaries, affiliates, and distributors harmless against
;* all claims, costs, damages, and expenses, and reasonable attorney fees arising out
;* of, directly or indirectly, any claim of personal injury or death associated with
;* such unintended or unauthorized use, even if such claim alleges that Freescale was
;* negligent regarding the design or manufacture of the part.
;*
;* Freescale is a registered trademark of Freescale, Inc.
;*****

;              XDEF Entry,Reset,TPMInt

;*****
;* Equates and Data Table Includes
;*****

        include "9S08RC16.equ"

;DEFAULT_RAM          SECTION SHORT

        org    RamStart

```

**Implementing a 10-Bit Sigma-Delta Analog-to-Digital Converter Using the HC9S08Rx MCU Family Analog Comparator, Rev. 0.1**

## Software Listings

```

;*****
;* Constants and Variables for this file
;*****

        include 'SigmaDelta.equ'

;DEFAULT_ROM          SECTION

;*****
;* Code Section
;*****

        org    FlashStart

Entry:
Reset:   ldhx   #RamEndAddress+1      ; H:X = #RamEndAddress + 1
        txs                   ; Stack Pointer (SP) = #RamEndAddress

        lda   #setSIMOPT
        sta   SIMOPT             ;configure SIMOPT register

        lda   #setPMCSC1
        sta   PMCSC1            ;configure PMCSC1 register

        mov   #setACMPSC,ACMPSC   ;configure analog comparator

        clr   TPMCNTH
        clr   TPMCNTL           ;reset TPM counter

        mov   #setTPMMODH,TPMMODH
        mov   #setTPMMODL,TPMMODL ;a Timer interrupt will be generated every TPMMOD
                                   ;BUSCLKs (62 -> ~128kHz)

        bset  TOIE,TPMSC        ;Timer Overflow Interrupt Enable = 1
        bset  CLKSA,TPMSC       ;Clock Source Select = BUSCLK

        bclr  PTA1,PTA          ;clear bit 1 of port PTA
        bclr  PTA2,PTA          ;clear bit 2 of port PTA
        bset  DDRA1,DDRA        ;define bit 1 of port PTA as output
        bset  DDRA2,DDRA        ;define bit 2 of port PTA as output

        bclr  PTE0,PTE          ;clear bit 0 of port PTE
        bset  DDRE0,DDRE        ;define bit 0 of port PTE as output

WaitState: bclr  PTE0,PTE          ;clear bit 0 of port PTE
        wait                   ;enable interrupts and halt
        bra   WaitState         ;jump to WaitState

```



```

;*****
;* Timer Interrupt
;*****

TPMInt:    lda    TPMSC                ;read TPMSC register (required before clearing TOF)
           bclr  TOF,TPMSC            ;clear Timer Overflow Flag

           bset  PTE0,PTE              ;set bit 0 of port PTE

           brset ACO,ACMPSC,SetOutHigh ;jump to SetOutHigh if ACO == 1

           bclr  PTA1,PTA              ;clear bit 1 of port PTA
           bclr  PTA2,PTA              ;clear bit 2 of port PTA

           rti                          ;return from interrupt routine

SetOutHigh: bset  PTA1,PTA              ;set bit 1 of port PTA
            bset  PTA2,PTA              ;set bit 2 of port PTA

            rti                          ;return from interrupt routine

END

```

---



### SigmaDeltaF.asm

```

;*****
;* Title: SigmaDeltaF.asm (c) Freescale Inc. 2004 All rights reserved.
;*****
;* Author: Marcus Espindola - Freescale SPS/BSTC
;*
;* Description: Implementing a 10-bit Sigma-Delta Analog to Digital Converter Using the
;*              HC9S08Rx MCU Family Analog Comparator - Code for Sigma-Delta A2D Coverter
;*
;* Documentation: 9S08RC16 Data Sheet for register and bit explanations
;*
;* Include Files: 9S08RC16.equ, SigmaDelta.equ
;*
;* Assembler: P&E Microcomputer Systems - CASM for HC08 and
;*              CodeWarrior 3.0
;*
;* Revision History:
;* Rev #      Date      Who      Comments
;* -----
;* 1.0        29-Jan-04  Espindola  Initial data entry
;*****
;*****
;* Freescale reserves the right to make changes without further notice to any product
;* herein to improve reliability, function, or design. Freescale does not assume any
;* liability arising out of the application or use of any product, circuit, or software
;* described herein; neither does it convey any license under its patent rights nor the
;* rights of others. Freescale products are not designed, intended, or authorized for
;* use as components in systems intended for surgical implant into the body, or other
;* applications intended to support life, or for any other application in which the
;* failure of the Freescale product could create a situation where personal injury or
;* death may occur. Should Buyer purchase or use Freescale products for any such
;* intended or unauthorized application, Buyer shall indemnify and hold Freescale and
;* its officers, employees, subsidiaries, affiliates, and distributors harmless against
;* all claims, costs, damages, and expenses, and reasonable attorney fees arising out
;* of, directly or indirectly, any claim of personal injury or death associated with
;* such unintended or unauthorized use, even if such claim alleges that Freescale was
;* negligent regarding the design or manufacture of the part.
;*
;* Freescale is a registered trademark of Freescale, Inc.
;*****
;
;          XDEF Entry,Reset,TPMInt
;*****
;* Equates and Data Table Includes
;*****

include "9S08RC16.equ"

;DEFAULT_RAM          SECTION SHORT

org RamStart

```

### Implementing a 10-Bit Sigma-Delta Analog-to-Digital Converter Using the HC9S08Rx MCU Family Analog Comparator, Rev. 0.1

```

;*****
;* Constants and Variables for this file
;*****

        include 'SigmaDelta.equ'

;DEFAULT_ROM          SECTION

;*****
;* Code Section
;*****

        org    FlashStart

Entry:
Reset:  ldhx  #RamEndAddress+1          ; H:X = #RamEndAddress + 1
        txs                               ; Stack Pointer (SP) = #RamEndAddress

        lda  #setSIMOPT
        sta  SIMOPT                    ;configure SIMOPT register

        lda  #setPMCSC1
        sta  PMCSC1                    ;configure PMCSC1 register

        mov  #setACMPSC,ACMPSC;configure analog comparator

        clr  TPMCNTH
        clr  TPMCNTL                    ;reset TPM counter

        mov  #setTPMMODH,TPMMODH
        mov  #stTPMMODLF,TPMMODL      ;a Timer interrupt will be generated every TPMMOD
                                        ;BUSCLKs (133 -> ~60kHz)

        bset TOIE,TPMSC                ;Timer Overflow Interrupt Enable = 1
        bset CLKSA,TPMSC                ;Clock Source Select = BUSCLK

        bclr PTA1,PTA                  ;clear bit 1 of port PTA
        bclr PTA2,PTA                  ;clear bit 2 of port PTA
        bset DDRA1,DDRA                ;define bit 1 of port PTA as output
        bset DDRA2,DDRA                ;define bit 2 of port PTA as output

        clr  PTB                        ;reset port PTB
        clr  PTC                        ;reset port PTC
        mov  #OutPorts,DDRB            ;define port PTB as output
        mov  #OutPorts,DDRC            ;define port PTC as output

        clr  CounterL                  ;reset Counter (low order)
        clr  CounterH                  ;reset Counter (high order)
        clr  SampleL                   ;reset Sample (low order)
        clr  SampleH                   ;reset Sample (high order)

        bclr PTE0,PTE                  ;clear bit 0 of port PTE
        bclr PTE1,PTE                  ;clear bit 1 of port PTE
        bset DDRE0,DDRE                ;define bit 0 of port PTE as output
        bset DDRE1,DDRE                ;define bit 1 of port PTE as output
    
```



## Software Listings

```
WaitState:  wait                ;enable interrupts and halt (2+ cycles)
            bclr  PTE0,PTE      ;clear bit 0 of port PTE (5 cycles)
            bra   WaitState     ;jump to WaitState (3 cycles) -> Total = 10+ cycles

;*****
;* Timer Interrupt (min = 48 cycles / max = 108 cycles)
;*****

TPMInt:    lda    TPMSC        ;read TPMSC register (required before clearing TOF) (3 cycles)
            bclr  TOF,TPMSC    ;clear Timer Overflow Flag (5 cycles)

            bset  PTE0,PTE     ;set bit 0 of port PTE (5 cycles)

            brset ACO,ACMPSC,SetOutHigh ;jump to SetOutHigh if ACO == 1 (5 cycles)

            bclr  PTA1,PTA     ;clear bit 1 of port PTA (5 cycles)
            bclr  PTA2,PTA     ;clear bit 2 of port PTA (5 cycles)

            bra   IncCounter   ;jump to IncCounter (3 cycles)

SetOutHigh: bset  PTA1,PTA     ;set bit 1 of port PTA (5 cycles)
            bset  PTA2,PTA     ;set bit 2 of port PTA (5 cycles)

            inc   SampleL      ;increment Sample (low order) (5 cycles)
            bne   IncCounter   ;jump to IncCounter if SampleL did not overflow (3 cycles)

            inc   SampleH      ;increment Sample (high order) (5 cycles)

IncCounter: inc   CounterL     ;increment Counter (low order) (5 cycles)
            bne   TPMIntEnd    ;jump to TPMIntEnd if CounterL did not overflow (3 cycles)

            inc   CounterH     ;increment Counter (high order) (5 cycles)

            brclr 2,CounterH,TPMIntEnd ; jump to TPMIntEnd if Counter < 1024 ( 5 cycles)

            bclr  PTE1,PTE     ;clear bit 1 of port PTE (5 cycles)
            mov   SampleH,PTB  ;PTB = CounterH (5 cycles)
            mov   SampleL,PTC  ;PTC = CounterL (5 cycles)
            bset  PTE1,PTE     ;set bit 1 of port PTE (5 cycles)

            clr   CounterL     ;reset Counter (low order) (5 cycles)
            clr   CounterH     ;reset Counter (high order) (5 cycles)
            clr   SampleL      ;reset Sample (low order) (5 cycles)
            clr   SampleH      ;reset Sample (high order) (5 cycles)

TPMIntEnd: rti                ;return from interrupt routine (9 cycles)

END
```

## SigmaDelta.equ

```

;*****
;* Title: SigmaDelta.equ                                Copyright (c) Freescale 2004
;*****
;* Author: Marcus Espindola - Freescale SPS/BSTC
;*
;* Description: Constants and variables definitions for 9S08RC16.
;*
;* Documentation: 9S08RC16 Data Sheet for register and bit explanations
;*
;* Include Files:
;*
;* Assembler: P&E Microcomputer Systems - CASM for HC08 and
;*              CodeWarrior 3.0
;*
;* Revision History:
;* Rev #      Date      Who      Comments
;* -----
;* 1.0        29-Jan-04  Espindola  Initial data entry
;*****
;*****
;* Freescale reserves the right to make changes without further notice to any product
;* herein to improve reliability, function, or design. Freescale does not assume any
;* liability arising out of the application or use of any product, circuit, or software
;* described herein; neither does it convey any license under its patent rights nor the
;* rights of others. Freescale products are not designed, intended, or authorized for
;* use as components in systems intended for surgical implant into the body, or other
;* applications intended to support life, or for any other application in which the
;* failure of the Freescale product could create a situation where personal injury or
;* death may occur. Should Buyer purchase or use Freescale products for any such
;* intended or unauthorized application, Buyer shall indemnify and hold Freescale and
;* its officers, employees, subsidiaries, affiliates, and distributors harmless against
;* all claims, costs, damages, and expenses, and reasonable attorney fees arising out
;* of, directly or indirectly, any claim of personal injury or death associated with
;* such unintended or unauthorized use, even if such claim alleges that Freescale was
;* negligent regarding the design or manufacture of the part.
;*
;* Freescale is a registered trademark of Freescale, Inc.
;*****
;*****
;* Constants and Variables for this file
;*****

;          org    RamStart

SampleH:   rmb    1
SampleL:   rmb    1

CounterH:  rmb    1
CounterL:  rmb    1

```

## Software Listings

```

setSIMOPT: equ %01000001 ;System Options Register
;          ||| ||| |||
;          ||| ||| ||| +-RESET Pin Enable = 1
;          ||| ||| ||| +-Background Debug Mode Pin Enable = 0
;          ||| ||| +-Unused during write
;          ||| +-Unused during write
;          || +-Unused
;          |+-----Stop Mode Enable = 0
;          |+-----COP Watchdog Timeout = Long Timeout
;          +-----COP Watchdog Enable = 0

setPMCS1: equ %01011000 ;System Power Management Status and Control 1 Register
;          ||| ||| |||
;          ||| ||| ||| +-Unused during write
;          ||| ||| +-Unused during write
;          ||| ||| +-Unused during write
;          ||| +-Low-Voltage Detect Reset Enable = 1
;          || +-SAFE System from interrupts = 1
;          |+-----Low-Voltage Detect Interrupt Enable = 0
;          |+-----Clear Low-Voltage Detect Flag
;          +-----Unused during write

setACMPSC: equ %11100011 ;ACMP Status and Control Register
;          ||| ||| |||
;          ||| ||| ||| +-Comparator output either rising or falling edge
;          ||| ||| +-Comparator output either rising or falling edge
;          ||| ||| +-Unused during write
;          ||| +-Unused during write
;          || +-Analog Comparator Interrupt Enable = 0
;          |+-----Clear Analog Comparator Flag
;          |+-----Analog Comparator Bandgap Select = 1
;          +-----Analog Comparator Module Enable = 1

setTPMMODH: equ $00      ;Timer Counter Modulo High Register value
setTPMMODL: equ $3E      ;Timer Counter Modulo Low Register value for modulator
stTPMMODLF: equ $85      ;Timer Counter Modulo Low Register value for filter

OutPorts: equ $FF        ;Set ports as output

;*****
;* Interrupt Vector Table
;*****

        org $FFF2        ;Timer Overflow Interrupt Vector
        fdb TPMInt

        org $FFFE        ;Reset Interrupt Vector
        fdb Reset

```



## **How to Reach Us:**

### **Home Page:**

[www.freescale.com](http://www.freescale.com)

### **E-mail:**

[support@freescale.com](mailto:support@freescale.com)

### **USA/Europe or Locations Not Listed:**

Freescale Semiconductor  
Technical Information Center, CH370  
1300 N. Alma School Road  
Chandler, Arizona 85224  
+1-800-521-6274 or +1-480-768-2130  
[support@freescale.com](mailto:support@freescale.com)

### **Europe, Middle East, and Africa:**

Freescale Halbleiter Deutschland GmbH  
Technical Information Center  
Schatzbogen 7  
81829 Muenchen, Germany  
+44 1296 380 456 (English)  
+46 8 52200080 (English)  
+49 89 92103 559 (German)  
+33 1 69 35 48 48 (French)  
[support@freescale.com](mailto:support@freescale.com)

### **Japan:**

Freescale Semiconductor Japan Ltd.  
Headquarters  
ARCO Tower 15F  
1-8-1, Shimo-Meguro, Meguro-ku,  
Tokyo 153-0064  
Japan  
0120 191014 or +81 3 5437 9125  
[support.japan@freescale.com](mailto:support.japan@freescale.com)

### **Asia/Pacific:**

Freescale Semiconductor Hong Kong Ltd.  
Technical Information Center  
2 Dai King Street  
Tai Po Industrial Estate  
Tai Po, N.T., Hong Kong  
+800 2666 8080  
[support.asia@freescale.com](mailto:support.asia@freescale.com)

### **For Literature Requests Only:**

Freescale Semiconductor Literature Distribution Center  
P.O. Box 5405  
Denver, Colorado 80217  
1-800-441-2447 or 303-675-2140  
Fax: 303-675-2150  
[LDCForFreescaleSemiconductor@hibbertgroup.com](mailto:LDCForFreescaleSemiconductor@hibbertgroup.com)

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

© Freescale Semiconductor, Inc. 2005 All rights reserved.