Introduction:

This Technical Note describes a software routine used to calibrate the ADuC812 for ADC endpoint offset and gain errors. The software referred to below can be found, along with this technical note, on the web at http://ww.analog.com/microconverter/technotes_code.html. This document describes both the functions of the software routines and their use in standard applications.

The ADuC812 has 2 ADC calibration coefficients, 1 for offset calibration and 1 for gain calibration. Both the offset and gain calibration coefficients are 14 bit words, located in the Special Function Register (SFR) area. The offset calibration coefficient is divided into ADCOFSH (6 bits) and ADCOFSL (8 bits) and the gain calibration coefficient is divided into ADCGAINH (6 bits) and ADCGAINL (8 bits).

Each ADuC812 is calibrated in the factory prior to shipping and the offset and gain calibration coefficients are stored in a hidden area of FLASH/EE memory. Each time the ADuC812 powers up, an internal power-on configuration routine, copies these coefficients into the offset and gain calibration registers in the SFR area.

The MicroConverter ADC accuracy may vary from system to system due to board layout, grounding, clock speed, etc. To get the best ADC accuracy in your system, you should perform the software calibration routine described in this technical note.

The Offset and Gain Calibration Coefficients:

The offset calibration coefficient compensates for DC offset errors in both the ADC and the input signal. Increasing the offset coefficient compensates for positive offset, and effectively pushes the ADC Transfer Function DOWN. Decreasing the offset coefficient compensates for negative offset, and effectively pushes the ADC Transfer Function UP. The maximum offset that can be compensated is typically ± 5% of VREF, which equates to typically ±125mV with a 2.5V reference.

Similarly, the gain calibration coefficient compensates for DC gain errors in both the ADC and the input signal. Increasing the gain coefficient, compensates for a smaller analog input signal range and scales the ADC Transfer Function UP, effectively increasing the slope of the transfer function. Decreasing the gain coefficient, compensates for a larger analog input signal range and scales the ADC Transfer Function DOWN, effectively decreasing the slope of the transfer function. The maximum analog input signal range for which the gain coefficient can compensate is 1.025 * VREF and the minimum input range is 0.975 * VREF which equates to typically ± 2.5% of the reference voltage.

ADuC812 ADC Offset Calibration Subroutine:

The offset calibration subroutine is described below and is called….

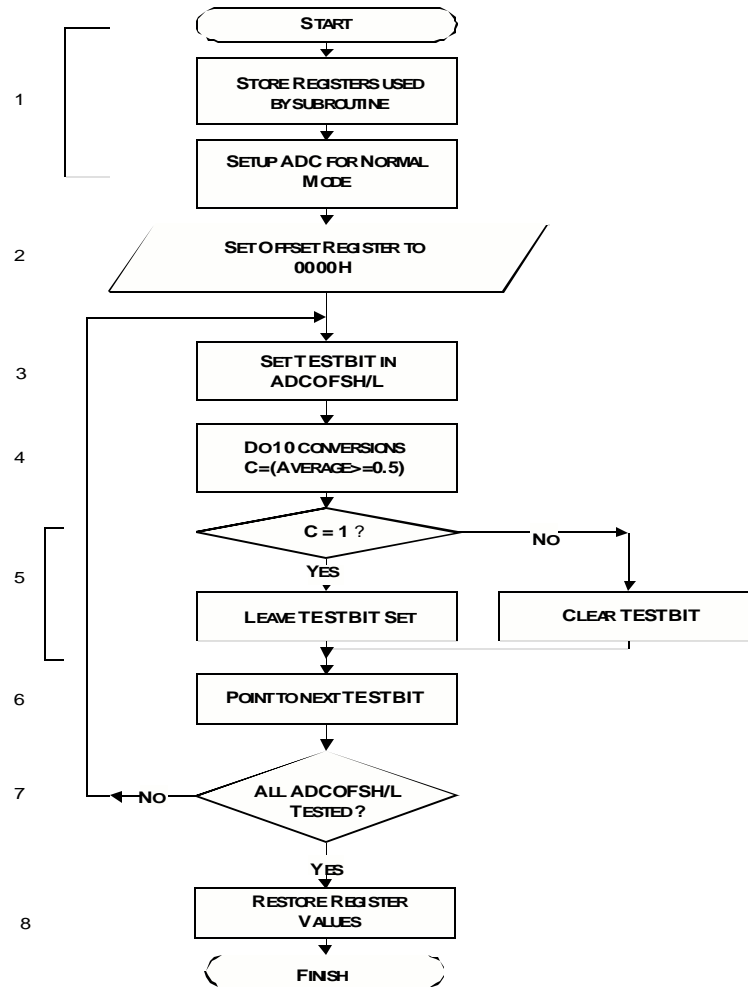> ***CALZEROSCALE***   - calibrates ADC for offset

This subroutine can be found in the '812cal.asm' assembly file accompanying this technical note. It calibrates the ADC for offset error on a selected channel passed to the subroutine via the Accumulator (ACC). The selected channel must be externally held at a fixed system zero-scale voltage i.e. zero volts ± (5% of VREF), when the subroutine is called.

This subroutine employs a successive approximation algorithm. On the first iteration, the subroutine sets the MSB (TESTBIT) in the ADCOFSH/L registers (i.e. sets the ADCOFSH/L registers to 2000h). 10 ADC conversions are generated and the Carry flag (C) is set if the average of these results is greater than or equal to 0.5. If C = 1, the TESTBIT (MSB) is left in the set state, and if C = 0, the TESTBIT is cleared. On the second iteration, the MSB-1 bit now becomes the TESTBIT bit, and the same test sequence is performed. This test sequence is repeated for each of the 14 bits in the ADCOFSH/L registers.

This subroutine uses 8 bytes of Stack to store registers used by the code. These registers are restored when the subroutine has completed.

The Offset Calibration Algorithm Flow:

1. General Housekeeping code, i.e. store registers used by the code on the stack, set up ADC, etc.

2. Clear the Offset Calibration coefficient (ADCOFSH/L registers) to 0000Hex.

3. Set the current TESTBIT in the Offset Calibration. On the first iteration the MSB is set to 1, effectively setting the Offset Calibration coefficient to 2000H.

4. Perform 10 ADC conversions, get the average and return C = (average result >=0.5).

5. If C = 1, leave TESTBIT set, if C = 0 then clear the TESTBIT

6. Point to next bit to be tested. After the first iteration, the MSB-1 bit becomes the new TESTBIT.

7. Repeat steps 3, 4, 5 and 6 until all bits in the ADCOFSH/L registers have been tested.

8. Restore registers stored on the stack in step 1.

**ANALOG
DEVICES**

```
                          ┌────────────────┐
                          │     START       │
                          └────────┬───────┘
                                   │
1          ┌──────────────────────────────────────┐
           │   STORE REGISTERS USED               │
           │   BY SUBROUTINE                      │
           └──────────────────┬───────────────────┘
           │   SETUP ADC FOR NORMAL               │
           │   MODE                               │
           └──────────────────┬───────────────────┘
                              │
2          ╱   SET OFFSET REGISTER TO     ╱
          ╱    0000H                      ╱
                              │
3          │   SET TESTBIT IN                     │
           │   ADCOFSH/L                          │
                              │
4          │   DO 10 CONVERSIONS                  │
           │   C=(AVERAGE>=0.5)                   │
                              │
5          ◇        C = 1 ?        ◇ ──NO──►
              YES │
           │  LEAVE TESTBIT SET  │    │  CLEAR TESTBIT  │
                              │
6          │  POINT TO NEXT TESTBIT │
                              │
7   ◄──NO── ◇  ALL ADCOFSH/L   ◇
              TESTED ?
              YES │
8          │  RESTORE REGISTER   │
           │  VALUES             │
                              │
                          ┌────────────────┐
                          │     FINISH      │
                          └────────────────┘
```

ADuC812 ADC Gain Calibration Subroutine:

The second subroutine calibrates the ADC for gain error and is called….
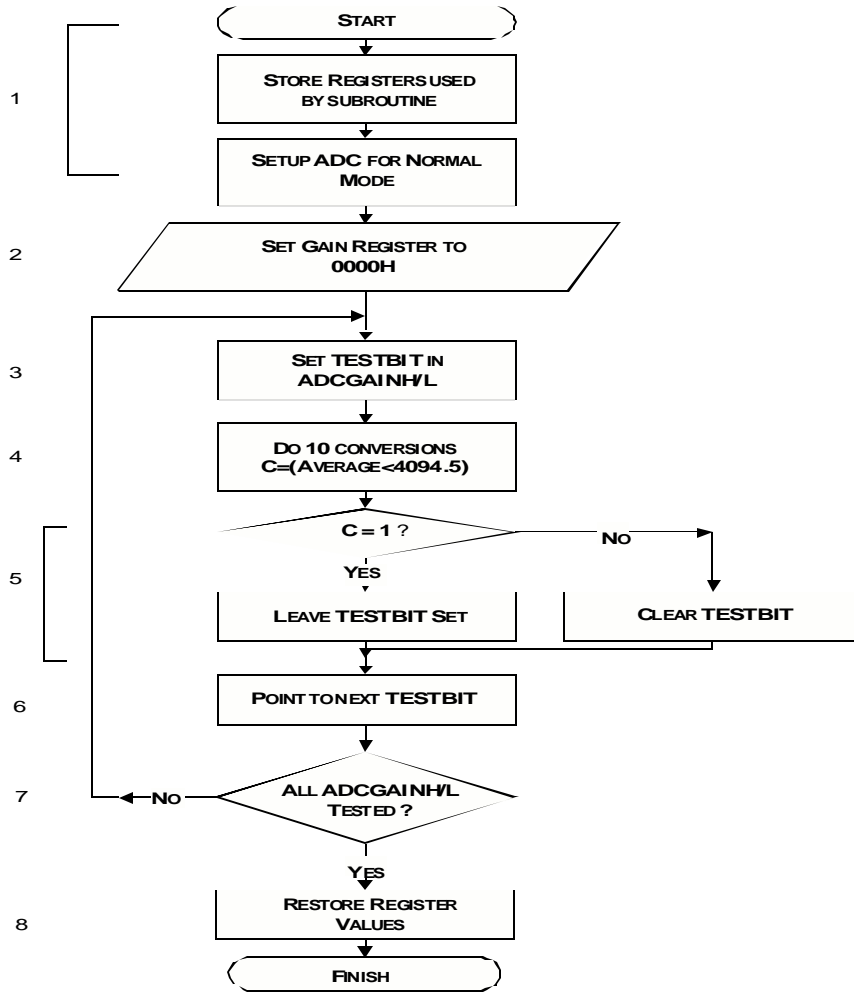
> **CALFULLSCALE** - calibrates ADC for gain

This subroutine can also be found in the '812cal.asm' assembly file accompanying this technical note. It calibrates the ADC for gain error on a selected channel passed to the subroutine via the ACC. The selected channel must be externally held at a fixed system full-scale reference voltage (within ±2.5% of VREF) when the subroutine is called.

This subroutine also employs a successive approximation algorithm. On the first iteration, the subroutine sets the MSB (TESTBIT) in the ADCGAINH/L registers (i.e. sets the ADCGAINH/L register to 2000h). 10 ADC conversions are generated and the Carry flag (C) is set if the average of these results is less than 4094.5. If C = 1, the TESTBIT (MSB) is left in the set state, if C = 0, then the TESTBIT is cleared. On the second iteration, the MSB-1 bit now becomes the TESTBIT bit, and the same test sequence is performed. This test sequence is repeated for each of the 14 bits in the ADCGAINH/L registers.

This subroutine uses 8 bytes of Stack to store registers used by the code. These registers are restored when the subroutine has completed.

The Gain Calibration Algorithm Flow:

1. General Housekeeping code, i.e. store registers used by the code on the stack, set up ADC, etc.

2. Clear the Gain Calibration coefficient (ADCGAINH/L registers) to 0000Hex

3. Set the current TESTBIT in the Gain Calibration coefficient. On the first iteration, the MSB is set to 1, effectively setting the Gain Calibration coefficient to 2000H.

4. Perform 10 ADC conversions, get the average and return C = (average result < 4094.5).

5. If C = 1, leave TESTBIT set, if C = 0 then clear the TESTBIT.

6. Point to next bit to be tested. After the first iteration, the MSB-1 bit becomes the new TESTBIT.

7. Repeat steps 3, 4, 5 and 6 until all bits in the ADCGAINH/L registers have been tested.

8. Restore registers stored on the stack in step 1.

START

1

STORE REGISTERS USED
BY SUBROUTINE

SETUP ADC FOR NORMAL
MODE

2

SET GAIN REGISTER TO
0000H

3

SET TESTBIT IN
ADCGAINH/L

4

DO 10 CONVERSIONS
C=(AVERAGE<4094.5)

C = 1 ?

NO

YES

5

LEAVE TESTBIT SET

CLEAR TESTBIT

6

POINT TO NEXT TESTBIT

7

NO

ALL ADCGAINH/L
TESTED ?

YES

8

RESTORE REGISTER
VALUES

FINISH

Evaluating the Offset and Gain Calibration Subroutines:

If you wish to evaluate the performance of the calibration subroutines without writing your own code, you can do so using this technical notes' companion code. The following description assumes the user is using the ADuC812 Quickstart Development System evaluation board and software tools, it also assumes that the software tools are installed in the default directory 'C:\ADuC'. The program '812cal00.asm' includes '812cal.asm' and 'UARTIO.asm' and allows the user to evaluate the offset and gain calibration subroutines using the pre-configured 'hyper terminal' program supplied with the ADuC812 QuickStart Development Kit.

This program demonstrates the offset and gain factory calibrated coefficients on your device as well as the new software calibrated coefficients generated using these routines. To use this program, follow the steps below.

1. Copy the '812cal00.hex' program to the 'ADuC\DownLoad' directory.
2. Ensure LK3 (PSEN) is inserted. double-click the *DownLoad* icon () and In the window that comes up, type '812cal00.hex' and click OK.

   The Download program will display the below information, indicating that the serial download was successful.

3. This program will communicate with any VT100 terminal, or any PC-based terminal emulator program (such as "hyperterminal" for windows), configured for 9600baud/8bits/no-parity/1-stop-bit. Hyperterminal users will find a simple configuration file packaged with the MicroConverter development tools. To open the Hyperterminal program, double-click the *9600COM1* icon (), located in the 'C:\ADuC' directory.

4. Remove LK3 (PSEN) and press the RESET button on the ADuC812 evaluation board. The following screen should appear.

```
Analog Devices MicroConverter ADuC812 - Calibration Demo Routine

offset coefficients (ADCOFS):  default=1A38 / present=1A38
gain coefficients  (ADCGAIN):  default=1EC8 / present=1EC8

ADC results (channel 0): 004,006,000,002,006,005,002,007,000,009

options:  0 - convert again & refresh screen
          1 - perform zero-scale calibration
          2 - perform full-scale calibration
          3 - select new primary ADC channel  ...choose option:_
```

The 'default' offset and gain coefficients are those calibrated in the factory and downloaded by the power-on configuration. The 'present' offset and gain calibration coefficients are the current contents of ADCOFSH/L and ADCGAINH/L. Both the 'default' and 'present' values will be different for each ADuC812. ADC results are 10 consecutive conversions on the selected channel (0 by default and in the case shown). This channel can be changed using option 3, which will be dealt with after we have calibrated the ADC.

5. A zero-scale calibration (Option 1) is selected by typing '1' on the keyboard and this adds another 2 lines to the above screen (see below) prompting the user to choose a particular ADC input to use when calibrating and to remind the user to connect that input to 'AGND'.

```
Analog Devices MicroConverter ADuC812 - Calibration Demo Routine

offset coefficients (ADCOFS):  default=1A38 / present=1A38
gain coefficients  (ADCGAIN):  default=1EC8 / present=1EC8

ADC results (channel 0): 004,006,000,002,006,005,002,007,000,009

options:  0 - convert again & refresh screen
          1 - perform zero-scale calibration
          2 - perform full-scale calibration
          3 - select new primary ADC channel  ...choose option:1

apply 0V to an ADC input, and enter the channel number (0-7)
of that input to launch the zero-scale calibration routine:_
```

6. After entering the channel number, a zero-scale calibration is performed.

```
performing zero-scale calibration....OK
_____

Analog Devices MicroConverter ADuC812 - Calibration Demo Routine

offset coefficients (ADCOFS):  default=1A38 / present=1AA9
gain coefficients  (ADCGAIN): default=1EC8 / present=1EC8

ADC results (channel 0): 000,000,003,000,000,000,001,000,002,000

options:  0 - convert again & refresh screen
          1 - perform zero-scale calibration
          2 - perform full-scale calibration
          3 - select new primary ADC channel  ...choose option:_
```

From the picture above, you can see that while the 'default' offset coefficient stays the same, the 'present' value changes. In this case the value in the offset register increases and therefore pushes the transfer function of the ADC down and the average of the 'ADC results' gets closer to zero.

7.   The next step is to perform a full-scale scale calibration by choosing option '2'. This brings up the following up the following screen.

```
_____

Analog Devices MicroConverter ADuC812 - Calibration Demo Routine

offset coefficients (ADCOFS):  default=1A38 / present=1AA9
gain coefficients  (ADCGAIN): default=1EC8 / present=1EC8

ADC results (channel 0): 000,000,003,000,000,000,001,000,002,000

options:  0 - convert again & refresh screen
          1 - perform zero-scale calibration
          2 - perform full-scale calibration
          3 - select new primary ADC channel  ...choose option:2

apply Vref to an ADC input, and enter the channel number (0-7)
of that input to launch the full-scale calibration routine:_
```

This screen which prompts the user to connect an ADC channel to VREF and enter the number of that channel(0 in this example). Once the channel number has been entered, the program performs a full-scale calibration.

```
performing full-scale calibration....OK
_____

Analog Devices MicroConverter ADuC812 - Calibration Demo Routine

offset coefficients (ADCOFS):  default=1A38 / present=1AA9
gain coefficients  (ADCGAIN):  default=1EC8 / present=1EE7

ADC results (channel 0): FFF,FFF,FFF,FFF,FF2,FFF,FFD,FFF,FFF,FFF

options:  0 - convert again & refresh screen
          1 - perform zero-scale calibration
          2 - perform full-scale calibration
          3 - select new primary ADC channel  ...choose option:_
```

The 'present' gain coefficient has now been changed to '1EE7' which is the current value in the ADCGAINH/L registers. As this value is greater than the factory calibrated value, this has the effect of scaling the transfer function of the ADC up, effectively increasing the slope of the transfer function. The calibration of the ADC is now complete.

8. After calibration, ADC channel 0 was connected to a voltage source set to mid-scale (1.25V). Using option 0 the following ADC results were gathered.

```
_____

Analog Devices MicroConverter ADuC812 - Calibration Demo Routine

offset coefficients (ADCOFS):  default=1A38 / present=1AA9
gain coefficients  (ADCGAIN):  default=1EC8 / present=1EE7

ADC results (channel 0): 7FB,800,7FD,802,800,801,800,803,7FE,801

options:  0 - convert again & refresh screen
          1 - perform zero-scale calibration
          2 - perform full-scale calibration
          3 - select new primary ADC channel  ...choose option:
```

9. Option 3 above allows the user to select a different channel to take conversions. Channel 2 was connected to mid-scale and then selected using option 3. When selecting a channel other than the one used when calibrating the ADC, Channel - to - Channel 'Offset Error Matching' and 'Gain Error Matching' specifications of the ADC input multiplexer should be taken into account. These are typically ± 1LSB and ± 1.5LSB respectively at VDD = 5V.

```
enter new channel for ADC conversions (0-8):2
_____

Analog Devices MicroConverter ADuC812 - Calibration Demo Routine

offset coefficients (ADCOFS):  default=1A38 / present=1AA9
gain coefficients  (ADCGAIN): default=1EC8 / present=1EE7

ADC results (channel 2): 807,800,7FF,801,7FF,7FF,800,806,7FD,7FF

options:  0 - convert again & refresh screen
          1 - perform zero-scale calibration
          2 - perform full-scale calibration
          3 - select new primary ADC channel  ...choose option:
```

The ADC results reported now are from ADC channel 2 as shown above.

Though you can use the un-modified ADuC812 evaluation board to very simply see the function of the calibration routine as described above, the single-supply op-amps that buffer the ADC inputs on the evaluation board will degrade performance (specifically noise and repeatability) at voltages near ground. Therefore, best *offset-cal* results will be obtained only when driving the ADC input directly from a low-impedance source such as a voltage calibrator or a dual supply op-amp. Gain-cal performance is not appreciably affected by single supply op-amps when using a 2.5V reference.

It should be noted that the new offset and gain calibration coefficients are a result of a software calibration and are the current values stored in the Special Function Register area registers ADCOFSH/L and ADCGAINL. These are volatile values and will be lost when power is removed or the ADuC812 has been RESET. On 'power up' or 'reset', the factory calibrated offset and gain coefficients are always copied from a hidden Flash/EE space as part of a 'power-on configuration routine'.

Integrating the Offset and Gain Calibration Routines into your Code:

A method to integrate the calibration subroutines into your code is shown below. This uses the '$INCLUDE' directive which includes and integrates the program '812cal.asm' (330 bytes approx.) into your code. This example requires ADC input 0 to be at AGND and input 1 to be at VREF before the code executes.

It should be noted that for optimal ADC endpoint calibration, the *Offset Calibration* should be performed **Before** the *Gain Calibration*.

```
MAIN:  SETB  LED
…
; note: channel number is passed to calibration routine via ACC
…
       MOV    A,#00H             ; pass channel number for zero-scale calibration to subroutine via accumulator
       CALL   CALZEROSCALE       ; perform zero-scale calibration
       MOV    A,#01H             ; pass channel number for full-scale calibration to subroutine via accumulator
       CALL   CALFULLSCALE       ; perform full-scale calibration

       JMP    MAIN

;_____
                    ; SUBROUTINE INCLUDE FILES
```

```
$INCLUDE(812CAL.asm)                    ; Assembler Directive to Integrate the file containing calibration subroutines
END
```

Calibrate Once Only and Store Coefficients in Flash/EE Data Space:

Sometimes it is desirable to perform a calibration only once in the factory when your product is first manufactured. This avoids the extra steps of calibrating every time power is applied. In these situations, the non-volatile Flash/EE data memory of the ADuC812 can be used to store the calibration results. These results, once stored, can be copied into the calibration coefficient registers in the SFR space whenever power is applied to the system.

In example code segment below, page 0 of the Flash/EE data memory space is used to store the four bytes of calibration data (2-bytes for offset + 2-bytes for gain). The code performs a zero-scale calibration followed by a full-scale calibration. It reads page 0, and copies the ADCOFSH/L and ADCGAINH/L registers to EDATA1, 2, 3, and 4 Flash/EE data registers in the SFR area. It then performs an erase of page 0 and finally programs the contents of the EDATA registers into the Flash/EE data memory space.

Note, if this method is used, the chosen ADC input channels for offset and gain calibration (channels 7 & 6 in this example) *must* be connected to ground and V_{REF} while the code executes.

```
MAIN:   MOV     A,#7                    ; select channel for zero-scale calibration
        CALL    CALZEROSCALE            ; perform zero-scale (offset) calibration
        JC      CALERROR                ; indicate error if calibration fails
        MOV     A,#6                    ; select channel for full-scale calibration
        CALL    CALFULLSCALE            ; perform full-scale (gain) calibration
        JC      CALERROR                ; indicate error if calibration fails

        MOV     EADRL,#0                ; point to page 0 of Flash/EE data space
        MOV     ECON,#1                 ; Read page 0
        MOV     EDATA1,ADCOFSL          ; move the cal coefficients into EDATA1,2,3,4..
        MOV     EDATA2,ADCOFLH
        MOV     EDATA3,ADCGAINL
        MOV     EDATA4,ADCGAINH
        MOV     ECON,#5                 ; Erase page 0
        MOV     ECON,#2                 ; ..and program EDATA1,2,3,4 to page 0 of Flash/EE space
;       the rest of the main program would reside here.

$INCLUDE(812cal.asm)                    ; includes the calibration subroutines
```

As mentioned previously, the code segment above is typically executed as part of a factory production test routine. However, the code segment below can be integrated into the final application program. This code simply copies the software calibrated coefficients from the first page of the Flash/EE data memory to their designated SFR locations.

```
MAIN:   MOV     EADRL,#0                ; point to page 0 of Flash/EE data space
        MOV     ECON,#1                 ; read data from page 0 into EDATA1,2,3,4
        MOV     ADCOFSL,EDATA1          ; restore ADC cal coefficients..
        MOV     ADCOFSH,EDATA2
        MOV     ADCGAINL,EDATA3
        MOV     ADCGAINH,EDATA4         ; ..and continue program
;       the rest of the main program would reside here.
```