TEXAS INSTRUMENTS

Architecture and Function of the MSP430 14-Bit ADC

Application Report

June 1999

Mixed Signal Products

SLAA045

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.

Contents

List of Figures

List of Tables

Architecture and Function of the MSP430 14-Bit ADC

Lutz Bierl

ABSTRACT

This application report describes the architecture and function of the 14-bit analog-to-digital converter (ADC) of the MSP430 family. The principles of the ADC are explained and software examples are given. The report also explains the function of all hardware registers in the ADC. The *References* section at the end of the report lists related application reports in the MSP430 14-bit ADC series.

1 Introduction

The analog-to-digital converter (ADC) of the MSP430 family can work in two modes: the 12-bit mode or the 14-bit mode. Hardware registers allow easy adaptation to different ADC tasks. The following paragraphs describe the modes and hardware registers.

> **NOTE:** The *MSP430 Family Architecture Guide and Module Library* data book[1] is recommended. The hardware-related information given there is very valuable and complements the information given in this application report.

> **NOTE:** For related application reports in the MSP430 14-bit ADC series, see the *References* section.

Figure 1 shows the block diagram of the MSP430 14-bit ADC.

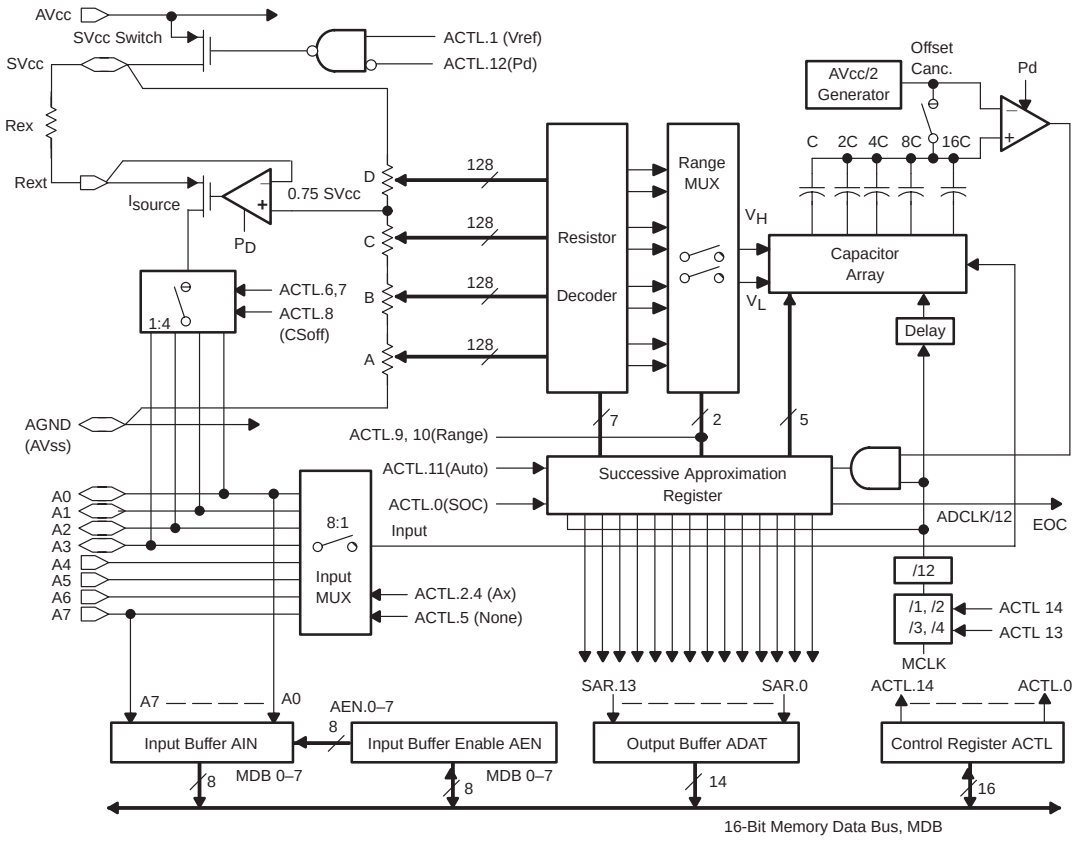

Figure 1. Hardware of the 14-Bit ADC

1.1 Characteristics of the 14-Bit ADC

- Monotonic over the complete ADC range
- Eight analog inputs; may be switched individually to digital input mode
- Programmable current source on four analog inputs. Independent of the selected conversion input: current source output and ADC input pins may be different
- Relative (ratiometric) or absolute measurement possible
- Sample and hold function with defined sampling time
- End-of-conversion flag usable with interrupt or polling
- Last conversion result is stored until start of next conversion
- Low power consumption and possibility to power down the peripheral
- Interrupt mode without CPU processing possible
- Programmable 12-bit or 14-bit resolution
- Four programmable ranges (one quarter of SVcc each)
- Fast conversion time
- Four clock adaptations possible (MCLK, MCLK/2, MCLK/3, MCLK/4)
- Internal and external reference supply possible
- Large supply voltage range

2 ADC Function and Modes

The MSP430 14-bit ADC has two range modes and two measurement modes.

The two range modes are:

- *14-bit mode:* The ADC converts the input range from AVss to SVcc. The ADC automatically searches for one of the four ADC ranges (A, B, C, or D) that is appropriate for the input voltage to be measured.
- *12-bit mode:* The ADC uses only one of the four ranges (A, B, C, or D). The range is fixed by software. Each range covers a quarter of the voltage at the SVcc terminal. This conversion mode is used if the voltage range of the input signal is known.

The two measurement modes are:

- *Ratiometric mode:* A value is measured as a ratio to other values, independent of the actual SVcc voltage.
- *Absolute mode:* A value is measured as an absolute value.

Figure 2 shows different methods to connect analog signals to the MSP430 ADC. The methods shown are valid for the 12-bit and 14-bit conversion modes:

1. Current supply for resistive sensors — Rsens1 at analog input A0
2. Voltage supply for resistive sensors — Rsens2 at analog input A1
3. Direct connection of input signals — Vin at analog input A2
4. Four-wire circuitry with current supply — Rsens3 at output A3 and inputs A4 and A5
5. Reference diode with voltage supply — Dr1 at analog input A6
6. Reference diode with current supply — Dr2 at analog input A7

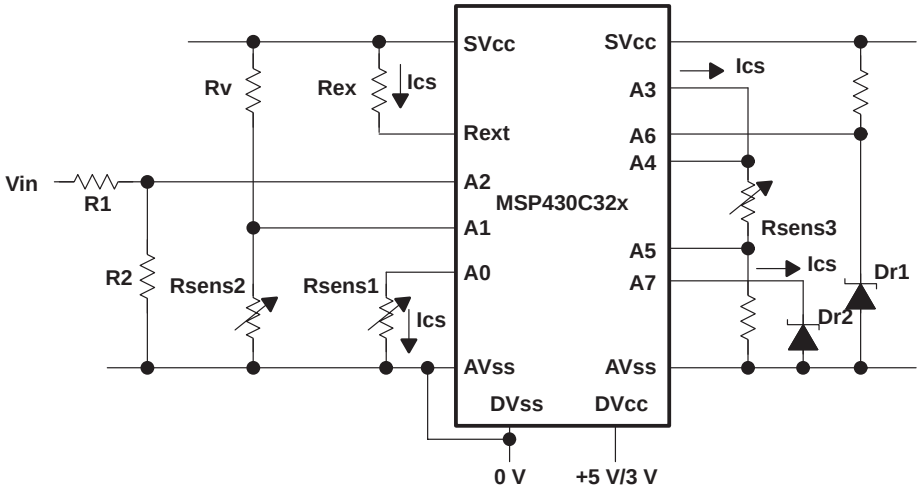

Figure 2. Possible Connections to the Analog-to-Digital Converter

The calculation formulas for all connection methods shown in Figure 2 are explained in the application report, *Application Basics for the MSP430 14-Bit ADC* (SLAA046). [3]

2.1 Function of the ADC

See Figures 1, 9, and 12 for this explanation. The full range of the ADC is made by 4×128 equal resistors connected between the SVcc pin and the AVss (AGND) pin. Setting the conversion-start (SOC) bit in the ACTL control register activates the ADC clock for a new conversion to begin.

The normal ADC sequence starts with the definition of the next conversion; this is done by setting the bits in the ACTL control register with a single instruction. The power-down (PD) bit is set to zero; the SOC bit is not changed by this instruction. After a minimum 6-µs delay to allow the ADC hardware to settle, the SOC bit may be set. The ADC clock starts after the SOC bit is set, and a new conversion starts.

- If the 12-bit mode is selected (RNGAUTO = 0) then a 12-bit conversion starts in a fixed range (A, B, C or D) selected by the bits ACTL.9 to ACTL.10.
- If the 14-bit mode is selected (RNGAUTO = 1), a sample is taken from the selected input Ax that is used only for the range decision. The found range is fixed afterwards – it delivers the two MSBs of the result – and the conversion continues like the 12-bit conversion. This first decision is made by the block range MUX.

This first step fixes the range and therefore the 2 MSBs. Each range contains a block of 128 resistors.

To obtain the 12 LSBs, a sample is taken from the selected input Ax and is used for the conversion. The 12-bit conversion consists of two steps:

- The seven MSBs are found by a successive approximation using the block resistor decode. The sampled input voltage is compared to the voltages generated by the fixed 2^7 (128) equally weighted resistors connected in series. The resistor whose leg voltages are closest to the sampled input voltage-which means between the two leg voltages—is connected to the capacitor array (see Figure 1).
- The five LSBs are found by a successive approximation process using the block capacitor array. The voltage across the selected resistor (the sampled voltage lies between the voltages at the two legs of the resistor) is divided into 2^5 (32) steps and compared to the sampled voltage.

After these three sequences, a 14-bit respective 12-bit result is available in the register ADAT.

Figure 3 shows where the result bits of an analog-to-digital conversion come from:

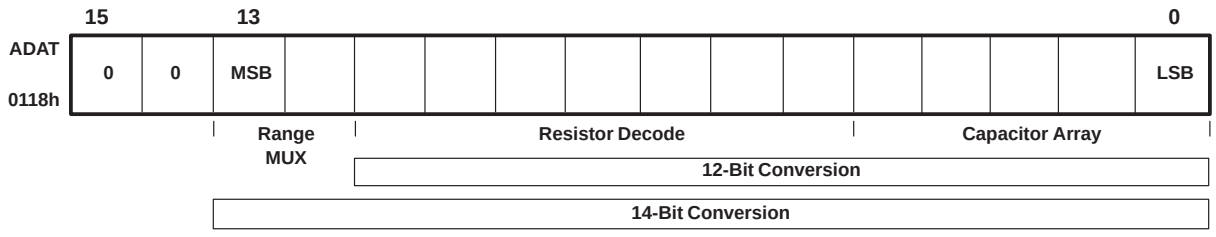

Figure 3. Sources of the Conversion Result

NOTE: The result of the 12-bit conversion does not contain range information: the result bits 12 and 13 are both zero. If these two bits are necessary for the calculation, they need to be inserted by software e.g. 2000h for range C.

2.1.1 ADC Timing Restrictions

To get the full accuracy for the ADC measurements, some timing restrictions need to be considered:

- If the ADCLK frequency is chosen too high, an accurate 14- or 12-bit conversion cannot be assured. This is due to the internal time constants of the sampling analog input and conversion network. The ADC is still functional, but the conversion results show a higher noise level (larger bandwidth of results for the same input signal) with higher conversion frequencies.
- If the ADCLK frequency is chosen too low, then an accurate 14- or 12-bit conversion cannot be assured due to charge losses within the capacitor array of the ADC. This remains true even if the input signal is constant during the sampling time.
- After the ADC module has been activated by resetting the power-down bit, at least 6 µs (power-up time in Figure 9) must elapse before a conversion is started. This is necessary to allow the internal biases to settle. This power-up time is automatically ensured for MCLK frequencies up to 2.5 MHz if the measurement is started the usual way: by separation of the definition and the start of the measurement inside of the subroutine:

```
MOV    #xxx,&ACTL    ; Define ADC measurement
CALL   #MEASR        ; Start measurement with SOC=1
...                  ; ADC result in ADAT
```

If higher MCLK frequencies are used, then a delay needs to be inserted between the definition and the start of the measurement. See the source of the MEASR subroutine in section 2.2.2. The number *n* of additional delay cycles (MCLK cycles) needed is:

$$n \geq (6\,\mu s \times MCLK) - 15$$

- If the input voltage changes very fast, then the range sample and the conversion sample may be captured in different ranges. See section 2.2.1 if this cannot be tolerated. For applications like an electricity meter, this doesn't matter: the error occurs as often for the increasing voltage as for the decreasing voltage so the resulting error is zero.
- After the start of a conversion, no modification of the ACTL register is allowed until the conversion is complete. Otherwise the ADC result will be invalid.

The previously described timing errors lead to spikes in the ADC characteristic: the ADC seems to get caught at certain steps of the ADC. This is not an ADC error; the reasons are violations of the ADC timing restrictions. See Figure 4. The x-axis shows the range A from step 0 to step 4096, the y-axis shows the ADC error (steps).

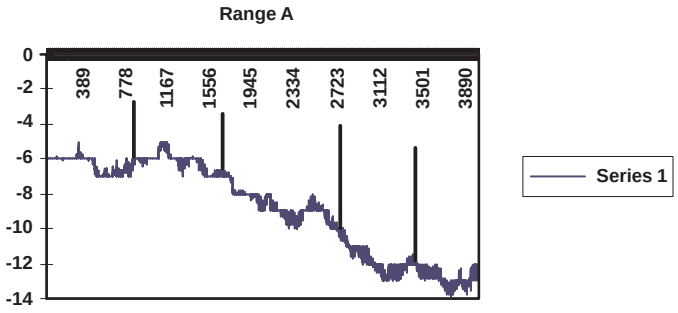

Figure 4. ADC Spikes Due to Violated Timing Restrictions

The ADC always runs at a clock rate set to one twelfth of the selected ADCLK. The frequency of the ADCLK should be chosen to meet the conversion time defined in the electrical characteristics (see data sheet). The correct frequency for the ADCLK can be selected by two bits (ADCLK) in the control register ACTL. The MCLK clock signal is then divided by a factor of 1, 2, 3, or 4. See Section 3.5.

2.1.2 Sample and Hold

The sampling of the ADC input takes 12 ADCLK cycles; this means the sampling gate is open during this time (12 μs at1 MHz). The sampling time is identical for the range decision sample and the data conversion sample.

The input circuitry of an ADC input pin, Ax, can be seen simplified as an RC low pass filter during the sampling period (12/ADCLK): 2 kΩ in series with 42 pF. The 42-pF capacitor (the sample-and-hold capacitor) must be charged during the 12 ADCLK cycles to (nearly) the final voltage value to be measured, or to within 2^{-14} of this value.

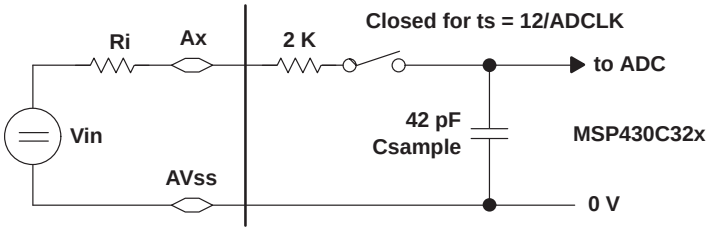

Figure 5. Simplified Input Circuitry for Signal Sampling

The sample time limits the internal resistance, *Ri*, of the source to be measured:

$$(Ri + 2\ k\Omega) \times 42\ pF\ < \frac{12}{ln\left(2^{14}\right) \times ADCLK}$$

Solved for *Ri* with ADCLK = 1 MHz this results in:

$$Ri < 27.4\ k\Omega$$

This means, for the full resolution of the ADC, the internal resistance of the input signal must be lower than 27.4 kΩ.

If a resolution of n bits is sufficient, then the internal resistance of the ADC input source can be higher:

$$Ri < \frac{12}{ln(2^n) \times 42\ pF \times ADCLK} - 2\ k\Omega$$

For example, to get a resolution of 13 bits with ADCLK = 1 MHz, the maximum *Ri* of the input signal is:

$$Ri < \frac{12}{ln(2^{13}) \times 42\ pF \times 10^6} - 2\ k\Omega = 31.7\ k\Omega - 2\ k\Omega = 29.7\ k\Omega$$

To achieve a result with 13 bit-resolution, *Ri* must be lower than 29.7 kΩ.

2.1.3 Absolute and Relative Measurements

The 14-bit ADC hardware allows absolute and relative modes of measurement.

2.1.3.1 Relative Measurements

As Figure 6 shows, relative measurements use resistances (sensors) that are independent of the supply voltage. This is the typical way to use the ADC. The advantage is independence from the supply voltage; it does not matter if the battery is new (Vcc = 3.6 V) or if it has reached the end of life (Vcc = 2.5 V).

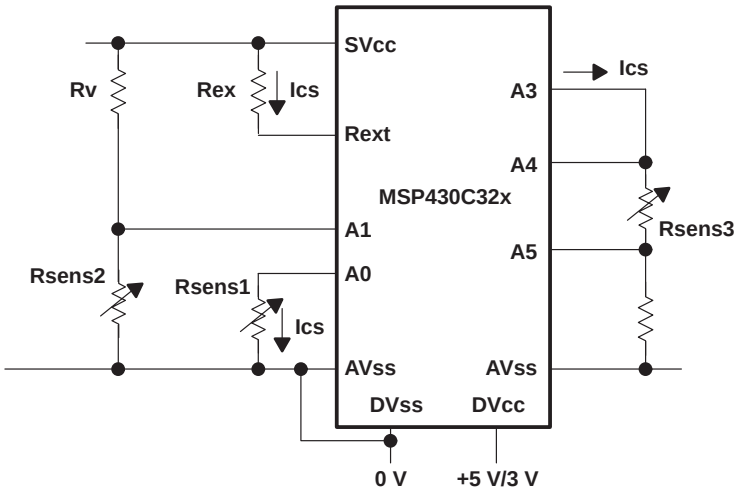

Figure 6. Relative Measurements With the MSP430C32x

2.1.3.2 Absolute Measurements

As Figure 7 shows, absolute measurements measure voltages and currents. The reference used for the conversion is the voltage applied to the SVcc terminal, regardless of whether an external reference is used or if SVcc is connected to AVcc internally. An external reference is necessary if the supply voltage AVcc (the normal reference) cannot be used for reference purposes, for example a battery supply.

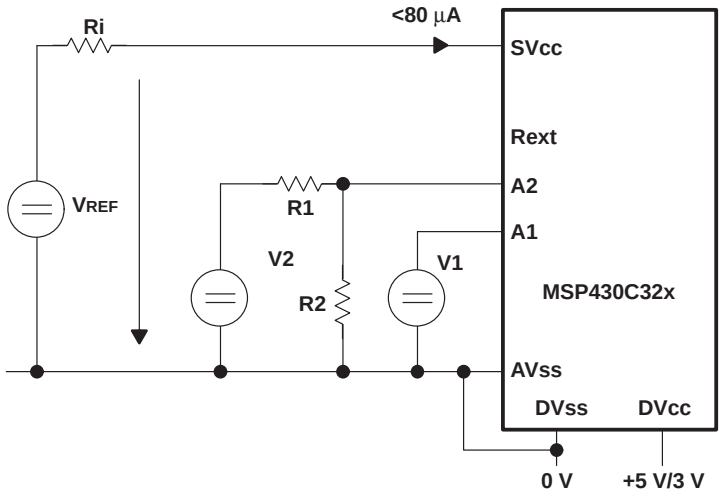

Figure 7. Absolute Measurements Using External Reference Voltage

2.2 Using the ADC in 14-Bit Mode

The 14-bit mode is used if the range of the input voltage exceeds one ADC range. The total input signal range is from analog ground (AVss) to the voltage at SVcc (external reference voltage or AVcc).

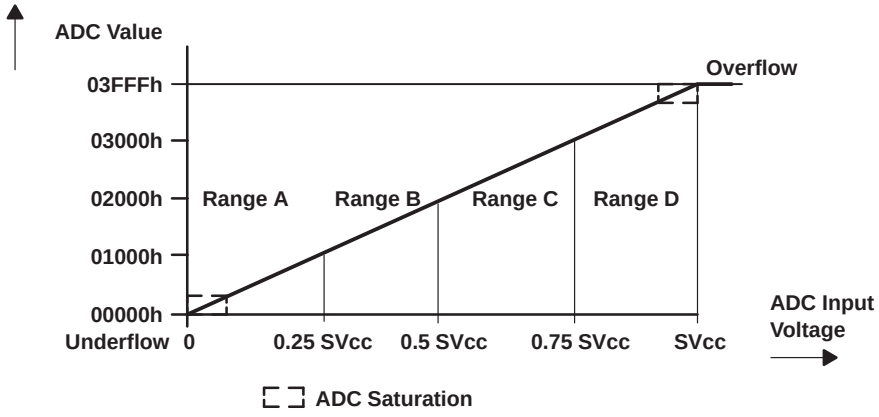

Figure 8. Complete 14-Bit ADC Range

The dashed boxes at the AVss and SVcc voltage levels indicate the saturation areas of the ADC; the measured results are 0h at AVss and 3FFFh at SVcc. The saturation areas are smaller than 10 ADC steps.

The nominal ADC formula for the 14-bit conversion is:

$$N = \frac{VAx}{VREF} \times 2^{14} \rightarrow VAx = \frac{N \times VREF}{2^{14}}$$

Where:

N = 14-bit result of the ADC conversion
VAx = Input voltage at the selected analog input Ax [V]
VREF = Voltage at pin SVcc (external reference or internal AVcc) [V]

2.2.1 Timing

The two ADCLK bits (ACTL.13 and ACTL.14) in the ACTL control register are used to select the ADCLK frequency best suited for the ADC. The MCLK clock signal can be divided by a factor 1, 2, 3, or 4 to get the best suited ADCLK.

Using the autorange mode (RNGAUTO/ACTL.11 = 1) executes a 14-bit conversion. The selected analog input signal at input Ax is sampled twice. The range decision is made after the first sampling of the input signal; the 12-bit conversion is made after the second sampling. Both samplings are 12 ADCLK cycles in length. Altogether the 14-bit conversion takes 132 ADCLK cycles. See Figure 9 for timing details.

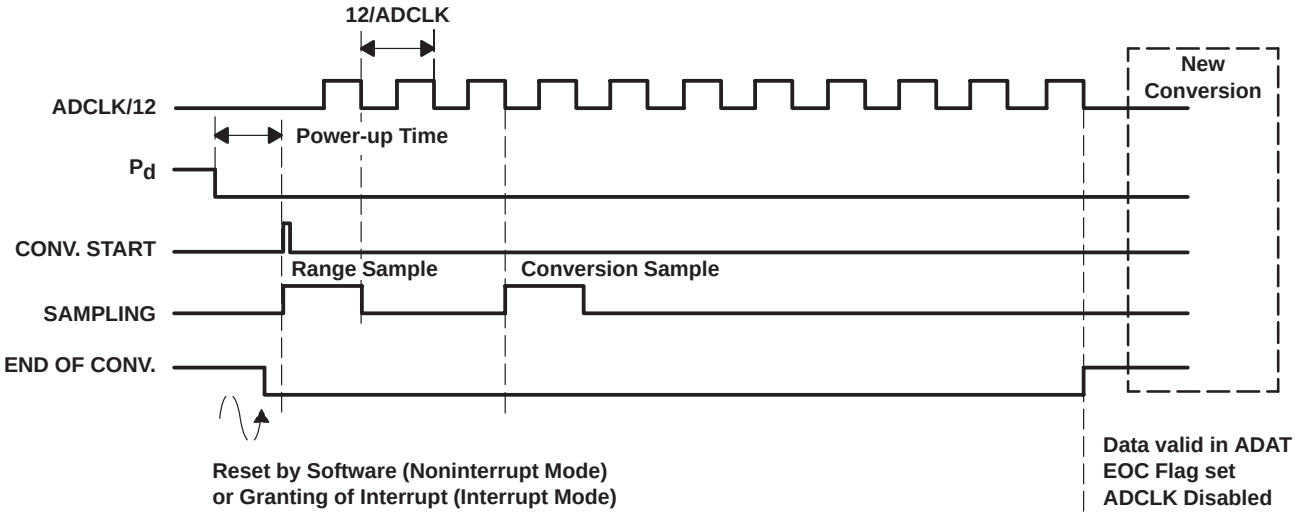

Figure 9. Timing for the 14-bit Analog-to-Digital Conversion

The input signal must be valid and steady during this sampling period to obtain an accurate conversion. It is also recommended that no activity occur during the conversion at analog inputs that are switched to the digital mode.

If the input voltage to the ADC changes during the measurement, it is possible for the range decision sample to be taken in a different ADC range than the conversion sample. The result of these conditions is saturated values:

- Increasing input voltage: nFFFh with range n = 0...2
- Decreasing input voltage: n000h with range n = 1...3

The saturated result is the best possible result under this circumstance: an analog input that changes from 2FF0h to 3020h during the sampling period delivers the saturated result 2FFFh and not 2000h.

The following software sequence can be used to check the result of an A/D conversion if the two samples (range and conversion) were taken in different ranges. If this is the case, the measurement is repeated.

```
LM  MOV    #xxx,&ACTL       ; Define measurement
    CALL   #MEASR           ; Measure ADC input
    MOV    &ADAT,R5         ; Copy ADC result
    AND    #0FFFh,R5        ; 12 LSBs stay
```

```
    JZ     LM              ; Yes, ADC value too high (n000h)
    CMP    #0FFFh,R5       ; Bits 11 to 0 all 1s?
    JEQ    LM              ; Yes, ADC value too low  (nFFFh)
    ...                    ; Both samples taken in same range
```

2.2.2 Software Example

The often-used measurement subroutine MEASR is shown below. It contains all necessary instructions for a measurement that uses polling for the completion check. The subroutine assumes a preset ACTL register; all bits except the SOC bit must be defined before the setting of the SOC bit. The subroutine may be used for 12-bit and 14-bit conversions. Up to an MCLK frequency of 2.5 MHz no additional delays are necessary to ensure the power-up time.

```
; ADC measurement subroutine.
; Call:   MOV    #xxx,&ACTL    ; Define ADC measurement. Pd=0
;         CALL   #MEASR        ; Measure with ADC
;         BIS    #PD,&ACTL     ; Power down the ADC
;  ...                         ; ADC result in ADAT
;
MEASR     BIC.B  #ADIFG,&IFG2  ; Clear EOC flag
          ...                  ; Insert delays here (NOPs)
          BIS    #SOC,&ACTL    ; Start measurement
M0        BIT.B  #ADIFG,&IFG2  ; Conversion completed?
          JZ     M0            ; No
          RET                  ; Result in ADAT
```

2.3 Using the ADC in 12-Bit Mode

The following mode is used if the range of the input voltage is known. If, for example, a temperature sensor is used whose signal range always fits into one range (for example range B), then the 12-bit mode is the right selection. The measurement time with MCLK = 1 MHz is only 96 µs compared with 132 µs if the autorange mode is used. Figure 10 shows the four ranges compared to the voltage at SVcc. The possible ways to connect sensors to the MSP430 are the same as shown for the 14-bit ADC in Figure 2.

This mode should be used only if the signal range is known and the saved 36 ADCLK cycles are a real advantage.

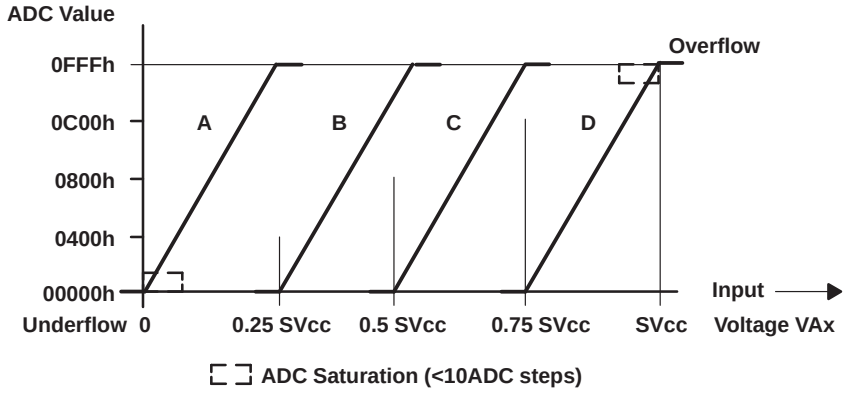

Figure 10. The Four 12-Bit ADC Ranges A to D

NOTE: The ADC results 0000h and 0FFFh mean underflow and overflow: the voltage at the measured analog input is below or above the limits of the programmed range.

All of the formulas given for the 12-bit mode assume a faultless conversion result N:

0 < N < 0FFFh

If underflow or overflow are not checked, erroneous calculation results occur.

Figure 11 shows how any of the four ADC ranges appears to the software:

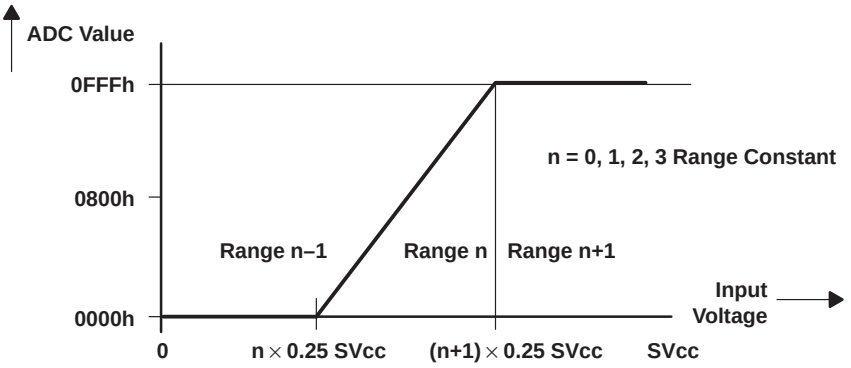

Figure 11. Single 12-Bit ADC Range

The nominal ADC formula for the 12-bit conversion is:

$$N = \frac{VAx - (n \times 0.25 \times VREF)}{VREF} \times 2^{14} \rightarrow VAx = VREF \times \left(\frac{N}{2^{14}} + n \times 0.25\right)$$

Where:

N = 12-bit result of the ADC conversion
VAx = Input voltage at the selected analog input Ax [V]
VREF = Voltage at pin SVcc (external reference or internal AVcc) [V]
n = Range constant (n = 0, 1,2, 3 for ranges A, B, C, D)

To get the 14-bit equivalent N_{14} of a 12-bit ADC result N_{12}, the following formula may be used:

$$N_{14} = N_{12} + n \times 1000h$$

To check if the result of a 12-bit A/D conversion is correct, the following software sequence can be used:

```
MOV    #xxx,&ACTL          ; Define measurement
CALL   #MEASR              ; Measure ADC input
TST    &ADAT               ; Check if underflow (000h)
JZ     UFL                 ; Underflow: go to error handling
CMP    #0FFFh,&ADAT        ; Check if overflow (0FFFh)
JEQ    OFL                 ; Overflow: go to error handling
...                        ; Result is correct: use ADAT
```

2.3.1 Timing

The two ADCLK bits (ACTL.13 and ACTL.14) in the ACTL control register are used to select the ADCLK frequency best suited for the ADC. The MCLK clock signal can be divided by a factor 1, 2, 3, or 4 to get the best suited ADCLK.

Disabling the autorange mode (RNGAUTO/ACTL.11 = 0) executes a 12-bit conversion; the range defined by the ACTL.10 and ACTL.9 bits is used. The selected analog input signal at input Ax is sampled once; after the sampling, the 12-bit conversion is executed. The 12-bit conversion takes 96 ADCLK cycles. See Figure 12 for timing details.

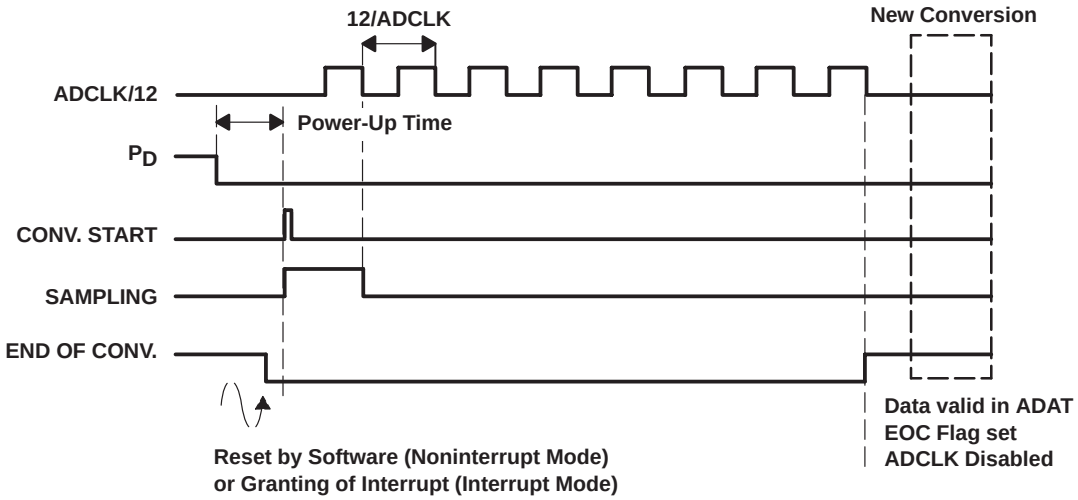

Figure 12. Timing for the 12-Bit A/D Conversion

2.3.2 Software Example

The measurement of Rsens1 is shown in Figure 2. With MCLK = 2.2 MHz, the result is always located in range B. The result must be converted to a 14-bit value to be used by software routines written for 14-bit results.

```
MOV    #ADCLK2+RNGB+CSA0+A0+VREF,&ACTL   ; Define ADC
CALL   #MEASR                          ; Measure with ADC
MOV    &ADAT,R5                        ; 12-bit ADC result to R5
ADD    #1000h,R5                       ; Add start address of range B
...                                    ; 14-bit value in R5
```

3 The A/D Controller Hardware

Paragraph 2 describes the analog-to-digital conversion. The mnemonics used are defined in the appendix.

3.1 ADC Control Registers

The ADC control registers are in the MSP430 memory area where only word addresses are possible. This means that all registers are word-structured and should be accessed by word instructions only. Byte addressing results in a nonpredictable operation.

The access description below the register bits has the following meaning:

- rw-0 read/write bit, reset after power-up clear (PUC)
- rw-1 read/write bit, set after power-up clear
- r0 read as zero
- r read only
- (w)r0 write only. Writing generates a pulse, no reset necessary. Read as zero

3.1.1 ACTL Control Register

The ACTL control register is the main register for programming the ADC. Its content (shown in Figure 13) defines the current operation. All of the bits should be changed only after a completed conversion. Otherwise a faulty result will occur.

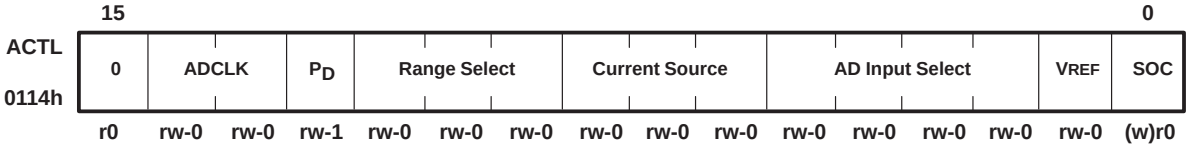

Figure 13. ACTL Control Register

3.1.1.1 Conversion Start (SOC)

The SOC write-only bit (see Figure 14) starts an analog-to-digital conversion. The conditions of the measurement are defined with the other bits of the ACTL register. It is not necessary to reset the bit. The SOC bit is always read as a zero.

Figure 14. Conversion Start (SOC)

EXAMPLE: start the ADC as defined by the content of the ACTL register.

```
MOV    #ADCLK2+RNGA+CSOFF+A0+VREF,&ACTL  ; Define ADC
...                                      ; Delay 6us to allow settling
BIS    #SOC,&ACTL                        ; Start ADC conversion
```

3.1.1.2 Voltage Reference Bit (VREF)

The VREF bit (see Figure 15) defines whether an internal or an external reference voltage is used for the A/D conversion.

Figure 15. Voltage Reference Bit (VREF)

VREF = 0: External reference. The transistor between AVcc and SVcc is switched off. The SVcc terminal is an input pin for an external reference voltage. The external reference source must be able to supply a current up to 80 μA. The voltage range for the external reference is AVcc/2 ≤ VREF ≤ AVcc.

VREF = 1: Internal reference. The transistor between AVcc and SVcc is switched on: the SVcc output terminal is connected to the analog supply voltage AVcc. No external voltage should be supplied to the SVcc terminal.

EXAMPLE: define an A/D conversion with the internal reference voltage AVcc.

```
MOV     #ADCLK2+RNGAUTO+CSOFF+A0+VREF,&ACTL
```

Start an A/D conversion with an external reference connected to the SVcc terminal (VREF = 0).

```
MOV     #ADCLK2+RNGAUTO+CSOFF+A0,&ACTL    ; Vref = 0
CALL    #MEASR                            ; Start the measurement
```

3.1.1.3 ADC Input Select Bits

The four ADC input select bits (see Figure 16) define which of the possible eight analog inputs is selected for the A/D conversion.

Figure 16. ADC Input Selection Bits

Table 2 lists the possible ADC input selections.

Table 1. ADC Input Selection Bits

INPUT SELECTION CODE	MNEMONIC	SELECTED ANALOG INPUT	COMMENT
0	A0	A0	Signal at the pin A0 is selected
1	A1	A1	Signal at the pin A1 is selected
2	A2	A2	Signal at the pin A2 is selected
3	A3	A3	Signal at the pin A3 is selected
4	A4	A4	Signal at the pin A4 is selected
5	A5	A5	Signal at the pin A5 is selected
6	–	A6	Not implemented with the MSP430C32x
7	–	A7	Not implemented with the MSP430C32x
8–15	–	None	No analog input is selected

EXAMPLE: to start an ADC conversion for analog input A3 with unchanged conditions:

```
BIC     #03Ch+PD,&ACTL          ; Reset all input select bits
...                             ; 6 μs delay
BIS     #A3+SOC,&ACTL           ; Start conversion for A3
```

3.1.1.4 Current Source Output Select Bits

The three current source output select bits (see Figure 17) define the analog input Ax where the output current of the current source is switched. To switch the current source off, ACTL.8 (CSOFF) is set to one.

Figure 17. Current Source Output Select Bits

Table 2. Current Source Output Select Bits

OUTPUT SELECTION CODE	MNEMONIC	SELECTED CURRENT OUTPUT	COMMENT
0	CSA0	A0	Current source connected to pin A0
1	CSA1	A1	Current source connected to pin A1
2	CSA2	A2	Current source connected to pin A2
3	CSA3	A3	Current source connected to pin A3
4–7	CSOFF	Off	Current source switched off

EXAMPLE: connect the current source to pin A3, and start measurement at pin A4. All other ADC conditions stay unchanged. This example refers to the hardware configuration for Rsens3 shown in Figure 2.

```
BIC    #01FCh+PD,&ACTL        ; Reset SOC and input sel. Bits
       ...                     ;6 µs delay
BIS    #CSA3+A4+SOC,&ACTL     ; Start conversion for A4
```

3.1.1.5 Range Selection Bits

The three range select bits (see Figure 18) define the ADC range that is used for the conversion.

Figure 18. Range Select Bits

Table 3. Range Select Bits

RANGE SELECTION CODE	MNEMONIC	SELECTED RANGE	COMMENT
0	RNGA	A	0 to 0.25 × SVcc
1	RNGB	B	0.25 to 0.5 × SVcc
2	RNGC	C	0.5 to 0.75 × SVcc
3	RNGD	D	0.75 × SVcc to SVcc
4–7	RNGAUTO	A, B, C, D	Automatic range select

EXAMPLE: prepare the ACTL register for measurement of analog input A3 using the internal reference, and with the current source connected to A3 and fixed to range B.

```
MOV    #RNGB+CSA3+A3+VREF,&ACTL
```

3.1.1.6 Power Down Bit (Pd)

The power-down bit (see Figure 19) reduces the power consumption of the ADC to the lowest possible value. It switches off the comparator, the SVcc switch, and the current source.

Figure 19. Power Down Bit (Pd)

Pd = 0: The ADC is switched on.

Pd = 1: The SVcc switch is off, the comparator is powered down and the current source is off. This ensures the minimum current consumption for the ADC.

EXAMPLE: power down the ADC for minimum current consumption.

```
BIS #PD,&ACTL ; Power down the ADC
```

3.1.1.7 Clock Frequency Selection Bits

The two clock frequency selection bits (see Figure 20) select the optimum clock frequency for the ADC. This is necessary due to the relatively low maximum ADCLK frequency (1.5 MHz) compared to the maximum MCLK frequency (3.3 MHz).

Figure 20. Clock Frequency Selection Bits

Table 4. Clock Frequency Selection Bits

SELECTION CODE	MNEMONIC	DIVISION FACTOR	ADCLK FREQUENCY	COMMENT
0	ADCLK1	1	MCLK	MCLK ≤ 1.5 MHz
1	ADCLK2	2	MCLK/2	MCLK > 1.5 MHz
2	ADCLK3	3	MCLK/3	MCLK > 3.0 MHz
3	ADCLK4	4	MCLK/4	(MCLK > 4.5 MHz)

EXAMPLE: For MCLK = 2.5 MHz, the highest possible ADCLK frequency (1.25 MHz) is set.

```
MOV #ADCLK2+RNGAUTO+A3+VREF,&ACTL
```

3.1.1.8 Bit 15

Bit 15 (see Figure 21) should always be set to zero to maintain software compatibility with future versions of the ADC.

Figure 21. Bit 15

3.1.2 A/D Data Register ADAT

The ADC data register ADAT contains the result of the last A/D conversion. The conversion data is valid in the ADAT register at the end of a conversion and stays valid until another A/D conversion is started with the setting of the SOC bit (ACTL.0). The read-only structure of the ADAT register does not allow read/modify/write instructions like ADD or BIC with the ADAT register used as the destination: only the instructions BIT, TST and CMP may be used this way. With the ADAT register as a source, all instructions may be used.

Figure 22 shows the result of a 12-bit conversion: the value is always between 000h (underflow) and FFFh (overflow), independent of the ADC range used. The missing range information (bits 12 and 13) must be added by the software.

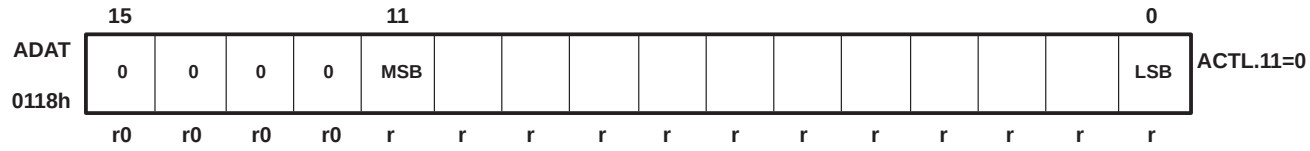

Figure 22. The Data Register ADAT, 12-Bit A/D Conversion

Figure 23 shows the result of a 14-bit conversion: the value is between 0000h (underflow) and 3FFFh (overflow). Result bits 13 and 12 indicate the range of the result:

- 00 Range A 0 to 0.25 × SVcc
- 01 Range B 0.25 × SVcc to 0.50 × SVcc
- 10 Range C 0.50 × SVcc to 0.75 × SVcc
- 11 Range D 0.75 × SVcc to SVcc

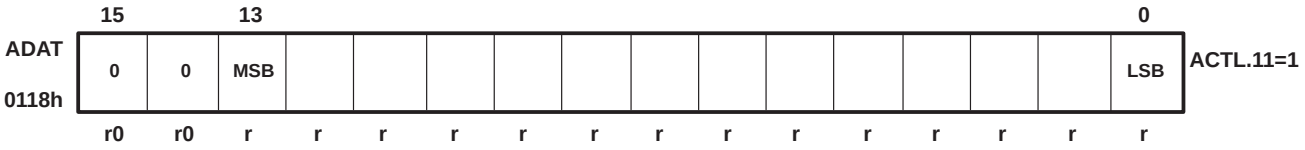

Figure 23. Data Register ADAT, 14-Bit A/D Conversion

To read the result of the last conversion, use a simple MOV instruction:

```
MOV&ADAT,R5     ; Copy the ADC result to R5
...             ; A new conversion may begin
```

3.1.3 Input Register AIN

Input register AIN (see Figure 24) is a read-only word register; however, only the low byte of the register is implemented. The same access restrictions are valid as described for the ADAT register. AIN.0 to AIN.7 correspond to the input terminals A0 to A7. The high byte of the register is read as 00h. Input register AIN shows the digital input information at the input terminals that are switched to the digital mode (AEN.x = 1). The formula for the bit AIN.x is:

$$AIN.x = Ax \ .and. \ AEN.x$$

Where:

AIN.x = Bit x of the input register AIN
Ax = Logic level at the analog input Ax
AEN.x = Bit x of the input enable register AEN

This means, that analog inputs (AEN.x = 0) are read as zero.

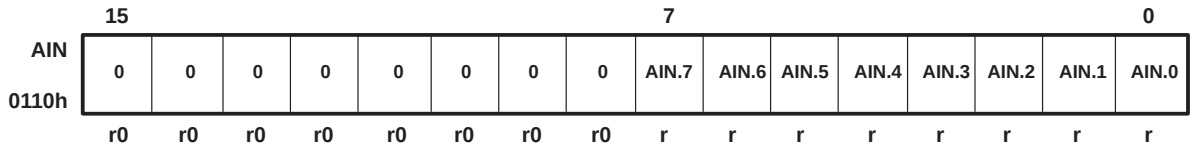

Figure 24. Input Register AIN

EXAMPLE: The A5 input terminal is used as a digital input. Test if this input is high; if yes, jump to label A5HI:

```
INITA5  BIS   #20h,&AEN     ; Use pin A5 as digital input
        ...
        BIT   #020h,&AIN    ; Pin A5 high?
        JNZ   A5HI          ; Yes, goto A5HI
        ...                 ; No, A5 is low
```

NOTE: Only digital inputs with very low activity or controlled access (e.g. keyboard scan) should be connected to inputs A0 to A7. Otherwise, this activity influences the measurement results of the analog inputs.

3.1.4 Input Enable Register AEN

Input enable register AEN (see Figure 25) is a read/write word. However, only the low byte of the register is implemented. AEN.0 to AEN.7 correspond to input terminals A0 to A7. The high byte of the register is read as 00h. The initial state of all bits is reset.

	15								7							0
AEN	0	0	0	0	0	0	0	0	AEN.7	AEN.6	AEN.5	AEN.4	AEN.3	AEN.2	AEN.1	AEN.0
0112h	r0	r0	r0	r0	r0	r0	r0	r0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0

Figure 25. Input Enable Register AEN

Input enable register bits AEN.x control the function of input pins A0 to A7:

AEN.x = 0: Input terminal Ax is used as an analog input. Bit AIN.x is read as zero.

AEN.x = 1: Digital input. The bit read in the AIN register represents the logical level at the appropriate Ax terminal.

EXAMPLE: The A5 and A4 input terminals are used as digital inputs. An application is given with the AIN terminal example.

```
BIS    #030h,&AEN          ; Pin A5 and A4 digital inputs
```

3.2 Current Source

A stable, programmable current source is available at the four analog inputs A0 to A3. With programming resistor Rex between terminals SVcc and Rext, it is possible to get a defined current, Ics, out of the programmed analog input Ax. Ics is directly related to the voltage at SVcc. This allows relative measurements to be made using the current source that are independent from the ADC supply voltage SVcc. The analog input to be measured and the analog input used for the current source are independent of each other; this means that the current source may be programmed to input A3 and the measurement taken from inputs A4 and A5, as shown in Figure 6 for Rsens3.

3.2.1 Normal Use of the Current Source

Figure 26 shows the normal use of the current source: the generated current Ics flows through the addressed analog input A0 and generates a voltage drop Vin at the connected sensor Rsens. This voltage drop Vin is multiplexed to the ADC and measured.

The current Ics defined by the external resistor Rex is:

$$Ics = \frac{0.25 \times Vref}{Rex}$$

The input voltage Vin at the selected analog input with the current Ics and a connected sensor Rsens is:

$$Vin = Rsens \times Ics = Rsens \times \frac{0.25 \times V_{REF}}{Rex} \rightarrow Rsens = \frac{Vin}{Ics}$$

The ADC result N for an input voltage Vin is:

$$N = 2^{14} \times \frac{Vin}{V_{REF}} \rightarrow Vin = \frac{V_{REF} \times N}{2^{14}}$$

The above equations lead to the measured sensor resistance Rsens:

$$Rsens = \frac{Rex}{0.25 \times V_{REF}} \times Vin = \frac{Rex}{0.25 \times V_{REF}} \times \frac{V_{REF}}{2^{14}} \times N = Rex \times N \times 2^{-12}$$

The result N of the A/D conversion is:

$$N = \frac{0.25 \times V_{REF}}{Rex} \times \frac{2^{14}}{V_{REF}} \times Rsens = \frac{Rsens}{Rex} \times 2^{12}$$

Where:

Rex = Resistor between pins SVcc and Rex (defines current Ics) [Ω]
Rsens= Resistor to be measured (connected between Ax and AGND) [Ω]
VREF = Voltage at SVcc. External (VREF = 0) or internal (VREF = 1) [V]

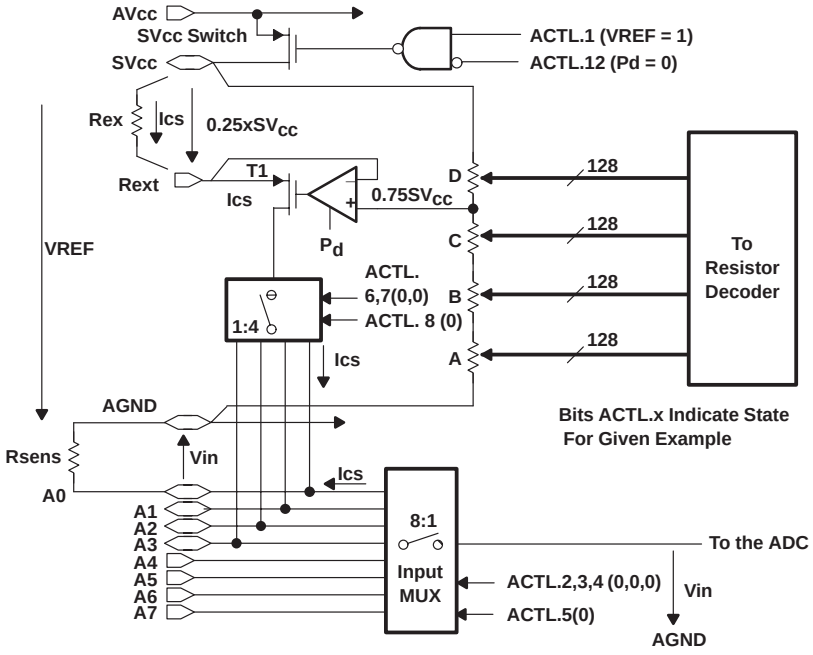

Figure 26. The Current Source

If the 12-bit conversion is used, the above equations change to:

$$N = \frac{\frac{0.25 \times V_{REF}}{Rex} \times Rsens - n \times 0.25 \times V_{REF}}{V_{REF}} \times 2^{14} = \left(\frac{Rsens}{Rex} - n\right) \times 2^{12}$$

This gives, for the unknown resistor Rsens:

$$Rsens = Rex \times \left(\frac{N}{2^{12}} + n\right)$$

The code sequence for the measurement shown in Figure 26 is:

```
MOV    #ADCLK1+RNGA+CSA0+A0+VREF,&ACTL   ; Define ADC
CALL   #MEASR                           ; Measure Rsens at A0
...                                     ; Result in ADAT
```

When using the current source, it is not possible to use the full range of the ADC: only the range defined with *Load Compliance* in the *Electrical Description* is valid (0.5 × SVcc, which means only the ranges A and B). Figures 28 and 29 show the typical error characteristics of the current source at its limit. Figure 28 shows the error characteristic for Vcc = 4.5 V and a relatively high Rex (1 kΩ). It shows that up to a ratio of 0.745 for VA0/SVcc (which means range A, B, and nearly all of range C) the current source works correctly. Then ΔIcs (the difference between the programmed Ics and the real Ics) increases linearly with

$$\frac{\Delta V}{\Delta I} = Rex$$

The reason is saturated transistor T1 of the current source. When T1 is saturated, only the external resistor Rex determines the current Ics. Figure 27 shows the measurement circuitry and an explanation of the error curves. The small dashed box indicates the area that is magnified in Figures 28 and 29.

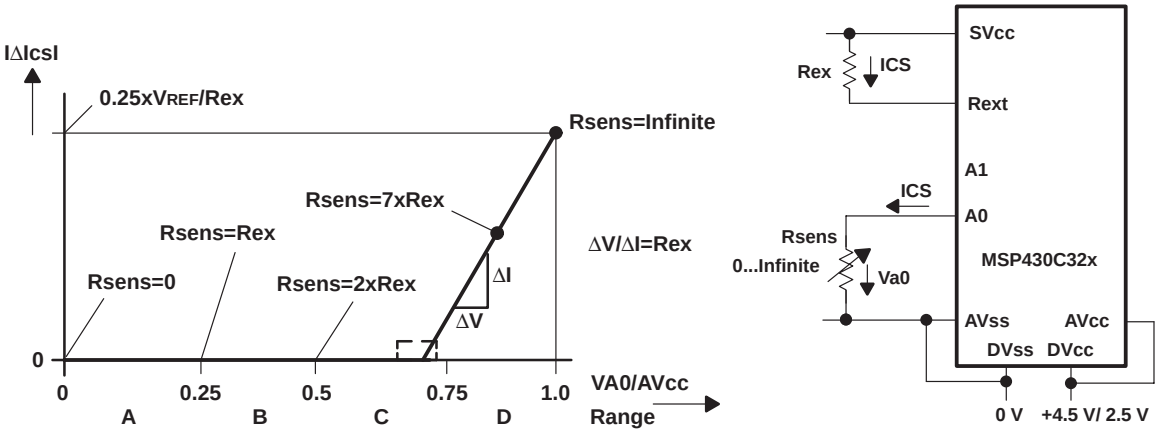

Figure 27. Measurement Circuitry for the Error of the Current Source

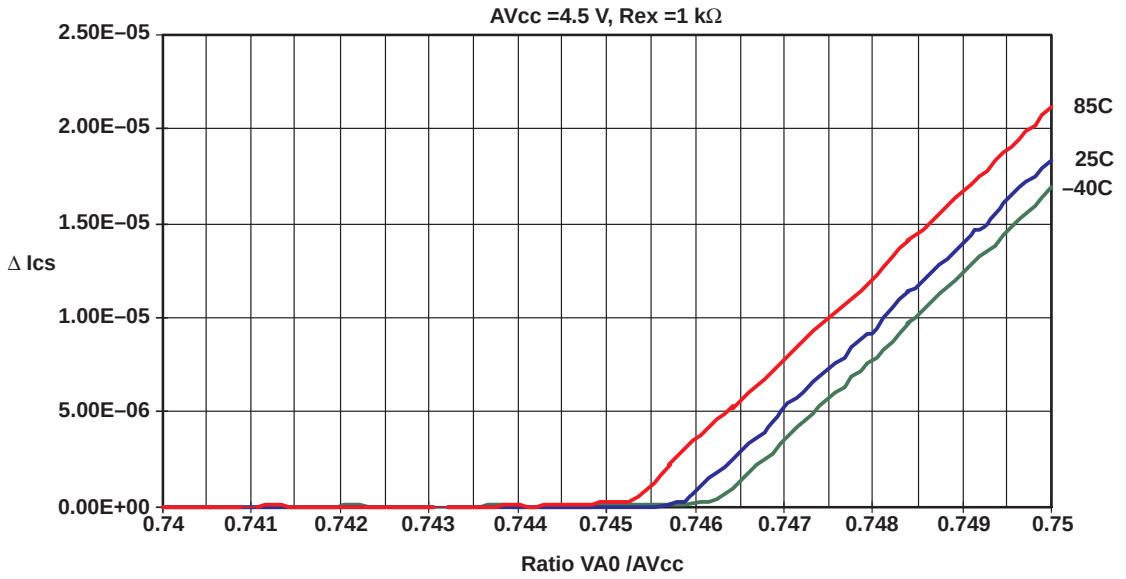

Figure 28. Error of the Current Source at the Limit

Figure 29 gives the characteristic at the other extreme: Vcc = 2.5 V and Rex = 150 Ω. The slope beyond the operation limit of the current source (here at VA0/AVcc = 0.7125) is also:

$$\frac{\Delta V}{\Delta I} = Rex$$

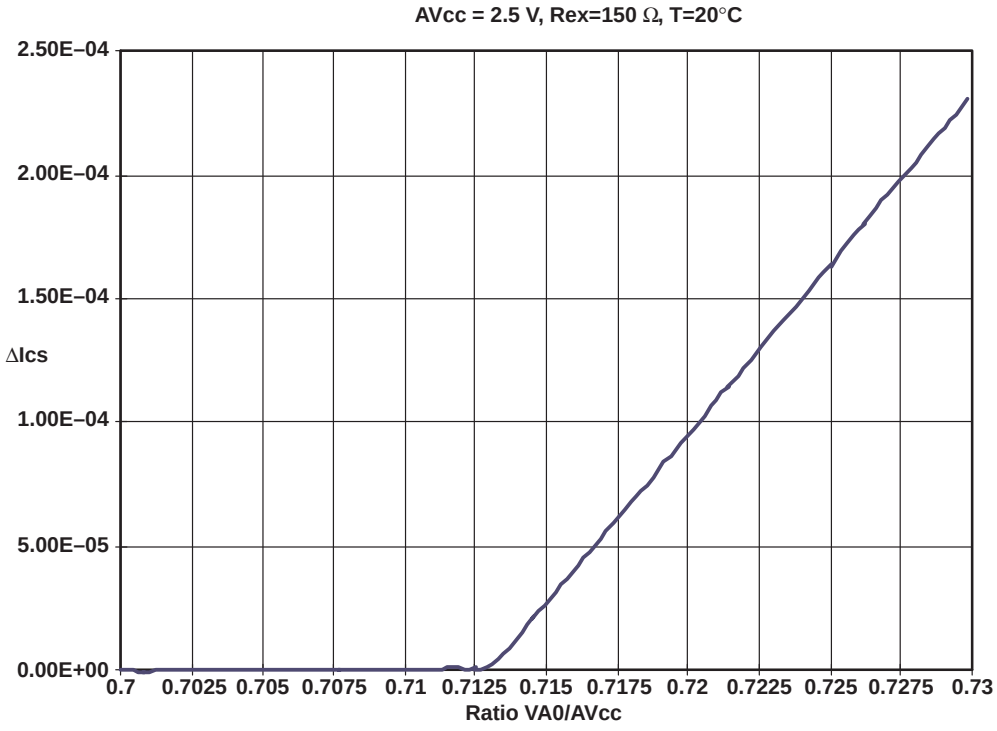

Figure 29. Error of the Current Source at the Limit

The characteristic shown in Figure 29 indicates that the current source works up to 71% of the applied AVcc under worst case conditions; this includes ADC ranges A, B and 84% of range C. If Rex is chosen as 1 kΩ and SVcc is 4.5 V, then the current source works up to a ratio of 0.745, which means it covers nearly 98% of range C.

If the current source is used with an external amplifier (operational amplifier) that amplifies the output signal coming from the current source, then the full range of the ADC can be used with a different ADC input. Figure 30 shows such a circuit. The signal at analog input A0 can use the full range of the A/D converter; the signal at A1 is restricted to the working area of Ics that is shown in Figures 28 and 29.

The equations for the circuitry are explained in *Application Basics for the MSP430 14-Bit ADC Application Report* (SLAA046).[3]

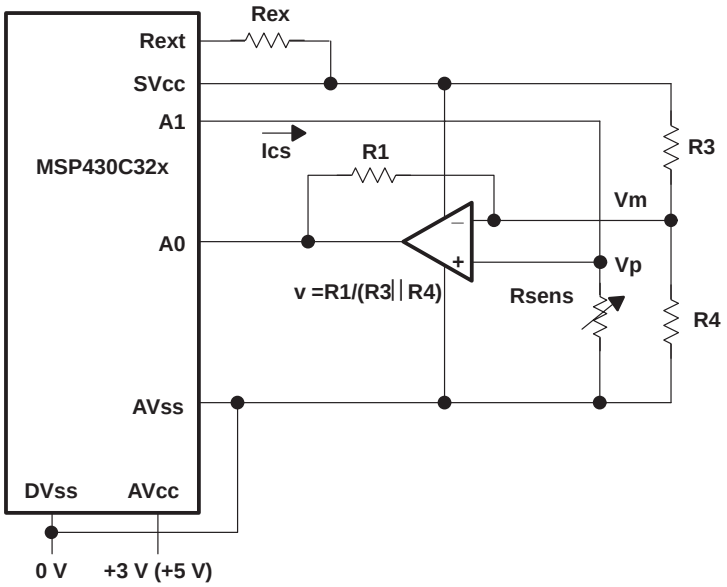

Figure 30. Application of the Current Source With the Full ADC Range at Input A0

3.2.2 Current Source Used for Level Shifting

If analog signals that lie partially or totally outside of the ADC range of the MSP430 (AVss to SVcc), need to be measured then the current source can be used to shift the signal level into the measurable range.

The current transformer, shown on the left in Figure 31, outputs a secondary voltage that is proportional to the primary current, Iac. The signed output voltage (symmetrical to the AVss voltage) is shifted into the middle of the ADC range by a current Ics through the resistor Rsh. This current Ics must be small, due to the sensitivity of current transformers to dc biasing.

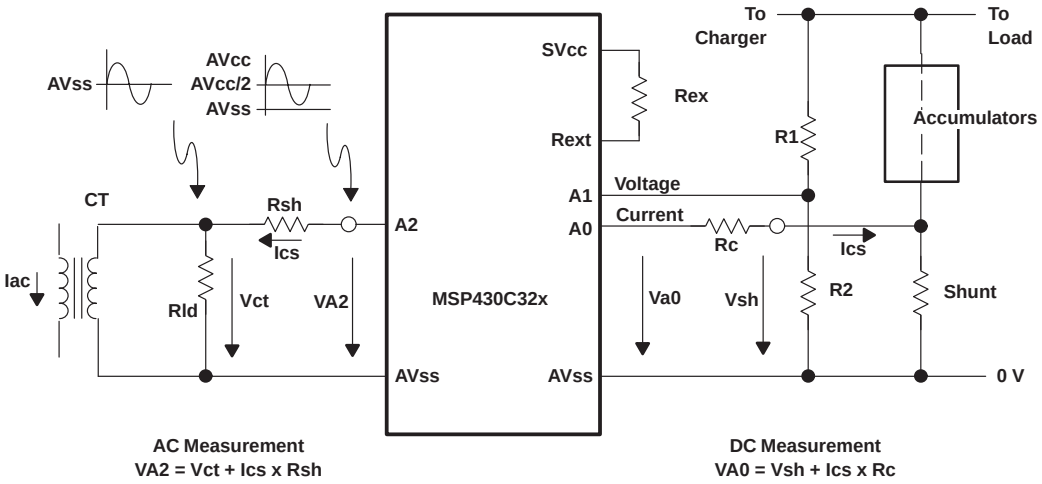

Figure 31. Current Measurement With Level Shifting

The right side of Figure 31 shows the measurement of a signed dc current. Due to the two directions of the accumulator current (charge and discharge current) level shifting is necessary: the charge current generates a positive voltage, Vsh; the discharge current generates a negative voltage, Vsh, at the shunt. The current, Ics, together with resistor Rc, also shifts the voltage drop of the discharge current into the ADC range.

The advantages of level shifting by the current source are:

- Possibility to measure signals that are outside of the ADC range
- Omission of the saturation area near the AVss voltage
- Possible readjustment of the zero current ADC value during periods with no current flow

3.3 SVcc Terminal

The SVcc terminal is the reference for all ADC measurements. The voltage applied to this terminal refers to the result value 2^{14} (16,384), regardless of whether the reference voltage is applied internally or externally (external VREF). The VREF bit located in the ACTL registers defines whether the internal reference AVcc is used (VREF = 1) or an external voltage is used (VREF = 0).

3.3.1 SVcc Terminal Used as an Output for the ADC Reference Voltage

Typically, the SVcc terminal is used to supply the reference and voltage to the ADC circuitry. It can be activated while measurements are being taken and deactivated for low power periods. Figure 32 shows an example of this. All of the sensors connected to the MSP430 are powered by the SVcc terminal.

The SVcc terminal outputs the AVcc voltage if the following conditions are true:

VSVcc = VREF .and. @ Pd .and. VAVcc

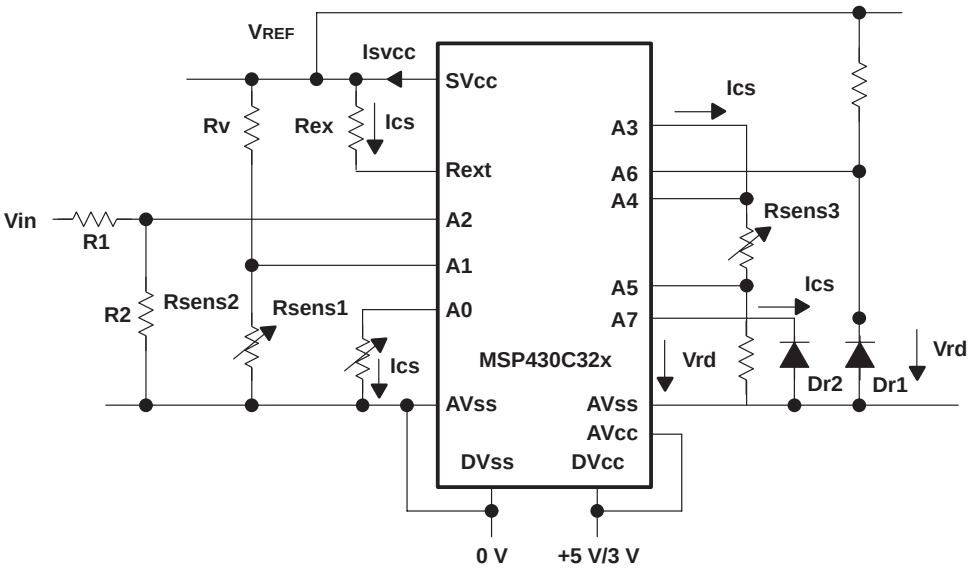

Figure 32. SVcc Terminal Used as an Output

The voltage, Vin, at analog input A2 is measured in comparison to the voltage at SVcc. If the voltage, VREF, at SVcc is known (AVcc is stable and known, ISVcc is small), then Vin can be measured exactly. Otherwise an external reference diode (or equivalent) may be connected to a free analog input, and its voltage, Vrd, is measured. See Figure 32. The formula for Vin is then:

$$Vin = Vrd \times \frac{Nin}{Nrd} \times \frac{R1 + R2}{R2}$$

Where:

Vrd = Voltage of the reference diode [V]
Nin = 14-bit result for Vin
Nrd = 14-bit result for the voltage Vrd of the reference diode

3.3.2 SVcc Terminal Used as an Input for the ADC Reference Voltage

For absolute voltage measurements an external reference voltage, VREF, is necessary (see Figure 33). The sensor measurements for Rsens1 to Rsens3 are made the same way as with the internal reference voltage. The only difference is the VREF bit of the ACTL register: it is set to zero to allow an external reference voltage to be used. The formula for Vin is:

$$Vin = V_{REF} \times \frac{N}{2^{14}} \times \frac{R1 + R2}{R2}$$

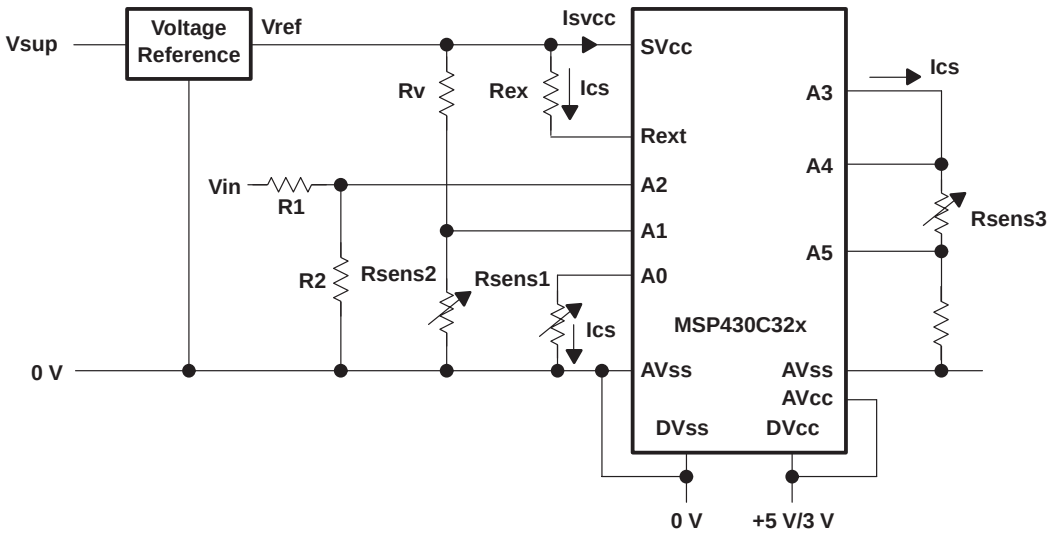

Figure 33. SVcc Terminal Used as an Input for a Reference Voltage

NOTE: If an external voltage reference is used, then it must be able to deliver not only the current for the external circuitry but also a maximum current of 80 μA at 5 V to supply the parts of the ADC that are connected to SVcc.

The maximum voltage at SVcc when used as an input is the voltage applied to AVcc.

Measurements with the ADC using external reference voltages down to 1.2 V at SVcc showed that the ADC does not change its characteristic. However, the noise of the result doubles when compared to a 5-V supply. This is due to the voltage-independent noise generated by the ADC.

3.3.3 Connection of Current Consuming Loads to SVcc

If the current drawn by the external ADC circuitry exceeds 8 mA, then an external switch for the external analog voltage should be considered. A simple PNP transistor can be used for this purpose as shown in Figure 34. The SVcc terminal is used as an input pin for the external reference voltage (ADC control bit VREF = 0). This method allows the full accuracy of the ADC also with current consuming loads. Output TP.0 switches the power to the current consuming loads off and on.

The schematic in Figure 34 is simplified for clarity. The connection principle shown in *Application Basics for the MSP430 14-Bit ADC Application Report* (SLAA046)[3] needs to be applied, especially with the larger currents flowing here.

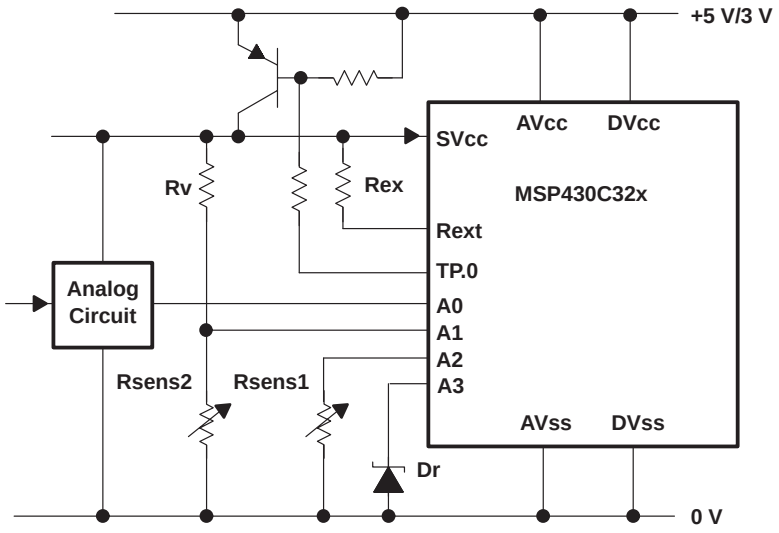

Figure 34. Connection of Current Consuming Loads to SVcc

The software for switching the PNP transistor follows. The TP-port handling may be included in the MEASR subroutine if this is an advantage. The example refers to the hardware shown in Figure 34. Rsens1 is measured.

```
BIC.B     #TP0,&TPD              ; TP.0 pin is low if enabled
BIS.B     #TP0,&TPE              ; Enable TP0: switch PNP on
MOV       #ADCLK+RNGA+CSA2+A2,&ACTL ; ADC: ext. reference
CALL      #MEASR                 ; Measure Rsens1 at A2
BIC.B     #TP0,&TPE              ; Switch PNP off: TP0 Hi-Z
...                              ; Result in ADAT
```

3.4 Interrupt Handling

All of the ADC software examples shown previously use polling techniques to check for conversion completion. This takes up computing power that can be used more effectively if interrupt techniques are used.

3.4.1 Interrupt Flags

ADC interrupt flags are not located in the ACTL control register. This allows advanced interrupt handling. Several interrupt enable flags in a common byte can be disabled and enabled together with minimal effort, something that is impossible with flags located in the individual control words. The two flags controlling the interrupt of the ADC are:

```
IE2    .EQU   01h    ; Interrupt Enable Register 2
ADIE   .EQU   04h    ; ADC interrupt enable bit (IE2.2)
;
IFG2   .EQU   03h    ; INTERRUPT FLAG REGISTER 2
ADIFG  .EQU   04h    ; ADC ”EOC” Bit (IFG2.2)
```

3.4.2 Interrupt Handlers

The interrupt structure of the ADC allows the conversion time to be used for other calculations or procesor tasks. Two ADC interrupt handler examples follow:

EXAMPLE: analog input A0 (without current source) and A1 (with the current source enabled) are measured alternately. The measured 14-bit results are stored in address MEAS0 for input A0 and MEAS1 for input A1. The time interval between the two measurements is defined by the 8-bit timer: each timer interrupt starts a new conversion for the previously prepared analog input. Other timers may also be used for the generation of the time interval.

```
; Analog input      A0        A1
; Current Source    OFF       ON
; Result to         MEAS0     MEAS1
; Range selection   AUTO      AUTO
; Reference         SVcc      SVcc
;
; Initialization part for the ADC:
;
       MOV    #RNGAUTO+CSOFF+A0+VREF,&ACTL
       BIS.B  #ADIE,&IE2    ; Enable ADC interrupt
       MOV.B  #0FFh–3,&AEN  ; Only A0 and A1 analog inputs
       ...                  ; Initialize other modules
;
; ADC interrupt handler: A0 and A1 are measured alternately.
; The next measurement is prepared but not started.
; The interrupt flag ADIFG is reset automatically
;
ADC_INT   BIT    #A1,&ACTL     ; A1 result in ADAT?
          JNZ    ADI           ; Yes
          MOV    &ADAT,MEAS0   ; A0 value is actual
          MOV    #RNGAUTO+CSON+A1+VREF,&ACTL   ; A1 next meas.
          RETI
ADI       MOV    &ADAT,MEAS1                   ; A1 value is actual
          MOV    #RNGAUTO+CSOFF+A0+VREF,&ACTL  ; A0 next meas.
          RETI
;
; 8-bit timer interrupt handler: the ADC conversion is started
; for the previously prepared ADC input
;
T8BINT    BIS    #SOC,&ACTL             ; Start conversion for the ADC
          ...                       ; Execute other timer tasks
          RETI
;
          .SECT  ”INT_VEC0”,0FFEAh   ; Interrupt vectors
          .WORD  ADC_INT             ; ADC interrupt vector
          .SECT  ”INT_VEC1”,0FFF8h
          .WORD  T8BINT              ; 8-bit timer interrupt vector
```

The software for the 12-bit conversion is similar to that for the 14-bit conversion, the only difference being the replacement of the RNGAUTO bit during the initialization of the ACTL control register. Instead, the desired range (RNGA, RNGB, RNGC, or RNGD) is included in the initialization part of each measurement.

NOTE: An independent timer—like that used in the example above—is recommended; do not use the ADC interrupt handler to restart the ADC. If the ADC interrupt handler starts the next conversion, then any interrupt failure leads to a flip-flop effect; the missing ADC interrupt does not start a new conversion, and the ADC activity ceases.

EXAMPLE: for best results the CPU is switched off during the ADC measurement. The measurement subroutine starts the conversion and switches off the CPU afterwards. The interrupt routine called by the conversion completion resets the CPUoff bit (SR.4) of the stored status register SR and allows the CPU to continue with the measured ADC result. The 12-bit result is moved to R5.

```
CPUoff    .equ   010h          ; SR: CPU off bit
GIE       .equ   008h          ; SR: General Intrpt enable
RNGB      .equ   0200h         ; ACTL: Select Range B
;
             ...
             BIC.B  #ADIFG,&IFG2  ; Reset ADC flag
             BIS.B  #ADIE,&IE2    ; ADC Intrpt Enable
             EINT                 ; Enable GIE interrupt
             MOV    #RNGB+CSOFF+A1+VREF,&ACTL  ; Define ADC
             CALL   #MEASURE      ; Measure with ADC
             MOV    &ADAT,R5      ; Result to R5
             ...                  ; Process result in R5
;
; Subroutine: CPU is switched off to get minimum noise
;
MEASURE      BIS    #SOC,&ACTL       ; Start ADC conversion
             BIS    #CPUoff,SR    ; Switch CPU off, MCLK active
             NOP                  ; Wait for completion of ADC
             RET                  ;
;
; Interrupt Handler for the Analog-to-Digital Converter
; The CPUoff bit of the saved SR is cleared to allow the
; software to continue after the RETI
;
ADC_INT      BIC    #CPUoff,0(SP); Allow SW run (CPUoff = 0)
             RETI
;
; Interrupt Vectors
;
             .sect  ”INT_VEC1”,0FFEAh
             .WORD  ADC_INT       ; ADC Vector
```

3.5 ADC Clock Generation

The frequency of the ADC clock, ADCLK, must be in a certain range as discussed in section 2.1.1 *ADC Timing Restrictions*. To allow the adaptation of the ADCLK to the full range of the MCLK frequency, four possibilities of prescaling are provided:

- MCLK if MCLK < 1.5 MHz
- MCLK/2 if MCLK > 1.5 MHz
- MCLK/3 if MCLK > 3.0 MHz

This allows an MCLK/ADCLK combination to be selected for nearly all applications that fits the calculation needs, while providing the necessary A/D conversion speed.

4 ADC Characteristics

The next four figures show typical measured ADC characteristics: the absolute error (ADC steps) is dependent on the input value (ADC steps from 5 to 16380). Error characteristics like these are used with *Additive Improvement of the MSP430 14-Bit ADC Characteristic Application Report* (SLAA047)[4], *Linear Improvement of the MSP430 14-Bit ADC Characteristic Application Report* (SLAA048)[5], and *Nonlinear Improvement of the MSP430 14-Bit ADC Characteristic Application Report* (SLAA050)[6] to illustrate the improvements possible by methods using different hardware and software.

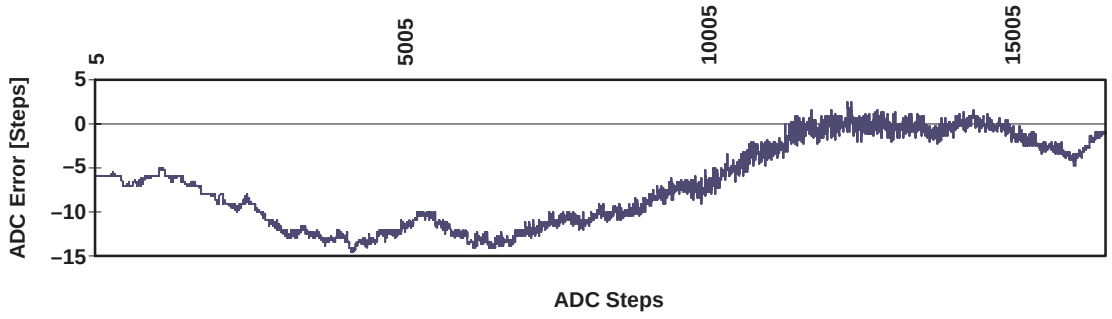

Figure 35. Error Characteristic Device 1

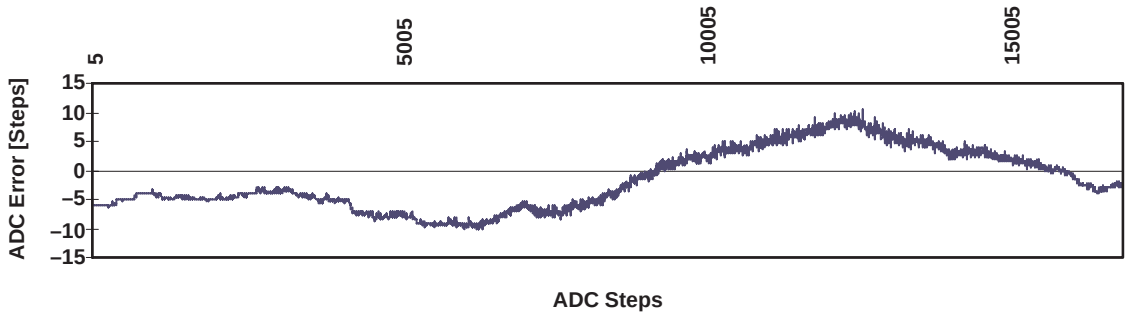

Figure 36. Error Characteristic Device 2

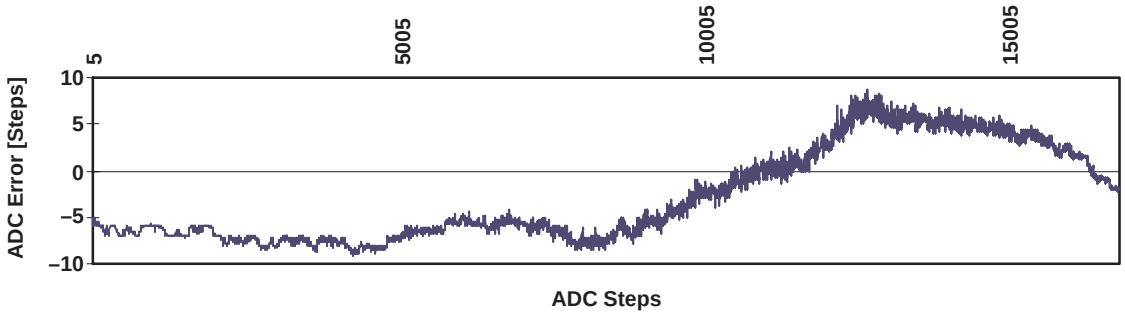

Figure 37. Error Characteristic Device 3

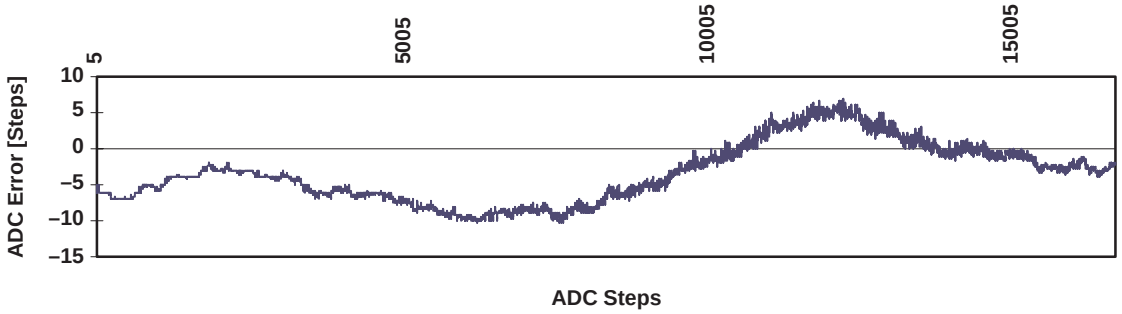

Figure 38. Error Characteristic Device 4

5 Summary

This application report complements *Application Basics for the MSP430 14-Bit ADC Application Report* (SLAA046)[3] that contains applications of the 14-bit ADC. *Additive Improvement of the MSP430 14-Bit ADC Characteristic Application Report* (SLAA047)[4] explains different methods to minimize the ADC error, and the limitations of the ADC.

All five of the application reports in the MSP430 14-bit ADC series include system applications (hardware and proven software) using all parts and modes of the ADC.

6 References

1. *MSP430 Family Architecture Guide and Module Library*, 1996, Literature #SLAUE10B
2. Data Sheet, MSP430x32x Mixed Signal Microcontroller, 1998, Literature #SLAS164
3. *Application Basics for the MSP430 14-Bit ADC Application Report*, 1999, Literature #SLAA046
4. *Additive Improvement of the MSP430 14-Bit ADC Characteristic Application Report*, 1999, Literature #SLAA047
5. *Linear Improvement of the MSP430 14-Bit ADC Characteristic Application Report*, 1999, Literature #SLAA048
6. *Nonlinear Improvement of the MSP430 14-Bit ADC Characteristic Application Report*, 1999, Literature #SLAA050
7. *MSP430 Metering Application Report*, 1998, Literature #SLAAE10C

Appendix A Definitions Used With the Application Examples

```
; HARDWARE DEFINITIONS
;
AIN       .equ   0110h  ; Input register (for digital inputs)
AEN       .equ   0112h  ; 0: analog input   1: digital input
;
ACTL      .equ   0114h  ; ADC control register: control bits
SOC       .equ   01h    ; Conversion start
VREF      .equ   02h    ; 0: ext. reference    1: SVcc on
A0        .equ   00h    ; Input A0
A1        .equ   04h    ; Input A1
A2        .equ   08h    ; Input A2
A3        .equ   0Ch    ; Input A3
A4        .equ   10h    ; Input A4
A5        .equ   14h    ; Input A5
CSA0      .equ   00h    ; Current Source to A0
CSA1      .equ   40h    ; Current Source to A1
CSA2      .equ   80h    ; Current Source to A2
CSA3      .equ   0C0h   ; Current Source to A3
CSOFF     .equ   100h   ; Current Source off
CSON      .equ   000h   ; Current Source on
RNGA      .equ   000h   ; Range select A (0 ... 0.25×SVcc)
RNGB      .equ   200h   ; Range select B (0.25..0.50×SVcc)
RNGC      .equ   400h   ; Range select C (0.5...0.75×SVcc)
RNGD      .equ   600h   ; Range select D (0.75..SVcc)
RNGAUTO   .equ   800h   ; 1: range selected automatically
PD        .equ   1000h  ; 1: ADC powered down
ADCLK1    .equ   0000h  ; ADCLK = MCLK
ADCLK2    .equ   2000h  ; ADCLK = MCLK/2
ADCLK3    .equ   4000h  ; ADCLK = MCLK/3
ADCLK4    .equ   6000h  ; ADCLK = MCLK/4
;
ADAT      .equ   0118h  ; ADC data register (12 or 14-bits)
;
IFG2      .equ   03h    ; Interrupt flag register 2
ADIFG     .equ   04h    ; ADC ”EOC” bit (IFG2.2)
;
IE2       .equ   01h    ; Interrupt enable register 2
ADIE      .equ   04h    ; ADC interrupt enable bit (IE2.2)
;
TPD       .equ   04Eh   ; TP–port: address data register
TPE       .equ   04Fh   ; TP–port: address of enable register
TP0       .equ   1      ; Bit address of TP.0
TP1       .equ   2      ; Bit address of TP.1
```