

# Using direct data transfer to maximize data acquisition throughput

By Mark Buccini (Email: m-buccini@ti.com)

MSP430 Applications Manager

## Introduction

Increasing real-time throughput in MCU-based data acquisition systems is a challenge faced in many new applications. By definition, real-time applications must sample, digitize, transfer, and process acquired data before subsequent samples are made. Adequate CPU reserves must also be kept between samples to allow digital signal processing of the data. When acquisition rates exceeds 10 kSPS, engineers are often forced to make compromises due to the extreme overhead placed on the MCU. The basic task of moving acquired data becomes dominant and often unmanageable. Increasing CPU clock speeds or using specialized digital signal processors (DSPs) is often the only solution to meeting high data acquisition rates. This article offers alternatives that maximize system performance using single-chip mixed-signal MCU solutions with direct data transfer control.

## System example using an external ADC

A typical multi-chip MCU-based data acquisition system is presented in Figure 1. The system includes an MSP430F123 MCU and a TLV1549 10-bit ADC. A low-power, 32-kHz watch crystal is used for the auxiliary clock (ACLK); and the MCU's internal digitally controlled oscillator (DCO) at its default frequency of ~1 MHz is used for the CPU's master clock (MCLK). The system samples a sensor signal with 10 bits of accuracy and transfers the acquired output code to on-chip RAM at 8192 samples per second. After 20 ADC samples have been acquired, the MCU will process the data stored in RAM. An application with such requirements could be a DTMF detection system.

The interface between the ADC and the MCU is glueless, using the MCU's integrated USART in SPI mode. To support the 8192-sample rate, the MCU's 16-bit Timer\_A uses the ACLK with capture compare register 0 (CCR0) configured to trigger an interrupt service routine (CCR0\_ISR) at the required sampling rate.

Inside the `MainLoop`, the MCU is normally in low-power mode 3 (LPM3). `CCR0_ISR` wakes the CPU, and an ADC sample is made. For each `CCR0_ISR`, software enables the ADC by resetting the ADC conversion start `CS`. Data are exchanged between the ADC and MCU as two 8-bit bytes for each 10-bit conversion. The software starts and times a conversion by writing two dummy bytes back-to-back to the MCU's USART transmit buffer (TXBUF). The software then polls the USART receive interrupt flag (RXIFG) to indicate the receipt of a data byte from the ADC to the USART's receive buffer (RXBUF). Two bytes are received, packed, and transferred to MCU RAM with software and a pointer register R4. `CS` is reset, disabling the ADC. After 20 samples have been acquired, the `MainLoop` breaks from LPM3; and the data are processed as shown in Code Listing 1.

Figure 1. External ADC MCU data acquisition system

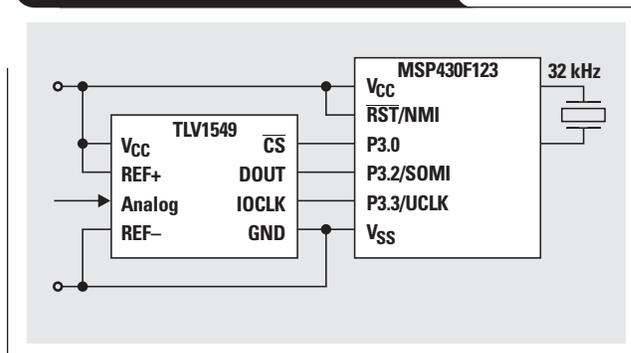
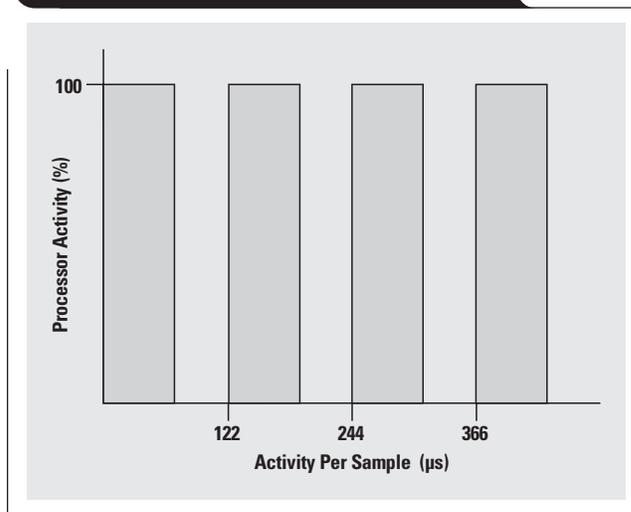


Figure 2. External ADC with MCU data acquisition system activity



For each external ADC measurement, 16 I/O clocks are required. The MCU's USART transfers data at half of the applied clock—the DCO, the same clock source used for the CPU MCLK. Thus 16 I/O clocks are equivalent to 32 MCLKs. The `CCR0_ISR` overhead and transfer of the ADC conversion code to memory takes additional cycles, for a total of 83 CPU MCLKs for each sample. With 8192 samples per second, an external ADC, and a 1-MHz MCLK, the CPU overhead is calculated as follows:

$$\text{CPU}_{\text{overhead (external ADC)}} = \frac{83 \times 8192}{1000000} = 0.68$$

The CPU is loaded 68%, as shown in Figure 2. The Timer\_A `CCR0_ISR` and two CPU registers are also required.

**Code Listing 1: MSP430F123/TLV1549 software example**

```

#include "msp430x12x.h"
;*****
; MSP-FET430P120 Demo - USART0 SPI Interface to TLV1549 10-bit ADC
;
; M.Buccini - Texas Instruments, Inc - July 2002
;*****
;-----
;          ORG      0F000h          ; Program Reset
;-----
RESET      mov.w    #0300h,SP      ; Initialize stackpointer
StopWDT    mov.w    #WDTPW+WDTHOLD,&WDTCTL ; Stop watchdog timer
SetupP3    bis.b    #0Ch,&P3SEL     ; P3.2,3 SPI option select
           bis.b    #09h,&P3DIR     ; P3.3,0 output direction
SetupSPI   bis.b    #USPIE0,&ME2    ; Enable USART0 SPI
           bis.b    #CKPH+SSEL1+SSEL0+STC,&UTCTL0 ; SMCLK, 3-pin
           bis.b    #CHAR+SYNC+MM,&UCTL0 ; 8-bit SPI Master
           mov.b    #02h,&UBR00     ; SMCLK/2 for baud rate
           clr.b    &UBR10         ; SMCLK/2 for baud rate
           clr.b    &UMCTL0        ; Clear modulation
           bic.b    #SWRST,&UCTL0   ; **SWRST**
SetupTA    mov.w    #TASSEL0+TACLR,&TACTL ; ACLK, clear TAR
SetupC0    mov.w    #CCIE,&CCTL0    ; CCR0 interrupt enabled
           mov.w    #4-1,&CCR0      ; CCR0 counts to 4
           bis.w    #MC0,&TACTL    ; Start Timer_a in upmode
           eint                    ; Enable interrupts
           ;
Mainloop   clr.w    R4              ; Clear pointer
Meas1549   bis.w    #LPM3,SR        ;
           bic.b    #01h,&P3OUT     ; Enable TLV1549, /CS reset
           mov.b    #00h,&TXBUF0    ; Dummy write to start SPI
           mov.b    #00h,&TXBUF0    ;
L1         bit.b    #URXIFG0,&IFG2  ; RXBUF ready?
           jnc     L1              ; 1 = ready
           mov.b    &RXBUF0,R5     ; R5 = 00|MSB
           swpb    R5              ; R5 = MSB|00
L2         bit.b    #URXIFG0,&IFG2  ; RXBUF ready?
           jnc     L2              ; 1 = ready
           mov.b    &RXBUF0,R6     ; R6 = 00|LSB
           add.w   R6,R5           ; R6 = MSB|LSB
           bis.b    #01h,&P3OUT     ; Disable TLV1549, /CS set
           mov.w    R5,0200h(R4)   ;
           incd.w  R4              ;
           cmp.w   #040,R4         ; R4 = 20 words?
           jne     Meas1549        ;
;          ** SIGNAL PROCESSING HERE
           jmp     Mainloop        ; Again
           ;
TAO_ISR    mov.w    #GIE ,0(SP)    ; System active on reti
           reti                    ;
           ;
           ORG     0FFFEh          ; MSP430 RESET Vector
           DW     RESET            ;
           ORG     0FFF2h          ; Timer_A0 Vector
           DW     TAO_ISR         ;
           END

```

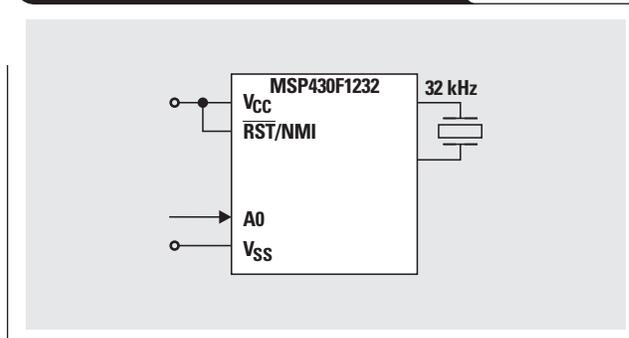
## Using an internal ADC

To miniaturize applications and reduce system cost, an ADC is commonly integrated into an MCU. This integration provides a more compact system on-chip solution with the twofold benefit of reduced board space and the elimination of a serial port required for communicating between the MCU and ADC. An example of an MCU with an integrated 10-bit ADC is the MSP430F1232, as shown in Figure 3.

The MSP430F1232 eliminates the external 10-bit ADC, using the integrated ADC10 instead. CCR0\_ISR is again used to wake the CPU from LPM3 in the Mainloop, as shown in Code Listing 2. The Mainloop manages the acquisition of 20 samples, storing the output code from the ADC10 memory buffer (ADC10MEM) to RAM. Only 40 MCLKs are required for each sample. With the internal ADC10 and a 1-MHz MCLK, the CPU overhead to support the 8192-sample rate is calculated as follows:

$$\text{CPU}_{\text{overhead (internal ADC)}} = \frac{40 \times 8192}{1000000} = 0.33$$

**Figure 3. MSP430F1232 MCU data acquisition system**



### Code Listing 2: MSP430F1232 ADC10 software example

```
#include "msp430x12x2.h"
;*****
; MSP-FET430P120 Demo - ADC10 Sample A0 20x, AVcc, TA0 ISR
;
; M.Buccini - Texas Instruments, Inc - July 2002
;*****
;-----
;          ORG      0F000h          ; Program Reset
;-----
RESET      mov.w    #0300h,SP      ; Initialize stackpointer
StopWDT    mov.w    #WDTPW+WDTHOLD,&WDTCTL ; Stop watchdog timer
SetupADC10 mov.w    #ADC10SHT_2+ADC10ON,&ADC10CTL0 ; 16x
           bis.b    #01h,&ADC10AE   ; P2.0 ADC10 option select
SetupTA    mov.w    #TASSEL0+TACLRL,&TACTL ; ACLK, clear TAR
SetupC0    mov.w    #CCIE,&CCTL0    ; CCR0 interrupt enabled
           mov.w    #4-1,&CCR0      ; CCR0 counts to 4
           bis.w    #MC0,&TACTL     ; Start Timer_a in upmode
           eint                    ; Enable interrupts
           ;
Mainloop   clr.w    R4              ; Clear pointer
MeasADC10  bis.w    #ENC+ADC10SC,&ADC10CTL0 ; Start sampling/conversion
           bis.w    #LPM3,SR        ;
           mov.w    &ADC10MEM,0200h(R4) ;
           incd.w   R4              ;
           cmp.w    #040,R4         ; R4 = 20 words?
           jne     MeasADC10        ;
;          ** SIGNAL PROCESSING HERE
           jmp     Mainloop          ; Again
           ;
TA0_ISR    mov.w    #GIE ,0(SP)     ; System active on reti
           reti                    ;
           ;
           ORG      0FFFEh          ; MSP430 RESET Vector
           DW       RESET           ;
           ORG      0FFF2h          ; Timer_A0 Vector
           DW       TA0_ISR         ;
           END
```

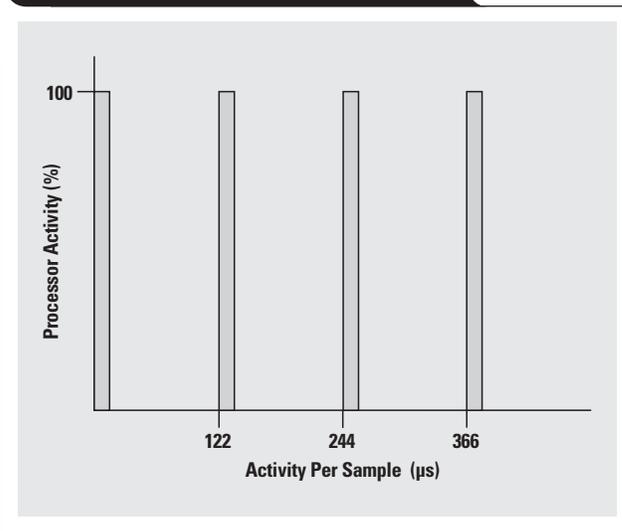
The CPU is loaded 33%, as shown in Figure 4. Using an internal ADC reduces the CPU overhead by 50% compared to when an external ADC is used. The CPU overhead reduction is accomplished by eliminating the software and delay associated with servicing the serial port communication between the MCU and the external ADC. The free serial port can be used for other features or completely eliminated in cost-sensitive applications.

**Data transfer controller**

In addition to ADC10, the MSP430F1232 contains a data transfer controller (DTC). The DTC provides the capability of automatically transferring conversion code from the ADC10 output buffer memory (ADC10MEM) directly to on-chip memory—without CPU intervention, as shown in Figure 5. The start address (ADC10SA) of the transfer sequence can point anywhere in the MCU's memory. The total number of conversions is configured with the ADC10DTC1 control register. The DTC can transfer ADC10 output code in a single-block sequence or a dual-block sequence, both with or without repeating, and both with or without interrupt capability. In the example, the DTC transfers code into a single block, repeating with interrupt.

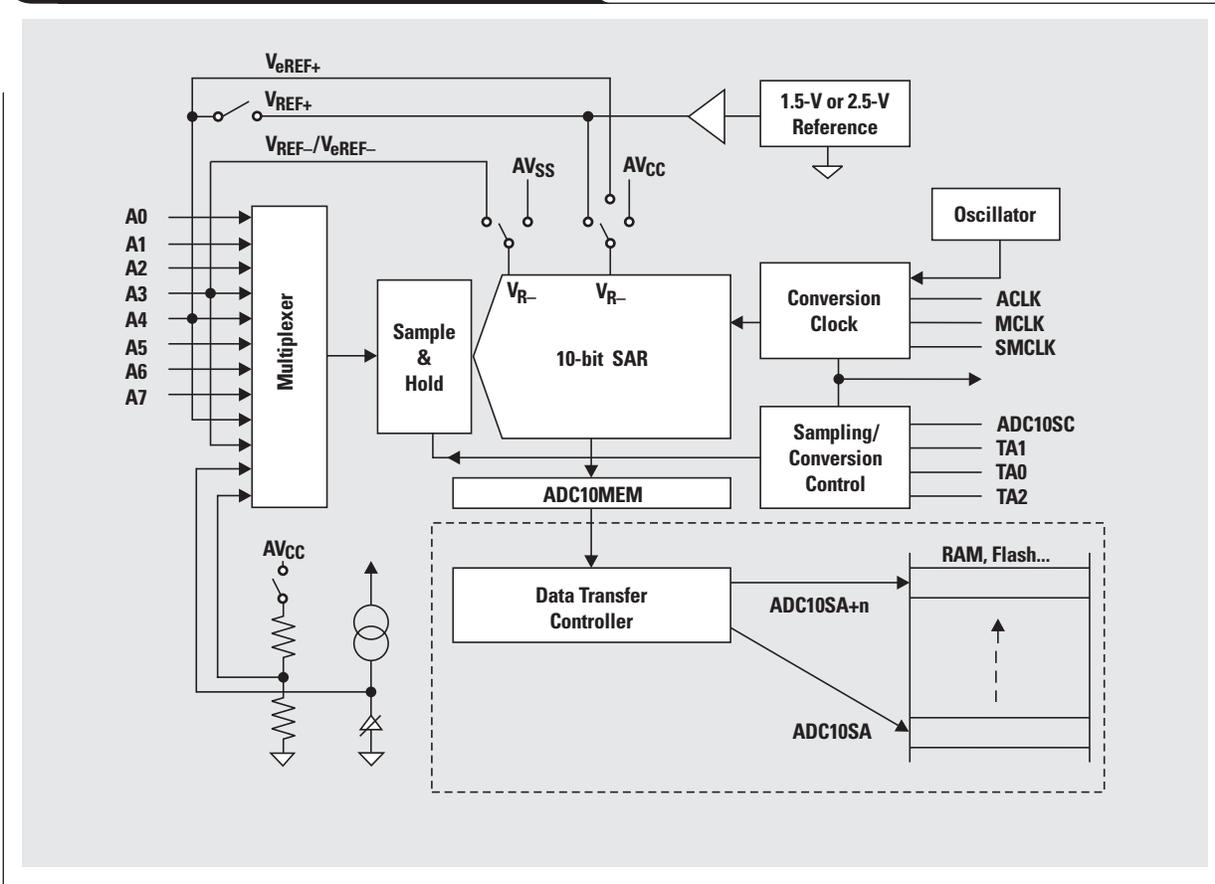
The DTC does not require the CPU to transfer ADC10MEM to memory, but it does require one memory data bus (MDB) clock, which is the same as the MCLK used by the CPU. When an MDB clock is required for

**Figure 4. MSP430F1232 ADC10 data acquisition system activity**



transfer, the DTC will halt any CPU activity for exactly one clock. This halt is to prevent memory conflict between the DTC and CPU. In effect, the DTC “steals” one MCLK from

**Figure 5. MSP430F1232 ADC10 block diagram**



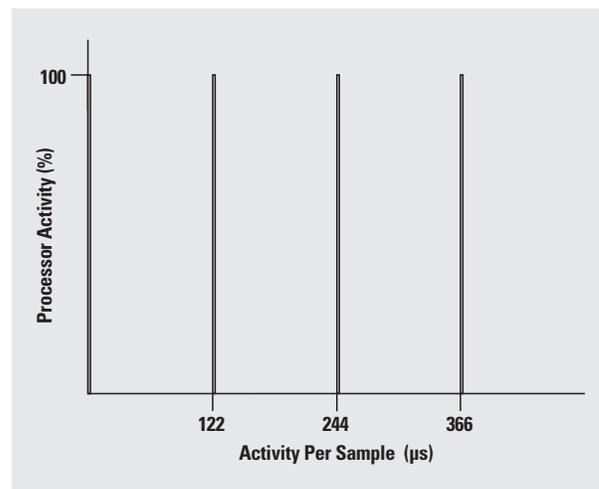
the CPU for each DTC transfer. This may be interpreted as CPU overhead. If the CPU is not active and using the MDB, the effect of the DTC is of no consequence.

With 8192 samples per second, a 1-MHz CPU MCLK, and use of the DTC, the CPU overhead is calculated as follows:

$$\text{CPU}_{\text{overhead (internal ADC with DTC)}} = \frac{1 \times 8192}{1000000} = 0.008$$

The CPU is loaded less than 1% while sustaining an 8192-sample rate, as shown in Figure 6. The DTC allows over 99% of the CPU resources to be available for digital signal processing and control. In the example, the DTC will set the ADC10 interrupt after a block of 20 samples has been transferred. The function of R4, used as a loop counter in the previous two examples, is no longer required. The DTC and ADC10 interrupt service routine (ADC10\_ISR) will wake the CPU from LPM3 in Mainloop only after a complete block of 20 samples has been taken and transferred automatically to RAM, as shown in Code Listing 3.

**Figure 6. MSP430F1232 ADC10 with DTC data acquisition system activity**



### Code Listing 3: MSP430F1232 ADC10 with DTC software example

```
#include "msp430x12x2.h"
;*****
; MSP-FET430P120 Demo - ADC10 Sample A0 20x, AVcc, TA0 Trigger, DTC DCO
;
; M.Buccini - Texas Instruments, Inc - July 2002
;*****
;-----
;          ORG      0E000h          ; Program Start
;-----
RESET      mov.w    #0300h,SP          ; Initialize stackpointer
StopWDT    mov.w    #WDTPW+WDTHOLD,&WDTCTL ; Stop WDT
SetupADC10 mov.w    #SHS_2+CONSEQ_2,&ADC10CTL1 ; TA0 trigger
           mov.w    #ADC10SHT_2+ADC10ON+ADC10IE,&ADC10CTL0;
           mov.b    #ADC10CT,&ADC10DTC0    ; Continuos
           mov.b    #020,&ADC10DTC1        ; 20 conversions
           bis.b    #001h,&ADC10AE         ; P2.0 ADC10 option select
SetupTA    mov.w    #TASSEL0+TACLRL,&TACTL ; ACLK, clear TAR
SetupC0    mov.w    #OUTMOD_4,&CCTL0       ; CCR0 toggle
           mov.w    #2-1,&CCRO             ; PWM Period
           mov.w    #0200h,&ADC10SA        ; Data buffer start
           bis.w    #ENC,&ADC10CTL0        ; Sampling and conversion ready
           bis.w    #MC0,&TACTL            ; Start Timer_a in upmode
           eint                             ; Enable interrupts
           ;
Mainloop   bis.w    #LPM3,SR              ; LPM3, ADC10 ISR will force exit
;          ** SIGNAL PROCESSING HERE
           jmp     Mainloop                ; Again
           ;
ADC10_ISR  mov.w    #GIE,0(SP)            ; System active on reti
           reti
           ;
           ORG     0FFFEh                  ; MSP430 RESET Vector
           DW     RESET
           ;
           ORG     0FFEAh                  ; ADC10 Vector
           DW     ADC10_ISR
           ;
END
```

## DTC channel scanning

Some systems, such as an electricity meter, require channel scanning, mixing alternate samples of voltage and current. Using software and an internal or external ADC to switch channels adds at least 10 MCLK cycles of overhead for each sample. At an 8192-sample rate with a 1-MHz MCLK, software-based channel scanning adds an additional 0.08% CPU overhead. DTC provides the ability to scan a mix of different channels (i.e., A2, A1, A0, A2, A1, A0...) automatically without any CPU resources. If channel scanning is useful in an application, the benefits of using a DTC are magnified.

## SOC-introduced phase error

High-performance data acquisition systems must take into account several factors in addition to the sampling speed discussed so far. In many applications, an analog sensor output must be sampled at very exact intervals to detect small phase changes. For example, stimuli may be presented to sensor input, with a delayed response measured at an output. For demonstration purposes, a repeating 10-kHz triangle signal from 0 to 3 V is shown in Figure 7. The rate of change of the triangle signal is calculated as follows:

$$\text{Signal}(dv/dt) = 3 \text{ V}/100 \mu\text{s} = 30 \text{ mV}/\mu\text{s}$$

With a 10-bit ADC in a 3-V system, the ADC step size is the ADC reference (assumed to be the supply) divided by 1024.

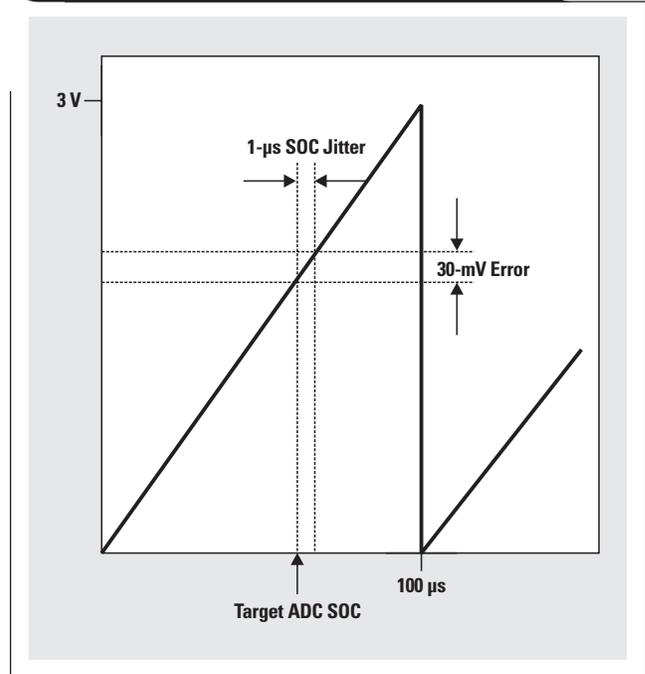
$$\text{ADC10}(\text{step}) = 3 \text{ V}/1024 = 3 \text{ mV}$$

If only software is used to trigger an ADC start of conversion (SOC), some uncertainty or jitter will always be present. Other always-present CPU activity such as servicing an interrupt or UART handler causes this uncertainty. Unless all other CPU activity is ceased prior to a software-initiated SOC—very impractical for modern embedded applications—some uncertainty with software-triggered SOC will always be possible.

For example, using the example system with an MCLK of 1 MHz (1- $\mu\text{s}$  clock), the effect of just one clock of SOC uncertainty will introduce 30 mV of ADC uncertainty or jitter error. This 30 mV of uncertainty error is equivalent to 10 10-bit steps. The 10-bit ADC output code with 1  $\mu\text{s}$  of SOC jitter is reduced to an accuracy of a 7-bit ADC!

When an ADC10 is used, a CCRx output can automatically trigger ADC10 SOC with the precision of the used

Figure 7. ADC sampling uncertainty error



timer clock—all interrupt and latency uncertainty associated with software-driven SOC is removed completely from the system. The ADC10 SOC will be triggered with perfect timing, using the CCRx hardware asynchronously and regardless of other CPU, software, or system activity.

## DTC system throughput performance increase

Using DTC in data acquisition applications relieves the CPU from the burden of transferring ADC output code to memory. The DTC in itself reduces CPU loading by a factor of 40 $\times$  compared with using an internal ADC10, and by a factor of greater than 80 $\times$  compared with using an external ADC.

In the example using an external ADC, the MCU MCLK would need to be increased to 4 MHz to acquire the ADC's maximum conversion rate of 38 kSPS—and the CPU would be loaded 79%. At the same 4-MHz MCLK and with the MSP430F1232's internal ADC10, 100 kSPS can be

acquired at 100% CPU loading; and, with the DTC, 235 kSPS are possible at only 6% CPU loading. See Figure 8.

MCUs with DTC-enabled ADCs allow well over an order-of-magnitude reduction in CPU loading in data acquisition systems. CPU performance reserves can be focused on differentiated digital signal processing instead of on the basic process of sampling and moving data.

**References**

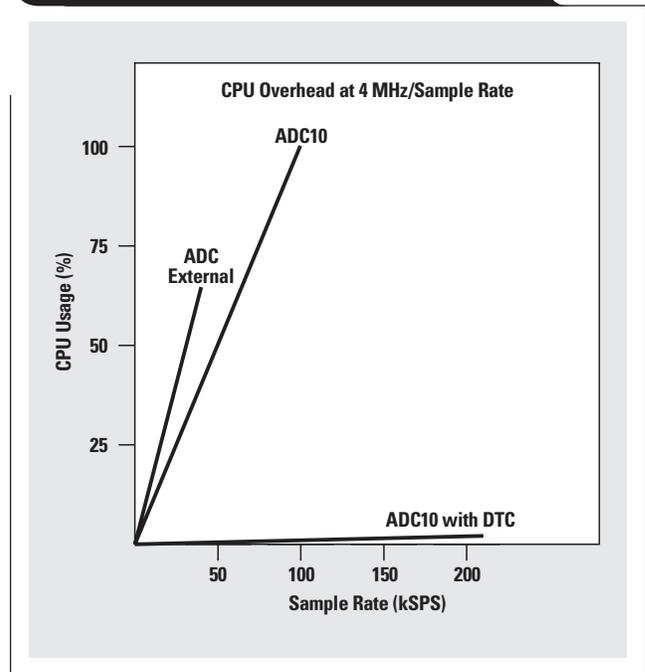
For more information related to this article, you can download an Acrobat Reader file at [www-s.ti.com/sc/techlit/litnumber](http://www-s.ti.com/sc/techlit/litnumber) and replace "litnumber" with the **TI Lit. #** for the materials listed below.

| <b>Document Title</b>   | <b>TI Lit. #</b> |
|---|------------------|
| 1. "MSP430x1xx Family," User's Guide . . . . .  | slau049          |
| 2. "MSP430x11x2, MSP430x12x2 Mixed Signal Microcontroller," Data Sheet . . . . .                                | slas361          |
| 3. "MSP430x12x Mixed Signal Microcontroller," Data Sheet . . . . .  | slas312          |
| 4. "TLV1549C, TLV1549I, TLV1549M 10-Bit Analog-to-Digital Converters with Serial Control," Data Sheet . . . . . | slas071          |

**Related Web sites**

[analog.ti.com](http://analog.ti.com)  
[www.ti.com/sc/device/partnumber](http://www.ti.com/sc/device/partnumber)  
 Replace *partnumber* with MSP430F123, MSP430F1232 or TLV1549

**Figure 8. Data acquisition CPU overhead**



## IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

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

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

### Products

|                  |  |
|------------------|--|
| Amplifiers       | <a href="http://amplifier.ti.com">amplifier.ti.com</a>             |
| Data Converters  | <a href="http://dataconverter.ti.com">dataconverter.ti.com</a>     |
| DSP              | <a href="http://dsp.ti.com">dsp.ti.com</a>                         |
| Interface        | <a href="http://interface.ti.com">interface.ti.com</a>             |
| Logic            | <a href="http://logic.ti.com">logic.ti.com</a>                     |
| Power Mgmt       | <a href="http://power.ti.com">power.ti.com</a>                     |
| Microcontrollers | <a href="http://microcontroller.ti.com">microcontroller.ti.com</a> |

### Applications

|                    |  |
|--------------------|--|
| Audio              | <a href="http://www.ti.com/audio">www.ti.com/audio</a>                   |
| Automotive         | <a href="http://www.ti.com/automotive">www.ti.com/automotive</a>         |
| Broadband          | <a href="http://www.ti.com/broadband">www.ti.com/broadband</a>           |
| Digital control    | <a href="http://www.ti.com/digitalcontrol">www.ti.com/digitalcontrol</a> |
| Military           | <a href="http://www.ti.com/military">www.ti.com/military</a>             |
| Optical Networking | <a href="http://www.ti.com/opticalnetwork">www.ti.com/opticalnetwork</a> |
| Security           | <a href="http://www.ti.com/security">www.ti.com/security</a>             |
| Telephony          | <a href="http://www.ti.com/telephony">www.ti.com/telephony</a>           |
| Video & Imaging    | <a href="http://www.ti.com/video">www.ti.com/video</a>                   |
| Wireless           | <a href="http://www.ti.com/wireless">www.ti.com/wireless</a>             |

## TI Worldwide Technical Support

### Internet

**TI Semiconductor Product Information Center Home Page**  
[support.ti.com](http://support.ti.com)

**TI Semiconductor KnowledgeBase Home Page**  
[support.ti.com/sc/knowledgebase](http://support.ti.com/sc/knowledgebase)

### Product Information Centers

#### Americas

|                |  |     |                  |
|----------------|--|-----|------------------|
| Phone          | +1(972) 644-5580   | Fax | +1(972) 927-6377 |
| Internet/Email | <a href="http://support.ti.com/sc/pic/americas.htm">support.ti.com/sc/pic/americas.htm</a> |     |                  |

#### Europe, Middle East, and Africa

|                   |  |                       |                       |
|-------------------|--|-----------------------|-----------------------|
| Phone             |  |                       |                       |
| Belgium (English) | +32 (0) 27 45 54 32  | Netherlands (English) | +31 (0) 546 87 95 45  |
| Finland (English) | +358 (0) 9 25173948  | Russia                | +7 (0) 95 7850415     |
| France            | +33 (0) 1 30 70 11 64  | Spain                 | +34 902 35 40 28      |
| Germany           | +49 (0) 8161 80 33 11  | Sweden (English)      | +46 (0) 8587 555 22   |
| Israel (English)  | 1800 949 0107  | United Kingdom        | +44 (0) 1604 66 33 99 |
| Italy             | 800 79 11 37   |                       |                       |
| Fax               | +(49) (0) 8161 80 2045   |                       |                       |
| Internet          | <a href="http://support.ti.com/sc/pic/euro.htm">support.ti.com/sc/pic/euro.htm</a> |                       |                       |

#### Japan

|                |  |          |              |
|----------------|--|----------|--------------|
| Fax            |  |          |              |
| International  | +81-3-3344-5317  | Domestic | 0120-81-0036 |
| Internet/Email |  |          |              |
| International  | <a href="http://support.ti.com/sc/pic/japan.htm">support.ti.com/sc/pic/japan.htm</a> |          |              |
| Domestic       | <a href="http://www.tij.co.jp/pic">www.tij.co.jp/pic</a>                             |          |              |

#### Asia

|               |  |             |  |
|---------------|--|-------------|--|
| Phone         |  |             |  |
| International | +886-2-23786800  |             |  |
| Domestic      | Toll-Free Number   |             |  |
| Australia     | 1-800-999-084  | New Zealand | 0800-446-934   |
| China         | 800-820-8682   | Philippines | 1-800-765-7404                                       |
| Hong Kong     | 800-96-5941  | Singapore   | 800-886-1028   |
| Indonesia     | 001-803-8861-1006  | Taiwan      | 0800-006800  |
| Korea         | 080-551-2804   | Thailand    | 001-800-886-0010                                     |
| Malaysia      | 1-800-80-3973  |             |  |
| Fax           | 886-2-2378-6808  | Email       | <a href="mailto:tiasia@ti.com">tiasia@ti.com</a>     |
| Internet      | <a href="http://support.ti.com/sc/pic/asia.htm">support.ti.com/sc/pic/asia.htm</a> |             | <a href="mailto:ti-china@ti.com">ti-china@ti.com</a> |

### C011905

**Safe Harbor Statement:** This publication may contain forward-looking statements that involve a number of risks and uncertainties. These "forward-looking statements" are intended to qualify for the safe harbor from liability established by the Private Securities Litigation Reform Act of 1995. These forward-looking statements generally can be identified by phrases such as "TI or its management believes," "expects," "anticipates," "foresees," "forecasts," "estimates" or other words or phrases of similar import. Similarly, such statements herein that describe the company's products, business strategy, outlook, objectives, plans, intentions or goals also are forward-looking statements. All such forward-looking statements are subject to certain risks and uncertainties that could cause actual results to differ materially from those in forward-looking statements. Please refer to TI's most recent Form 10-K for more information on the risks and uncertainties that could materially affect future results of operations. We disclaim any intention or obligation to update any forward-looking statements as a result of developments occurring after the date of this publication.

**Trademarks:** All trademarks are the property of their respective owners.

Mailing Address: Texas Instruments  
Post Office Box 655303  
Dallas, Texas 75265

© 2005 Texas Instruments Incorporated