
Interfacing the MSP430x11x(1) and TLV0831

Mark E. Buccini

Mixed Signal Products

ABSTRACT

This application report describes how to interface an MSP430x11x(1) 16-bit RISC-like mixed signal microcontroller and a TLV0831 8-bit A/D converter. This report is written for the MSP430x11x(1) family, but can be adapted to any MSP430 derivative.

Contents

1	Introduction	2
2	Theory of Operation	3
	2.1 External Hardware Required	3
	2.2 TLV0831 Considerations	3
3	Description of Software Listing fet_x17.s43	4
4	Summary	4
5	References	5
Appendix A	Software Listing fet_x17.s43	6

2 Theory of Operation

Using any three available I/O pins, the MSP430x11x(1) controls and communicates with the TLV0831. In this report, MSP430 I/O pins P2.0 and P2.1 are configured as outputs using the P2 direction register (P2DIR) and set using the P2 output register (P2OUT). Pin P2.0 interfaces to the TLV0831 chip select (CS), pin P2.1 to the TLV0831 clock (CLK). Conversion data from the TLV0831 data out (DO) is read on pin P2.3 of the MSP430x11x(1).

The MSP430 initiates a TLV0831 A/D conversion by resetting CS, followed by the application of a single clock pulse on CLK. Eight additional clock pulses are sent to CLK to shift out the 8-bit digital code from the TLV0831. The digital code-conversion data are read from DO after each clock pulse, and shifted into an MSP430 software buffer ADCData. Data are read from the TLV0831 MSB first. The ADCData buffer can be any available CPU register or RAM byte. CS is set upon completion of the A/D conversion, placing DO and CLK into a 3-state condition and the TLV0831 in standby mode.

2.1 External Hardware Required

The MSP430x11x(1)-to-TLV0831 interface is glueless. No external hardware is required. The demonstration circuit in Figure 1 communicates with a PC (optional) and powers up directly from the PC RS232 interface. To accomplish RS232 line level translation, integrated circuits such as TI's MAX232 provide a compliant solution. For the demonstration circuit in Figure 3, a single-gate TI SN74AH1GC04 inverter is used for the PC serial port interface. A TPS77033 low-power low-dropout voltage regulator and 32,768-HZ watch crystal are also used to support the circuit.

2.2 TLV0831 Considerations

No special considerations are required when interfacing an MSP430 and TLV0831. The demonstration circuit in Figure 1 connects the TLV0831 REF pin to VCC and the IN⁻ pin to VSS. In this configuration the analog voltage measured on the IN⁺ pin is ratiometric to VCC. An external voltage reference can be applied to the REF pin allowing an absolute conversion from a voltage other than VCC. If a differential measurement is required, the IN⁺ and IN⁻ pins can be used to measure a voltage difference.

Important: The demonstration circuit in this report powers directly from a PC serial port. Certain PC switching power supplies are very noisy and this noise can be carried on the PC serial port to the MSP430. If excessive jitter is seen in the A/D result, the demonstration circuit may need to be modified using a separate battery for power and adding bypass components.

3 Description of Software Listing fet_x17.s43

Software listing fet_x17.s43 is sectioned into several subroutines. Upon reset, an Init_Sys subroutine is called to initialize the MSP430 system. The I/O and peripherals are configured and the subroutine Set_DCO is called. Set_DCO calibrates the high-speed digitally-controlled oscillator (DCO). In this report the DCO is calibrated to 1,228,800 Hz and configured to drive the master (MCLK) and submaster (SMCLK) MSP430x11x(1) clocks. Initially, the MSP430 is set inside Mainloop to run in low-power mode 3 (LPM3). The watchdog is configured as an interval timer, interrupting and waking the MSP430 at 64-ms intervals. The TLV0831 is then sampled as follows:

```
Meas_ADC;      Shift TVL0831 data into ADCData, R15 used as counter.
               bic.b   #CS,&P2OUT           ; Chip Select low
               mov     #09,R15             ; 9 bits *1 start* + 8 data
ADC_Loop      bis.b   #CLK,&P2OUT         ; Clock high
               bic.b   #CLK,&P2OUT         ; Clock low
               bit.b   #DO,&P2IN          ; DO -> C (carry)
               rlc.b   ADCData            ; C -> ADCData
               dec     R15                 ; All shifted in?
               jnz     ADC_Loop           ; If not -> ADC_Loop
               bis.b   #CS,&P2OUT         ; Chip Select high
```

The above coding only requires 144 MCLK cycles and 25 bytes of assembler code for MSP430 software to drive a complete TLV0831 A/D conversion. The A/D conversion speed is calculated as MCLK/144. For example, with the MCLK set at 1228800 Hz, the conversion time is calculated to be 117 us. The maximum MSP430x11x(1) MCLK is software programmable and much higher than 1,228,800 Hz. Care must be taken not to exceed the maximum allowable TLV0831 CLK rate. Please see current data sheets for maximum device ratings. Upon completion of the A/D conversion, a subroutine TX_2_PC is called. It transmits the conversion data to a PC using a hardware/software UART configuration of timer_A. See the *References* section of this report to locate a description of timer_A used as a UART.

4 Summary

The TLV0831 A/D converter is an excellent complement to the MSP430x11x(1) microcontroller in many cost-sensitive embedded applications. The two-chip solution is versatile and low-power. Using a TLV0831 in an MSP430x11x(1) application allows the designer to add precision A/D conversion capability quite easily. As coded, only 144 MCLK cycles and 25 bytes of MSP430 assembler code are required to drive a TLV0831 conversion. Any three MSP430 I/O pins can be used to communicate with the TLV0831. When the TLV0831 is in standby mode with CS set, DO and CLK are switched to the high-impedance state. In this condition, DO and CLK have no influence on other parts of the circuit. The MSP430 I/O pins used to control DO and CLK can then be used to control other devices such as digital-to-analog converters.

5 References

1. MSP430x11x1 data sheet, 2000, literature number SLAS241B
2. TLV0831 data sheet, 1996, literature number SLAS148
3. *MSP430 Family Architecture and Module Library*, 1996, literature number SLAUE10B
4. *MSP430 Application Report*, 1998, Lutz Bierl, literature number SLAA024
5. *Implementing a UART Function with Timer_A3*, 1999, Mark Buccini, literature number SLAA078

Appendix A Software Listing fet_x17.s43

```
#include          "msp430x11x1.h"
;*****
NAME  FET_x17  ; TLV0831 Interface
;
;  Description: This program will read a TLV0831 ADC and transmit the 8-bit
;  digital code @9600 baud using 8N1 protocol. The TLV0831 is sampled every
;  64ms as timed by the WDT. Program runs normally in LPM3 with the WDT
;  interrupt waking the system to initiate the conversion. Subroutine
;  set_DCO is used to calibrate DCO to a nominal value of 1228800Hz.
;  An external TLV0831 and watch crystal are required.
;
;
;              MSP430F1121
;
;              -----
;
;              |              |-
;  TLV0831     |              |32k
;  -----    |              |-
;  |          CS|<---|P2.0      | 9600 8N1
;  ~>| IN+   CLK|<---|P2.1      P1.1|-----> PC
;  |          DO|--->|P2.3      |
;
;
;  Dedicated CPU registers used
#define      BitCnt    R6
#define      RXTXData  R5
#define      ADCData   R11
;
;  User definitions, 9600 Baud HW/SW UART, MCLK = 37.5x32768= 1228800
Bitime      equ      0128          ; 104 us
Delta       equ      150          ; Delta = (target DCO)/(32768/4)
TXD         equ      002h         ; TXD on P1.1
CS          equ      001h         ; P2.0 Chip Select
CLK         equ      002h         ; P2.1 Clock
DO          equ      008h         ; P2.3 Data Out
LF          equ      0ah          ; ASCII Line Feed
CR          equ      0dh          ; ASCII Carriage Return
;
;  M. Buccini
;  Texas Instruments, Inc
;  March 2000
;*****
;-----
;              ORG      0F000h          ; Program Start
;-----
RESET       mov      #0300h,SP      ; Initialize 'x112x stackpointer
           call     #Init_Sys      ; Initialize system
```

```

;
Mainloop    bis        #LPM3,SR                ; Enter LPM3
;
Meas_ADC;   Shift TVL0831 data into ADCData, R15 used as counter
            bic.b     #CS,&P2OUT                ; Chip Select low
            mov       #09,R15                  ; 9 bits *1 start* + 8 data
ADC_Loop    bis.b     #CLK,&P2OUT              ; Clock high
            bic.b     #CLK,&P2OUT              ; Clock low
            bit.b     #DO,&P2IN                ; DO -> C (carry)
            rlc.b     ADCData                  ; C -> ADCData

            dec       R15                      ; All shifted in?
            jnz       ADC_Loop                 ; If not --> ADC_Loop
            bis.b     #CS,&P2OUT              ; Chip Select high
;
            call     #TX_ADC_2PC              ; ADC result --> PC
            jmp      Mainloop                  ; Repeat
;

;-----
Init_Sys;   Subroutine sets up Modules and Controls Registers
#DIVA1+RSEL2+RSEL0,&BCSCTL1
;-----
StopWDT     mov       #WDTPW+WDTHOLD,&WDTCTL   ; Stop Watchdog Timer
SetupBC     mov.b    #DIVA1+RSEL2+RSEL0,&BCSCTL1 ; ACLK/4 RSEL=5
SetupP1_2   bis.b    #TXD,&P1SEL                ; P1.1/TA0 for TXD function
            bis.b    #TXD,&P1DIR                ; TXD output on P1
SetupP2     bis.b    #CS,&P2OUT                ; CS, Set
            bis.b    #CS+CLK,&P2DIR            ; CS and Clk Output direction
SetupTA     mov      #TASSEL1+TACLK,&TACTL     ; SMCLK, clear timer
SetupC0     mov      #OUT,&CCTL0               ; TXD Idle as Mark
            call     #Delay                    ; Time for crystal to stabilize

            bis      #MC1,&TACTL               ; Start timer in Continuous Mode
            call     #Set_DCO                  ; Set DCO to target frequency
SetupWDT    mov      #WDT_ADLY_16,&WDTCTL      ; WDT 16ms*4 Interval Timer
            bis.b    #WDTIE,&IE1               ; Enable WDT Interrupt
            eint                    ; General Interrupt Enable
            ret                                ; Return from subroutine
;

;-----
TX_ADC_2PC; Subroutine: Send a CR and ADCData to PC/user
;-----
            mov      #CR,RXTXData              ; CR to UART buffer
            call     #TX_Byte                  ; CR --> PC/user
;

;-----
TX_Byte_ASCII; TX Byte from ADCData in two ASCII bytes.

```

```

;-----
        push    ADCData                ; transmit ..x. of value
        call    #NUM_ASCIR             ;
        push    ADCData                ; transmit ..x. of value
        call    #NUM_ASCIA             ;
        ret                                ; Return from subroutine
        ;
NUM_ASCIR;    Convert Numbers 0..f into ASCII left aligned 2(SP)
        rrc     2(SP)                   ; 1. and 3. pass
        rrc     2(SP)                   ;
        rrc     2(SP)                   ;
        rrc     2(SP)                   ;
        ;
NUM_ASCIA    and     #0fh,2(SP)         ; 2. and 4. pass
        add     #030h,2(SP)            ;
        cmp     #03ah,2(SP)           ;
        jlo     NUM_End                ;
        add     #039,2(SP)             ;
NUM_End      mov     2(SP),RXTXData     ; load transmit buffer, FALL
        mov     @SP+,0(SP)             ; Clean-up w\ return address TOS
        ;
;-----
TX_Byte;     Subroutine that TX Byte from RXTXData Buffer.
;-----
        mov     #TX_Count,BitCnt       ; TX_Count --> Branch Pointer
        push   &TAR                    ; Current TA Count
        add    #Bitime,0(SP)           ; Offset to next bit
        pop    &CCR0                   ; Time to next bit in CCR0
        mov    #OUTMOD2+OUTMOD0+CCIE,&CCTL0 ; TXD = Space Start Bit
TX_Wait      bit    #CCIE,&CCTL0       ; Wait for TX completion
        jnz    TX_Wait                 ;
        ret                                ; Return from subroutine
        ;
;-----
TA0_ISR     ;    CCR0/UART ISR:         RXTXData Buffer holds UART Data.
;-----
        add    #Bitime,&CCR0           ; Bitime till next bit
        mov    @BitCnt+,PC             ; Branch To Routine
        ;
TX_Bit      rra.b  RXTXData            ; LSB is shifted to carry
        jc     TX_Mark                 ; Jump if bit = 1
TX_Space     bis    #OUTMOD2,&CCTL0    ; TX Space
        reti                                ;
TX_Comp      bic    #CCIE,&CCTL0       ; All Bits RXed, Disable Interrupt
TX_Mark      bic    #OUTMOD2,&CCTL0    ; TX Mark
        reti                                ;
    
```

```

;
;
TX_Count    even
            DW      TX_Bit          ; TX First Data Bit
            DW      TX_Bit          ;
            DW      TX_Bit          ;
            DW      TX_Bit          ;
            DW      TX_Bit          ;
            DW      TX_Bit          ;
            DW      TX_Bit          ;
            DW      TX_Bit          ;
            DW      TX_Mark         ; TX Stop Bit= Mark
TX_End      DW      TX_Comp         ; TX Complete and Complete
;
;-----
WDT_ISR;    Exit LPM3 on reti
;
            bic     #LPM3,0(SP)     ; Clear LPM3 from TOS
            reti
;
;-----
Delay;      Software delay
;
            push   #0FFFFh         ; Delay to TOS
L1          dec    0(SP)           ; Decrement TOS
            jnz    L1              ; Delay over?
            incd   SP              ; Clean TOS
            ret
;
;-----
Set_DCO;    Subroutine: Sets DCO to selected frequency based on Delta.
;           R14 and R15 are used, ACLK= 32768/4 TA clocked by DCOCLK
;
            clr    R15              ;
Setup_CC2   mov    #CCIS0+CM0+CAP,&CCTL2 ; Define CCR2,CAP,ACLK
Test_DCO    bit    #CCIFG,&CCTL2    ; Test capture flag
            jz     Test_DCO         ;
            bic    #CCIFG,&CCTL2    ; Clear capture flag
;
AdjDCO      mov    &CCR2,R14        ; R14 = captured SMCLK
            sub    R15,R14          ; R14 = capture difference
            mov    &CCR2,R15        ; R15 = captured SMCLK
            cmp    #Delta,R14       ; Delta = SMCLK/(3278/4)
            jlo   IncDCO            ;
            jeq   DoneDCO           ;
DecDCO      dec.b  &DCOCTL          ;

```

```

        jmp      Test_DCO          ;
IncDCO   inc.b    &DCOCTL        ;
        jmp      Test_DCO          ;
DoneDCO  clr      &CCTL2         ; Disable CCR2
        ret      ; Return from subroutine
        ;
;-----
        RSEG    INTVEC          ; MSP430x11x1 interrupt vectors
;-----
        DW      RESET          ; no source
        DW      RESET          ; no source
        DW      RESET          ; P1.x
        DW      RESET          ; P2.x
        DW      RESET          ; no source
        DW      RESET          ; no source
        DW      RESET          ; no source
        DW      RESET          ; no source
        DW      RESET          ; Timer_A
        DW      TA0_ISR        ; Timer_A0
        DW      WDT_ISR        ; Watchdog/Timer, Timer mode
        DW      RESET          ; Comparator_A
        DW      RESET          ; no source
        DW      RESET          ; no source
        DW      RESET          ; NMI, Osc. fault
        DW      RESET          ; POR, ext. Reset, Watchdog
END
    
```

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		Applications	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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

Copyright © 2003, Texas Instruments Incorporated