

# DESIGNER'S NOTEBOOK



## TMS320C2x/'C5x EVM AIC Initialization and Configuration

Contributed by Thomas G. Horner, P.E.

### Design Problem

What are the issues in initializing a TMS320C2x/'C5x fixed-point EVMs?

### Solution

Texas Instruments' TMS320C2x and TMS320C5x EVMs come with 'C26/'C51 DSPs interfaced by serial port to a TLC32046 Wide-Band Analog Interface Circuit (AIC) with the AIC providing the frame sync pulses and shift clocks. The AIC is a configurable device which uses the serial port to download commands from the DSP. The communications protocol uses an interleaving technique which will not disrupt normal output of the DAC.

There are primary and secondary transmit data word formats. The primary data word is the normal data output format and the secondary data word carries configuration data to the AIC. Both data word formats use bits 0 – 1 to send commands to the AIC, while bits 2 – 16 are for either the data word (Primary) or configuration word (Secondary). A list of the functions is shown in Table 1 (Primary) and Table 2 (Secondary):

Table 1. Primary Data Word Commands		
D1	D0	Function
0	0	Normal Output
0	1	Increase Sample Rate
1	0	Decrease Sample Rate
1	1	Initiate Secondary Communications

Table 2. Secondary Data Word Commands		
D1	D0	Function
0	0	Update TA/RA Registers
0	1	Update TA'/RA' Registers
1	0	Update TB/RB Registers
1	1	Update Control Register

The timing between the Primary and Secondary data words is fairly tight for the 'C26. This design note is intended to clarify the technique required when reconfiguring the AIC using either the 'C2x or 'C5x EVMs.

The DSP and AIC use separate oscillators to generate their respective Master Clocks which introduces an additional constraint in the timing between the Primary and Secondary data words due to the potential phase offset between the two CLKs.

'C26 Master Clock = 40.000 MHz  
'C26 Instruction Cycle = 10 MHz

'C50 Master Clock = 20.000 MHz  
'C50 Instruction Cycle = 20 MHz

AIC Master Clock = 10.368 MHz  
AIC Shift Clock = 2.592 MHz  
= 3.86 'C26 Instruction Cycles  
= 7.72 'C50 Instruction Cycles

The maximum AIC conversion frequency is 25 KHz, which gives a minimum period of 40  $\mu$ sec between data samples. When the Primary data word command bits are set to 11b, the Secondary transmit frame sync pulse goes LOW FOUR AIC SHIFT CLOCKS after the end of the Primary transmission. This timing allows the secondary command communications to occur between normal data communications to the DAC on the AIC. To correctly reconfigure the AIC, the secondary command word must be written to the DSP's Serial Port Transmit Register (DXR) before the secondary frame sync pulse goes low. If the TRANSMIT interrupt is used to control writes to the DSP DXR during AIC configuration, the XINT signal occurs approximately 15 ('C26) or 154 ('C50) instruction cycles before the FSX goes LOW to signal start of transmission (best case—assuming that the DSP and AIC Master Clocks are in phase). If the Master Clocks are not in phase (high probability), then there will be 2 – 3 fewer instruction cycles available before FSX goes low again. The diagram in Figure 1 shows the timing between the Primary and Secondary AIC transmissions and the 'C26 and 'C50 XINT

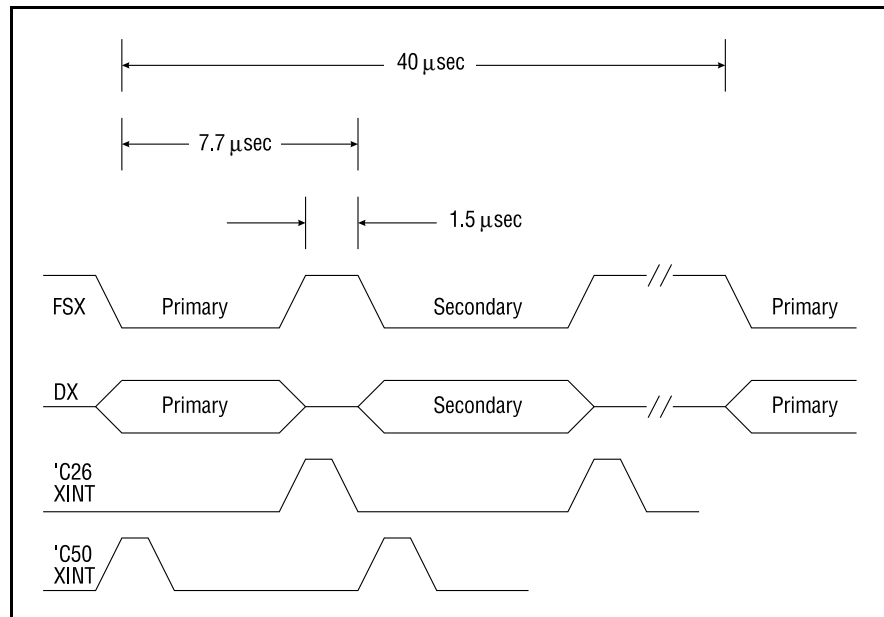


Figure 1. Timing diagrams

signal. Note that the 'C26 XINT occurs after all 16 bits have been shifted out of the transmit shift register (SXR), while in the 'C50, XINT occurs after the contents of the data register (DXR) are loaded into the SXR at the beginning of the transmission. This difference in serial port operation makes the timing on the 'C50 much easier to meet.

#### Software Examples:

The following software shows an example of how to reconfigure the AIC by writing to the AIC control register. This technique can be used for any secondary communications to the AIC. There is a latency of approximately 10/17 instruction cycles ('C26/'C51) from XINT to writing to the DXR.

```

;-----
;                                     'C26 PROGRAM
;-----
AIC_RESET_LO      .set      0ffefh    ;Define constants for AIC
AIC_RESET_HI      .set      010h
AIC_SETUP         .set      03h
AIC_CONTROL       .set      0277h
; Primary Transmit Data Word Format
;
;      d12
;      |d10 d6 d2
;      d14|d8 |d4 |d0
;      |  |  |  |  |
;      v v v v v v v v
;      xxxxxxxxxxxxxxx11b    Signals secondary Xmit mode
;
; Secondary Transmit Data Word Format
;
;                                     Secondary Command Syntax
;                                     (d1/d0 indicate mode)
;      |TA | |RA |
;      xx10010xx1001000b    TA and RA counter setup example
;                               valid range: 4-63
;      | TB | |RB |
;      x100100x10010010b    TB and RB counter setup example
;                               valid range: 15-127
;
;      | ctrl |
;      xxxxxx1010110111b    Control word setup example
;                               (0/1)
;      | | | | | | | | | |
;      | | | | | | | | | | d2 = (out/in) A/D highpass filter
;      | | | | | | | | | | d3 = (out/in) loopback function
;      | | | | | | | | | | d4 = (no/yes) Aux input pins
;      | | | | | | | | | | d5 = (no/yes) RX & TX synchronous
;      | | | | | | | | | | d7/d6 = Gain 0/0 = 1
;      | | | | | | | | | | 0/1 = 2
;      | | | | | | | | | | 1/0 = 4
;      | | | | | | | | | | 1/1 = 1
;      | | | | | | | | | | d8 = don't care
;      | | | | | | | | | | d9 = (out/in) second order sin x/x filter
;
;-----
;                                     MEMORY DEFINITION
;-----
;Reserve RAM for operands
.bss      AIC_CNTL, 1    ;AIC control temp memory

```

Figure 2. 'C26 program

```

;-----
;                               INTERRUPT VECTORS
;-----
        .sect      "ext_vecs"      ;Section for external
                                   ;interrupt vectors

        B          START           ;Processor Reset
        B          INT0            ;External Interrupt #0
        B          INT1            ;External Interrupt #1
        B          INT2            ;External Interrupt #2

        .sect      "int_vecs"      ;Section for internal
                                   ;interrupt vectors

        B          TINT            ;Timer Interrupt
        B          RINT            ;Serial Port Receive
                                   ;Interrupt
        B          XINT            ;Serial Port Transmit
                                   ;Interrupt
        B          TRAP            ;S/W Trap

;-----
;                               CODE
;-----
        .text                      ;Section for program code

START
        dint                      ;Global interrupt disable

;-----  INITIALIZE SERIAL PORT
        ldpc      0
        fort      0               ;Set for 16-bit word
                                   ;operation
        sfsm                      ;Set for frame sync control
        rtxm                      ;Set for external Xmit frame
                                   ;sync

;-----  INITIALIZE TLC32046 AIC
        ldpc      AIC_CNTL
        lalk      AIC_RESET_LO    ;Force AIC RESET low
        sac1      AIC_CNTL
        out       AIC_CNTL, PA2
        rptk      20              ;Keep LO for 2 usec
                                   ;(Spec=800 nsec min)

        nop
        ork       AIC_RESET_HI    ;Set AIC RESET high
        sac1      AIC_CNTL
        out       AIC_CNTL, PA2

        ldpc      0
        lack      020h           ;Enable transmit interrupt
        sac1      IMR
        eint                      ;Enable global interrupts
        zac                      ;Dummy Xmit to synchronize the
                                   ; 'C26 and AIC
        sac1      DXR             ;Signal secondary Xmit mode
        lalk      AIC_SETUP
        idle
        lalk      AIC_CONTROL     ;Send control word
        idle

```

Figure 2. 'C26 program (continued)

```

        dint                                ;Disable interrupts to reconfig
        lack      010h                     ;serial port
        sac1      IMR                       ;Enable serial port RECEIVE
        eint                                             ;interrupts
                                             ; REC = IMR b4
                                             ;Enable global interrupts

;----- MAIN ROUTINE
MAIN
        idle
        b      MAIN
;-----
;----- INTERRUPT SERVICE ROUTINES
;-----
RINT                                ;SERIAL PORT RECEIVE INTERRUPT
        ldpk      0
        lac      DDR, 4                   ;Read latest AIC input w/ 16x
                                             ;gain
        sac1      DXR                     ;Echo to AIC output
        eint                                             ;Re-enable GLOBAL interrupt
        ret                                             ;Return to MAIN

XINT                                ;SERIAL PORT TRANSMIT INTERRUPT
        sac1      DXR                     ;Write to DXR register
        eint
        ret

;----- UNUSED INTERRUPT TRAPS
INT0    idle                           ;External Interrupt #0
INT1    idle                           ;External Interrupt #1
INT2    idle                           ;External Interrupt #2
TINT    idle                           ;Timer Interrupt
;RINT    idle                           ;Serial Port Receive Interrupt
;XINT    idle                           ;Serial Port Transmit Interrupt
TRAP    idle                           ;S/W Trap
        .end

;*****
;*****

```

Figure 2. 'C26 program (continued)

#### NOTE:

The 'C5x EVM requires the following command be incorporated into the EVMINIT.CMD file to force the 'C50 into microprocessor mode:

```
E      PMST=0x08
```

In addition, whenever the processor is RESET from within the Debugger and software reloaded, you need to issue the following command to get the 'C50 back into microprocessor mode:

```
?PMST=0x08
```



```

;-----
;                               INTERRUPT VECTORS
;-----
        .sect      "vectors"      ;Section for external interrupt
;vectors
        B          START          ;Processor Reset
        B          INT0           ;External Interrupt #0
        B          INT1           ;External Interrupt #1
        B          INT2           ;External Interrupt #2
        B          TINT           ;Timer Interrupt
        B          RINT           ;Serial Port Receive Interrupt
        B          XINT           ;Serial Port Transmit Interrupt
        B          TRINT          ;TDM Serial Port Receive
;Interrupt
        B          TXINT          ;TDM Serial Port Transmit
;Interrupt
        B          INT3           ;External Interrupt #3
        .space     10*16          ;Reserved space - 10 words
        B          TRAP           ;S/W Trap
        B          NMI            ;Non-maskable external interrupt
;-----
;                               CODE
;-----
        .text                    ;Section for program code

START
        ldp        #0             ;Initialize data pointer
        setc       INTM           ;Global interrupt disable.
        splk       #0h,IMR        ;Clear interrupt mask register

;-----  SETUP S/W WAIT-STATE GENERATOR
        splk       #08, CWSR      ;Normal wait-state mapping,
;I/O space=64K
        splk       #0, PDWSR      ;Prog/Data space=0 wait-states
        splk       #05555h, IOWSR ;I/O space=1 wait-states

;-----  INITIALIZE SERIAL PORT
        ldp        #0
        splk       #08h, SPC      ;DBL=0 (b1): loopback mode
;disabled
;FO=0 (b2): 16-bit word
;operation
;FSM=1 (b3): frame sync control
;MCM=0 (b4): external CLKX
;TXM=0 (b5): external FSX
;XRST=0 (b6): Xmit RESET
;RRST=0 (b7): Rec RESET
        opl        #0c0h, SPC     ;XRST=1 (b6): Xmit ENABLE
;RRST=2 (b7): Rec ENABLE

;-----  INITIALIZE TLC32046 AIC
        ldp        AIC_CNTL
        lacc       #AIC_RESET_LO ;Force AIC RESET low
        sac1       AIC_CNTL
        out        AIC_CNTL, PA2
        rpt        #40            ;Keep LO for 2 usec
; (Spec=800 nsec min)
        nop

```

Figure 3. 'C50 program (continued)

```

        or          #AIC_RESET_HI ;Set AIC RESET high
        sac1        AIC_CNTL
        out         AIC_CNTL, PA2

        ldp         #0
        opl         #020h,IMR      ;Enable Xmit interrupt bit
        opl         #0h,IFR        ;Clear pending interrupts.
        clrc        INTM           ;Global interrupt enable.
        lacl        #0             ;Dummy Xmit for
                                   ;synchronization

        sac1        DXR
        lacc        #AIC_SETUP     ;Signal secondary Xmit mode
        idle
        lacc        #AIC_CONTROL   ;Send control word
        idle

        setc        INTM           ;Global interrupt disable.
        lacl        #010h          ;Enable serial port RECEIVE
                                   ;interrupts

        sac1        IMR            ; REC = IMR b4
        opl         #0h, IFR        ;Clear pending interrupts.
        clrc        INTM           ;Global interrupt enable.

;-----      MAIN ROUTINE
MAIN
        idle
        b          MAIN

;-----
;-----      INTERRUPT SERVICE ROUTINES
;-----
RINT          ;SERIAL PORT RECEIVE INTERRUPT
        ldp         #0
        lacc        DRR, 4         ;Read latest AIC input w/ 16x
                                   ;gain
        samm        DXR            ;Echo to AIC output
        rete        ;Return to MAIN w/ interrupt
                                   ;enable
        XINT        ;SERIAL PORT TRANSMIT
                                   ;INTERRUPT
        samm        DXR
        rete

;-----      UNUSED INTERRUPT TRAPS
INT0          idle                ;External Interrupt #0
INT1          idle                ;External Interrupt #1
INT2          idle                ;External Interrupt #2
TINT          idle                ;Timer Interrupt
;RINT         idle                ;Serial Port Receive
                                   ;Interrupt
;XINT         idle                ;Serial Port Transmit Interrupt
TRINT         idle                ;TDM Serial Port Receive
                                   ;Interrupt
TXINT         idle                ;TDM Serial Port Transmit
                                   ;Interrupt
INT3          idle                ;External Interrupt #3
TRAP          idle                ;S/W Trap
NMI           idle                ;Non-maskable external interrupt
        .end

```

Figure 3. 'C50 program (continued)