TMS320 DSP
DESIGNER'S NOTEBOOK
Number 66

Interfacing a TMS320C2x, 'C2xx, or 'C5x DSP to a TLC548 8-bit A/D convertor

Contribued by Gary Przyborski, Advantor Corp.

Design Problem How do you interface a 'C2x, 'C2xx, or 'C5x DSP to a TLC548 8-bit A/D convertor?

Solution In referencing the Texas Instruments *Data Acquisition & Conversion Applications Manual*, a suggested hookup for this A/D to a TMS32020 is shown. The following suggestion uses the processing power of the DSP plus the addition of one octal register, 74HC377, placed in I/O space to eliminate the 74LS175, 74LS74, and the two 74LS02 gates used in that design, shown on page 1-213. Connect the low data bus to the 377 input, the clock enable of the 377 to $\overline{IO}$, the clock of the 377 to $\overline{WE}$ of the processor, and the A2dOUT from the A/D to CLKR on the DSP. This approach gives the design six spare-registered outputs and uses less than 5 microseconds of DSP time per A/D conversion. A TMS320C52 can produce a clock at the maximum 2-MHz rate of the TLC548 (see *Linear Circuits*, volume 2 for a data sheet).

```
*     Definitions...
*     A2dCs on register bit 7 of the output register
*     A2dClk on register bit 2 of the output register
OutReg          .set    50h     ;MMIOPort w/ data
A2dOut          .set    7       ;bit placement for a2dout pin
A2dCsMask       .set    0FF7Fh  ;and out chip select on a2d to 0
A2dCsHigh       .set    00080h  ;or for chip sel on a2d to 1

*       Data setup BLOCK B2...
RegMem          .space  10h     ;Memory copy of register
ClkHI           .space  10h     ;A2d clock high
ClkLO           .space  10h     ;A2d clock low
A2dClk          .space  10h     ;Mask for A2d Clock

*       Initialize A2dClk...
        ldp     #0
        splk    #00004h,A2dClk  ;Initialize A2dClk space
```

Figure 1. Example code for a TMS320C52

```
*       BLOCK B0 Program...
*       In the interrupt routine start the converter going near the
*       start of the service routine.
        ldp     #0
        lacl    RegMem          ;Start A2d converting...
        and     #A2dCsMask
        sacl    ClkLO           ;Save for A2d routine, cs=0, clk=0
        samm    OutReg
        .....                   ;your code, to meet requirement below

*       Verify that at least 1.4us has elapsed from the last
*       instruction above until the out instruction at ReClock.
*       This is required to meet the TLC548's CS* to the first
*       I/O clock time. OutReg is the address for the output
*       register, with data, assumes data page = 0.

A2dConv lacl    sreg0           ; Restore value sent to OutReg
        or      A2dClk
        sacl    ClkHI           ; With Clock "high"
        lacl    #7              ; Generate 8 clocks
        samm    brcr
        lacl    #0
        sacb                    ; Initialize A2d value
        rptb    endclk          ; Symmetrical clock @ 2MhZ
ReClock out     ClkHI, OutReg   ; Clock high...
        lamm    SPC
        bsar    8
        out     ClkLO, OutReg   ; Force clock low...
        ror
endclk  rolb
        lacl    RegMem
        or      #A2dCsHigh      ;Force cs back high...
        samm    OutReg
        lacb                    ;A2d value...
```

Figure 1. Continued

Hookup the DOUT pin of the converter to the CLKR pin on the DSP. Hookup the OCLK and $\overline{CS}$ of the converter to two separate output registered pins A2DCLK and $\overline{A2DCS}$, respectively.

In conclusion, this technique has been used successfully in applications where low cost is of prime concern.