

*TMS320 DSP
DESIGNER'S NOTEBOOK*

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

APPLICATION BRIEF: SPRA262

*Gary Przyborski
Advantor Corporation*

October 1995



IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software 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 application 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, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

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

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI 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.

TRADEMARKS

TI is a trademark of Texas Instruments Incorporated.

Other brands and names are the property of their respective owners.

CONTACT INFORMATION

US TMS320 HOTLINE	(281) 274-2320
US TMS320 FAX	(281) 274-2324
US TMS320 BBS	(281) 274-2323
US TMS320 email	dsph@ti.com

Contents

Abstract.....	7
Design Problem	8
Solution	8

Examples

Example 1. Example code for a TMS320C52	8
---	---

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



Abstract

This document describes how to interface a TMS320C2x, 'C2xx, or 'C5x DSP to a TLC548 8-bit A/D converter. A lengthy code example is provided.



Design Problem

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

Solution

In referencing the Texas Instruments "Data Acquisition & Conversion Applications Manual", a suggested hookup for the TLC548 8-bit A/D converter 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 IO, the clock of the 377 to 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).

Example 1. Example code for a TMS320C52

```
*      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

*      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
sac1     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
          sac1 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...
```

Hookup the DOUT pin of the converter to the CLKR pin on the DSP. Hookup the OCLK and CS of the converter to the two separate output register pins A2DCLK and A2DCS, respectively.

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