

## Interfacing the LTC1090 to the TMS7742 MCU

Guy Hoover  
 William Rempfer

### Introduction

This application note describes an interface between the LTC1090 10-bit data acquisition system and the TMS7000 family of microcomputers (e.g., TMS7742). The simple four wire interface is capable of completing a 10-bit conversion and shifting the data to the TMS7742 in 103 $\mu$ s. Configuration of the LTC1090 and the TMS7742 will be discussed as it applies to this interface. Schematics, code, and timing diagrams will be shown. Finally, a summary of the key points of this interface will be given including data throughput rates.

### Interface Details

The LTC1090 has two clock lines: ACLK and SCLK. ACLK controls the A/D conversion rate while SCLK controls the data shift rate. Data is transferred in a full duplex format over D<sub>IN</sub> and D<sub>OUT</sub>.

The TMS7742 contains a synchronous, full duplex, serial port that allows the user to construct a simple communication path to the LTC1090. The serial port provides clock, transmit and receive lines that are compatible with the LTC1090. The only additional line required is one pro-

grammable output pin (A0) to control  $\overline{CS}$  on the LTC1090. The schematic of Figure 1 shows how the two devices are connected.

### Hardware Description

The TMS7742 was chosen because it contains 4k of EPROM which can be programmed using a standard EPROM programmer. Any member of the TMS7000 family which contains a serial port should be able to use this code with only modifications to the peripheral register numbers.

The timing diagram of Figure 2 was obtained using an HP1631A logic analyzer. ACLK of the LTC1090 was 2 MHz and the TMS7742 clock was 5 MHz.

The analog section of the schematic of Figure 1 is omitted for clarity. For a complete discussion of the analog considerations involved in using the LTC1090 please see the data sheet.

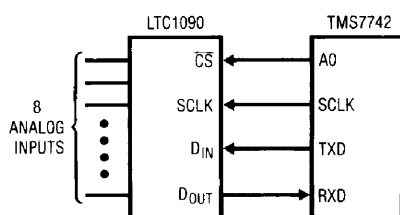


Figure 1. Schematic

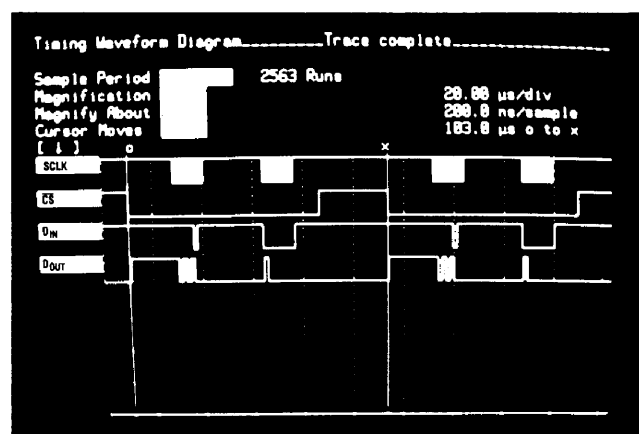
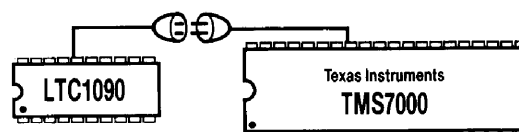


Figure 2. Timing Diagram

# Application Note 26E

## Software Description

The software configures and controls the serial port of the TMS7742. Additionally, the software manipulates A0 ( $\overline{CS}$  of the LTC1090) and generates a delay during which time the LTC1090 performs a conversion.

The code first disables all interrupts and initializes the stack. Next the serial port is configured. Tx is enabled, the serial port is reset, and the SMODE register is configured for 8 bits, no parity, and one stop bit. The SCLK rate is set to the processor clock frequency divided by 4. The  $D_{IN}$  word of the LTC1090 is next loaded into the ACC. This  $D_{IN}$  word (\$DF) configures the LTC1090 for CH7 with respect to COM, unipolar mode, LSB first and a 16-bit word length. Examine Figure 3 to see how this is constructed keeping in mind that the TMS7742 transmits data LSB first.

A subroutine SXTNBIT is called next. This is a routine that does the actual data shifting. A0 ( $\overline{CS}$ ) is cleared. Then, the LTC1090  $D_{IN}$  word is placed into the transmit buffer. The serial port is turned on and the data is shifted while the processor idles in a loop. The first eight bits containing the LSBs are then placed in the B register. The procedure is repeated for the next eight bits which contain the two MSBs and the result is placed in the A register. A0 ( $\overline{CS}$ ) is then set and the subroutine returns to the original program. The data in the A and B registers is then stored in R5 and R6.

At this time 44 ACLK cycles must be allowed for the A/D to perform its next conversion. Enough time is consumed by this program however that no additional delay for the conversion is required.

1	1	0	1	1	1	1	1	P5
WL0	WL1	MSBF	UNI	S2	S1	O/S	S/D	

Figure 3.  $D_{IN}$  Word for LTC1090 Stored in TMS7742 RAM

								LSB	
LSB	7	6	5	4	3	2	1	0	R5
fill with zeroes								MSB	
MSB	0	0	0	0	0	0	9	8	R6

$D_{OUT}$  from LTC1090 stored in TMS7742 RAM

Figure 4. Memory Map

LABEL	MNEMONIC	COMMENTS
START	DINT	DISABLES ALL INTERRUPTS
	MOVP % > 2A, P0	DISABLE INTERRUPT FLAGS
	MOVP % > 02, P16	DISABLE INTERRUPT FLAGS
	MOV % > 60, B	ADDRESS OF STACK
	LDSP	PUT ADDRESS INTO POINTER
	MOVP % > DF, P5	CONFIGURE PORT A
	MOVP % > 08, P6	ENABLE Tx BY SETTING B3 = 1
	MOVP % > 00, P17	P17 POINTS TO SCTL0
	MOVP % > 40, P17	RESET THE SERIAL PORT
	MOVP % > 0C, P17	CONFIGURE THE SERIAL PORT
	MOVP % > 00, P21	TURN START BIT OFF
	MOVP % > 00, P17	ENABLE THE SERIAL PORT
	MOVP % > 00, P20	SET SCLK RATE (TIMER 3)
	MOVP % > C0, P21	START TIMER
LOOP	MOV % > DF, A	LOAD LTC1090 $D_{IN}$ WORD IN A
	CALL SXTNBIT	ROUTINE THAT SHIFTS DATA
	MOV B, R5	PUT FIRST 8 LSBs IN R5
	MOV A, R6	PUT MSBs IN R6
SXTNBIT	ANDP % > FE, P4	A0 CLEARED ( $\overline{CS}$ GOES LOW)
	MOVP A, P23	PUT LTC1090 $D_{IN}$ INTO TXBUF
	MOVP % > 40, P21	SCLK OFF (TIMER 3 DISABLED)
	MOVP % > 17, P17	ENABLE SERIAL PORT
	MOVP % > C0, P21	SCLK ON (TRANSFER BEGINS)
	MOVP % > 14, P17	TXEN GOES LOW
	MOV % > 02, A	LOAD COUNTER
WAIT1	DJNZ A, WAIT1	LOOP WHILE SHIFT OCCURS
	NOP	DELAY
	MOVP P22, B	PUT $D_{OUT}$ FROM LTC1090 IN B
	MOVP A, P23	LOAD TXBUF
	MOVP % > 40, P21	SCLK OFF (TIMER 3 DISABLE)
	MOVP % > 17, P17	ENABLE SERIAL PORT
	MOVP % > C0, P21	SCLK ON (TRANSFER BEGINS)
	MOVP % > 14, P17	TXEN GOES LOW
	MOV % > 02, A	LOAD COUNTER
WAIT2	DJNZ A, WAIT2	LOOP WHILE SHIFT OCCURS
	NOP	DELAY
	MOVP P22, A	PUT $D_{OUT}$ FROM LTC1090 IN A
	ORP % > 01, P4	A0 SET ( $\overline{CS}$ GOES HIGH)
	RETS	RETURN TO MAIN PROGRAM

Figure 5. TMS7742 Code

## Summary

A four wire interface between the LTC1090 and the TMS7742 with a combined data conversion and transfer time of 103 $\mu$ s was demonstrated. The interface used the serial port of the TMS7742. Because the serial port transfers data LSB first, care must be taken to properly construct the  $D_{IN}$  word so that the bits are transmitted in the proper order to the LTC1090. The 10 data bits of the LTC1090 are shifted LSB first in two eight bit transfers. The data is stored right justified in the TMS7742's internal RAM.