



# THE MC68230 PARALLEL INTERFACE/TIMER PROVIDES AN EFFECTIVE PRINTER INTERFACE

Prepared by  
Victor A. Scherer  
and  
William G. Peterson  
Microprocessor Group  
Austin, Texas

## INTRODUCTION

The MC68230 Parallel Interface/Timer (PI/T) provides versatile parallel interface ports and an operating system-oriented timer to MC68000 microprocessor systems. This application note describes a PI/T interfaced with a Printronix Model 300 printer using the industry-standard Centronics interface.

## THE PRINTRONIX INTERFACE

The standard Centronics printer interface allows transfer of byte-wide, parallel data transfer under control of two real-time handshake lines, DATA STROBE, and ACKNLG (acknowledge). Short data setup (50 ns) and hold times, with respect to the active edge of the DATA STROBE input, are required.

Three additional printer status signals, BUSY, PE (printer error), and SLCT (select) are provided to maintain printer status. A timing diagram for the Centronics interface implemented in the Printronix 300 is shown in Figure 1. When PE is high, the printer is in check (disabled due to an internal condition such as out of paper or malfunction). When SLCT is high, the printer is on line and not in check.

When the printer is initially ready to receive data, its PE (printer error) output furnishes a logic low (inverted and applied to PC0 of the PI/T) and its SLCT (select) line furnishes a logic high (inverted and applied to PC1 of the PI/T). The PI/T then outputs data (on the PA0-PA6 lines) plus a logic low data strobe pulse (on the H2 pin). Once the printer accepts the data character, it returns a low ACKNLG pulse (inverted at PI/T H1). This indicates that the printer is ready for the next character. If the printer is currently unavailable, due to a form-feed or printing operation, it outputs a high BUSY signal (which is inverted before application to the

PI/T on the PA7 line); however, in this case it does not output the ACKNLG pulse until it (the printer) is again available.

## USING THE PI/T

The PI/T requires only buffer and inverter devices to connect directly to the printer as shown in the schematic diagram of Figure 2. This implementation does not use the optional Printronix expanded character set; thus, only seven data bits need be connected. Seven data bits are sufficient for ASCII characters and for use in the plot mode. The PI/T is programmed for the unidirectional 8-bit mode, with double-buffered output transfers chosen for Port A. (Port B, H3, and H4 are not used and are totally independent of the printer interface; therefore, a second printer could be attached.) Port A bits 0 through 6 are double-buffered by the PI/T and are applied to the printer by a 74LS244 driver. Port A bit 7 is not required as a data output in this application and is used as the BUSY input from the printer. (When double-buffered output transfers are chosen, input pins such as PA7 are unbuffered and the instantaneous level of the pin may be read by the microprocessor.) Depending upon operating system requirements, BUSY, PE, and SLCT may or may not be useful and are not rigorously supported in the driver software provided in Figure 3.

The H2 pin is configured, for use with the PI/T, in the pulsed output handshake protocol. This pin produces a 4-clock cycle pulse (500 nanoseconds @ 8 MHz) whenever new data is available at the Port A output pins. This pulse is received by the printer as its DATA STROBE input and is used to latch the newly available character. The low active ACKNLG output from the printer is then received by a 74LS14 and applied as an inverted input to the edge-sensitive

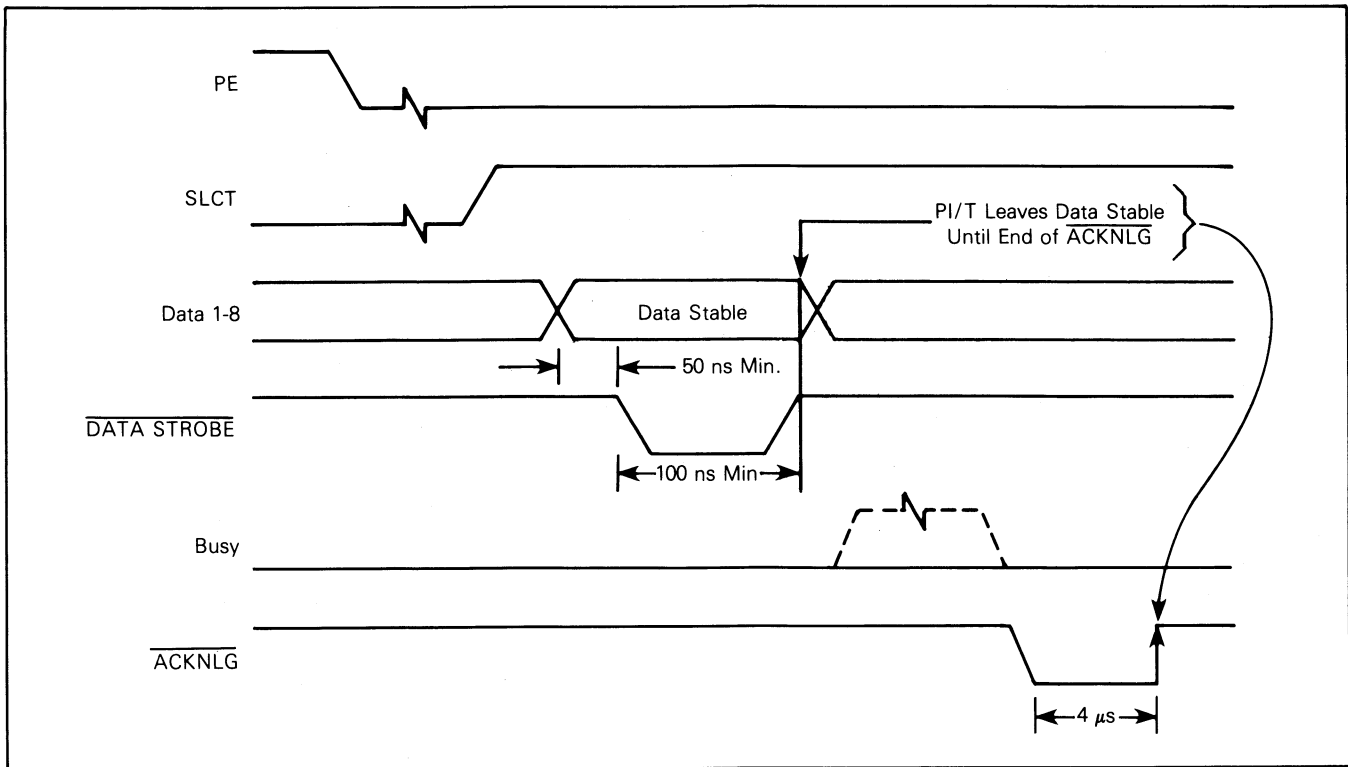


FIGURE 1 — Centronics Interface, Timing Requirements

H1 handshake pin of the PI/T. An active-going edge (in this case the trailing edge, as shown in Figure 2) on H1 indicates that the printer is ready to accept the new data. Each of the PI/T handshake pins (H1-H4) may be programmed independently as either active high or active low inputs. In this application, to satisfy printer timing requirements, H1 was chosen (by the Port General Control Register configuration) to recognize the trailing edge of the 4-microsecond  $\overline{\text{ACKNLG}}$  pulse. Refer to the timing diagram of Figure 1.

#### DOUBLE-BUFFERING

The PI/T provides the full double-buffered data path support required in this application. The best throughput to the printer is obtained if the data transferring device (the PI/T) is always ready to pass the next character to the printer whenever the  $\overline{\text{ACKNLG}}$  pulse is received. The PI/T contains two fully-static 8-bit latches in the output data path of each port. Thus, assuming both are full, when the  $\overline{\text{ACKNLG}}$  pulse is recognized by the active-going edge on H1, the next character is immediately moved to the pins without waiting for interrupt or direct memory access latency. At that time, an interrupt or direct memory access request (as programmed) is made to fill the remaining buffer(s). To take best advantage of multiple buffering levels, I/O driver software should check the H1S status (bit 0 in the Port Status Register) after each character is stored to verify that room is not available for an additional character. The driver should return only when the data path is full. Somewhat primitive but functional driver software is shown in Figure 3.

#### PI/T — PRINTRONIX DRIVER SOFTWARE

This group of routines constitutes the device driver portion of a line printer controller. Some routines are not shown (such as LPCLOSE), but the actual device interface code is

illustrated. The basic sequence of operations is as follows:

- 1) When the system is booted, LPOPEN is called by a server process. This routine is called only once.
- 2) When the server process has output to print, it makes successive calls to LPWRITE, passing the buffer address and character count. LPWRITE returns status in D0.
- 3) LPWRITE enables interrupts after checking the printer status. The PI/T generates an interrupt as soon as interrupts are enabled.
- 4) The LPINTR interrupt routine then initiates the output of characters to the printer. After each character is received by the printer, an  $\overline{\text{ACKNLG}}$  signal is sent back to the PI/T. This initiates moving another character to the output lines and also initiates movement of a new character to the double-buffered input. Thus, the double-buffering scheme of the PI/T will avoid many unneeded interrupts. In fact, tests have shown that the PI/T and printer can usually keep up with the processor and need only interrupt after a line feed is output.

This simple routine requires modification in order to be integrated into a real operating system. After the line labeled lpwgo is executed, the program loops while waiting for the finished flag (finflag) to be set. At this point, a semaphore call may be made to suspend the LPWRITE process. When all characters are printed, LPINTR will wake-up LPWRITE, which will return. In addition, more complicated buffering is generally used.

The MC68230 is as easy to program as most line printer controllers and is much cheaper. The program of Figure 3 demonstrates that the MC68230 is easy to use in this common application.

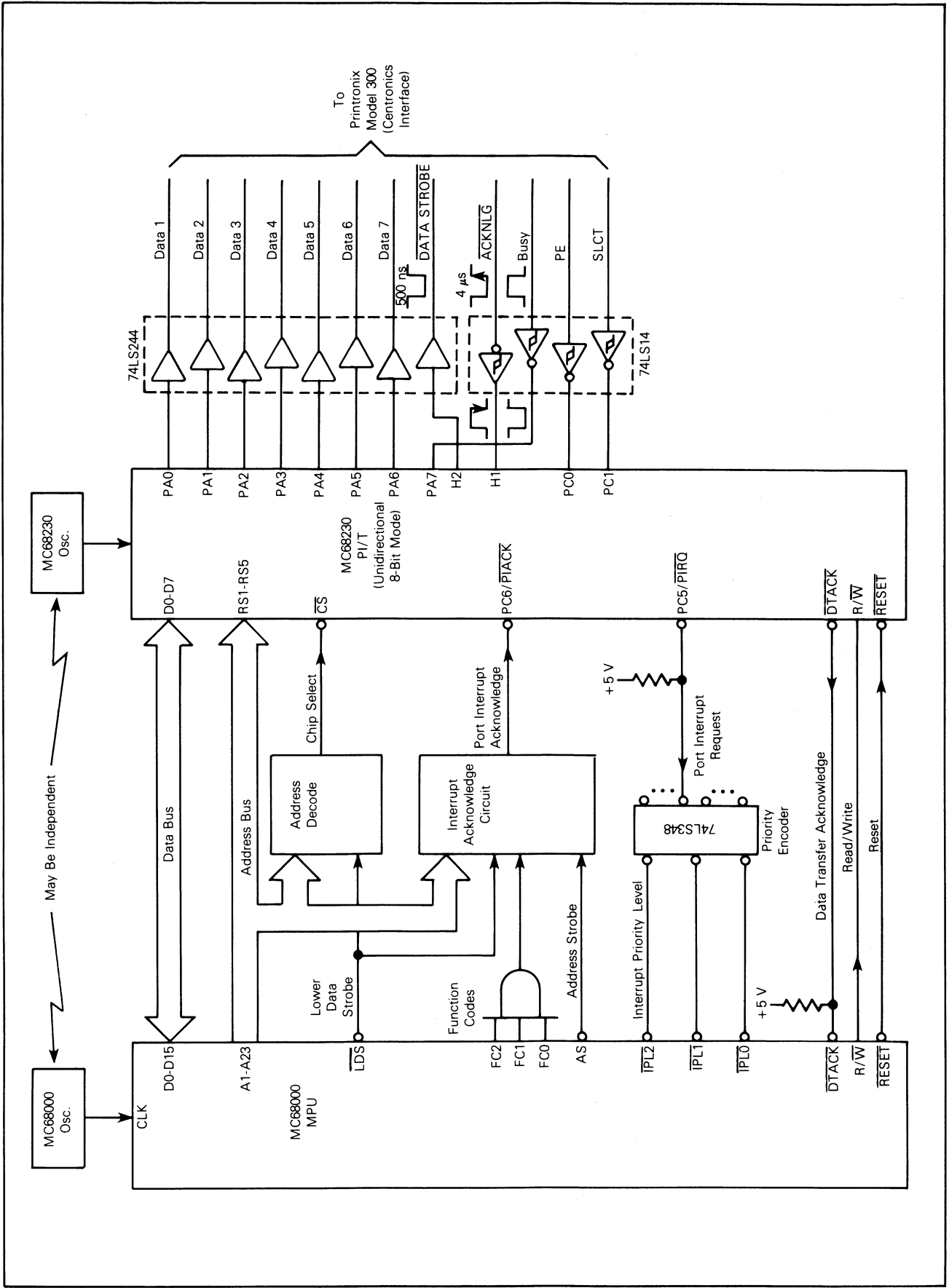


FIGURE 2 — MC68000-PI/T-Printer Interface Schematic Diagram

```

*
*   MC68000 - MC68230 PI/T LINE PRINTER DRIVER
*
*   MICROPROCESSOR DIVISION, MOTOROLA INC.
*   VICTOR SCHERER AND WILLIAM G. PETERSON
*   AUGUST 13, 1981
*
*   PRINTRONIX PRINTER, CENTRONICS INTERFACE
*
PIT          EQU          $C000          BASE ADDRESS FOR PI/T
PGR          EQU          PIT+1          PORT GENERAL CONTROL REGISTER
PSRR         EQU          PIT+3          PORT SERVICE REQUEST REGISTER
PADDR        EQU          PIT+5          PORT A DATA DIRECTION REGISTER
PIVR         EQU          PIT+$B        PORT INTERRUPT VECTOR REGISTER
PACR         EQU          PIT+$D        PORT A CONTROL REGISTER
PADR         EQU          PIT+$11       PORT A DATA REGISTER
PCDR         EQU          PIT+$19       PORT C DATA REGISTER
PSR          EQU          PIT+$1B       PORT STATUS REGISTER

                ORG          $800

* -----
*
*   LPOPEN -- CALLED ONCE BY A PRINTER SERVER ROUTINE
*           SETS UP PI/T FOR UNIDIRECTIONAL 8-BIT MODE, PORT A OUTPUT,
*           H2 PULSED OUTPUT HANDSHAKE PROTOCOL
*
LPOPEN:        ST          FINFLAG          $FF = FINISHED, IDLE
                MOVE.B    #$7F, PADDR      7 BITS OUT, HIGH BIT IN
                MOVE.B    #$7B, PACR      PORT A SUBMODE 01, PULSED
                MOVE.B    #$10, PGR       ENABLE PORT A, MODE 0
                MOVE.B    #$40, PIVR      INTERRUPT VECTOR
                MOVE.B    #$1B, PSRR      ENABLE INTRPINS
                RTS

* -----
*
*   LPWRITE -- USER EXECUTES TRAP INSTRUCTION, TRAP HANDLER SETS UP
*            PARAMETERS : DO = BYTECOUNT, AO = BUFFER ADDRESS
*            IF PRINTER IS ONLINE, ROUTINE JUST ENABLES INTERRUPTS
*            DO = RETURN STATUS
*
LPWRITE:       CLR.B      FINFLAG          JUST STARTING
                MOVE.L    DO, BYTECNT     SAVE USER PARAMS
                MOVE.L    AO, BUFFADDR
                BTST      #0, PCDR        IN CHECK?
                BEQ.S     NOGD
                BTST      #1, PCDR        ON LINE?
                BEQ.S     LPWGD

NOGD:         ST          DO              SET TO ALL 1'S
                RTS

LPWGD:        BSET      #1, PACR          ENABLE H1S INTERRUPT
LPW1:         TST.B     FINFLAG          WAIT FOR FINFLAG EQU $FF
                BEQ.S     LPW1            OS BUFFERING HERE
                CLR.B     DO              NORMAL STATUS
                RTS          RETURN

* -----

```

FIGURE 3 — Printronix Driver Software

```

*
*      LPINTR -- INTERRUPT SERVICE ROUTINE. GETS CHARS FROM BUFFER AND
*              SENDS TO PI/T TO PRINT.
*              WHFN DONE, DISABLES INTERRUPTS.
*
LPINTR:      MOVE. L      AO, -(SP)          SAVE AO
              MOVE. L      BUFFADDR, AO      GET BUFFER ADDRESS
              TST. L        BYTECNT          IS THERE ANYTHING TO PRINT?
              BEQ. S        EMPTY            IF NOT, FINISHED

PRINTSOME:   MOVE. B      (AO)+, PADR        MOVE TO PI/T
              SUBQ. L      #1, BYTECNT      DEC CHAR COUNTER
              BEQ. S        EMPTY            STOP IF OUT OF CHARS
              BTST         #0, PSR          SEE IF ROOM FOR ANOTHER CHARACTER
              BNE. S        PRINTSOME       IF SO, DO IT AGAIN
              BRA. S        NOTREADY        NOT RDY

EMPTY:       BCLR         #1, PACR          DISABLE HIS INTERRUPTS
              ST           FINFLAG         SET FINISHED STATUS

NOTREADY:    MOVE. L      AO, BUFFADDR      SAVE BUFFER ADDRESS
              MOVE. L      (SP)+, AO       RESTORE AO
              RTE

BUFFADDR     DC. L        0
BYTECNT      DC. L        0
FINFLAG      DC. B        0

              END

```

FIGURE 3 — Printronix Driver Software  
(Concluded)



**MOTOROLA** *Semiconductor Products Inc.*

3501 ED BLUESTEIN BLVD., AUSTIN, TEXAS 78721 • A SUBSIDIARY OF MOTOROLA INC.