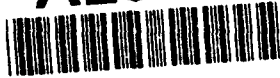


AD-A263 468



WL-TM-93-102

2



**LOW COST INSTRUMENTATION:
PARALLEL PORT ANALOG TO
DIGITAL CONVERTER**

Matthew P. Dierking

WL/AAWP-3

3050 C Street

Wright-Patterson AFB, Ohio 45433-7300

February, 1993

Technical Memorandum for the Period
July 1992 to December 1992

Electro-Optics Group
Passive ECM Branch
Electronic Warfare Division

Approved for public release;
Distribution is unlimited.

Avionics Directorate
Wright Laboratory
Wright-Patterson AFB, Ohio 45433-7300

DTIC
ELECTE
MAY 05 1993
S B D

93-09444



1208

93

5

00

9

FOREWORD

This technical memorandum was prepared by Matthew P. Dierking. This report documents the development of a low cost, highly portable instrumentation system for use by the Electro-Optics Laboratory, Passive ECM Branch, Electronic Warfare Division, Wright-Patterson AFB, OH 45433-7300. This development was performed in support of A-Lambda, an in-house development program investigating techniques for laser wavelength identification for laser warning receivers under work unit 76331211, E-O Measurements.

This Technical Memorandum has been reviewed and approved.


MATTHEW P. DIERKING
Electro-Optics Group, EW Division
Avionics Directorate


Paul S. Hadorn, PHD, Chief
Passive ECM Branch, EW Division
Avionics Directorate


JOHN H. SEWARD JR., Lt Col
Chief, Electronic Warfare Division
Avionics Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE February 1993	3. REPORT TYPE AND DATES COVERED INTERIM July 1992 to December 1992
4. TITLE AND SUBTITLE Low Cost Instrumentation: Parallel Port Analog to Digital Converter		5. FUNDING NUMBERS 62204F 7633 1211
6. AUTHOR(S) MATTHEW P. DIERKING		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Avionics Directorate Wright Laboratory Air Force Materiel Command Wright-Patterson AFB OH 45433-7409		8. PERFORMING ORGANIZATION REPORT NUMBER WL-TM-93-102
9. SPONSORING MONITORING AGENCY NAME(S) AND ADDRESS(ES) Avionics Directorate Wright Laboratory Air Force Materiel Command Wright-Patterson AFB OH 45433-7409		10. SPONSORING MONITORING AGENCY REPORT NUMBER WL-TM-93-102

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The personal computer (PC) has become a powerful and cost effective computing platform for use in the laboratory and industry. This Technical Memorandum presents the use of the PC parallel port adapter to implement a low cost analog to digital converter for general purpose instrumentation and automated data acquisition.

14. SUBJECT TERMS

Electronic Instrumentation
Automated Data Acquisition
Computer Interface

15. NUMBER OF PAGES

10

16. DISTRIBUTION CODE

SECURITY CLASSIFICATION
OF ABSTRACT

UNCL

UNCL

UNCL

UL

LOW COST INSTRUMENTATION: PARALLEL PORT ANALOG TO DIGITAL CONVERTER

1.0 BACKGROUND

The personal computer (PC) has become a powerful and cost effective computing platform, and as the PC's capabilities have expanded, so has its use in the laboratory and industry. PCs are used for both data acquisition and control. A wide variety of standard instruments support PCs, usually through plug-in cards or remote devices. However, in many cases, the use of such standard instruments is infeasible or impractical due to the installation requirements, movement, and/or cost of the devices. To facilitate better mobility and cost effectiveness, this development utilizes the standard Centronics parallel port for instrumentation purposes. The parallel port is extremely attractive for this purpose since it is standard on virtually all PC's, and use of the port provides direct, moderate speed access through external connections. This memo demonstrates the port's use through the implementation of a high speed analog to digital converter (ADC) for data acquisition and process monitoring. The original intent of this design was as a sub-component test apparatus for an in-house laser warning wavelength identification program.

2.0 THE PARALLEL PORT (IBM Printer Adapter)

The IBM parallel port adapter was specifically designed to communicate with printers, but it can be used for general purpose input/output (I/O) as long as the external device I/O requirements match the parallel port capabilities. Figure 1 shows a block diagram of the parallel port adapter. The port has 12 TTL-buffer output lines which can be read and written under software control, and has five read-only steady state-input lines. One I/O line can be used as a processor interrupt and can be enabled and disabled under software control. The data lines are latched into TTL totem pole outputs, and the status lines are implemented using open collector gates.

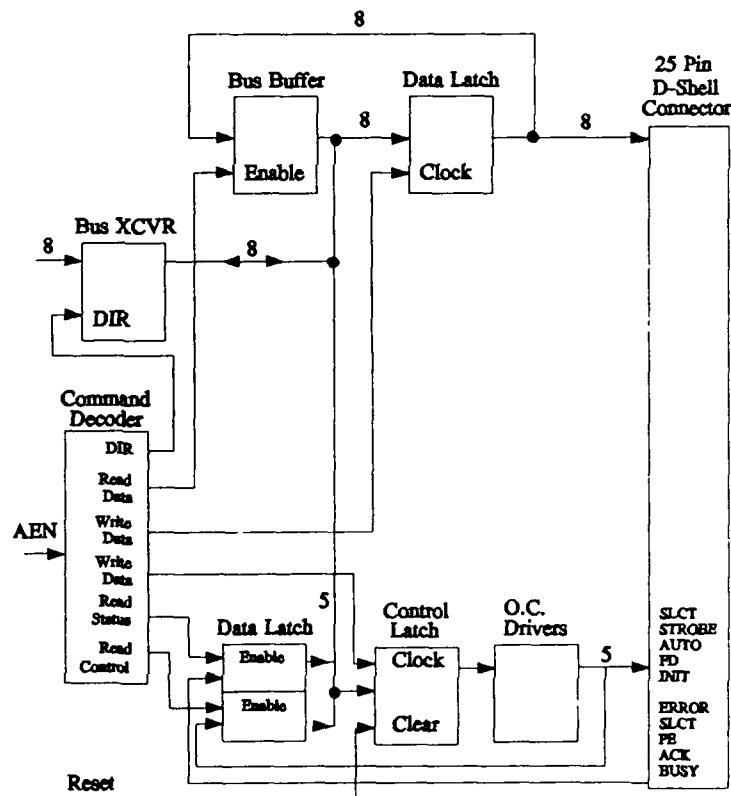


Figure 1 IBM parallel port adapter block diagram

2.1 Parallel Port Connector:

The parallel port connector is an external female 25 pin d-shell connector (DB-25F). The signals are assigned to the pins as shown in Figure 2. The length of the attaching cable can be up to 25 feet in length using proper matching techniques and shielded printer cable. Ribbon cable may be used for shorter runs. The 8 ground lines provide independent signal returns for each of the data lines, and best performance is obtained when these are implemented as twisted pairs.

2.2 Port Access: Access to the parallel port is provided through five I/O instructions which operate on 3 consecutive I/O addresses. The address of the each parallel port is stored by the BIOS as one word starting at 0040:0008. If no adapter is installed for a port the address is zero. In most cases, port one (LPT1) is located at 378h. The first of the three addresses is the data register, the second is a read-only status register, and the third is control register. The parallel port signals, their pin assignments and address locations are summarized in Table 1.

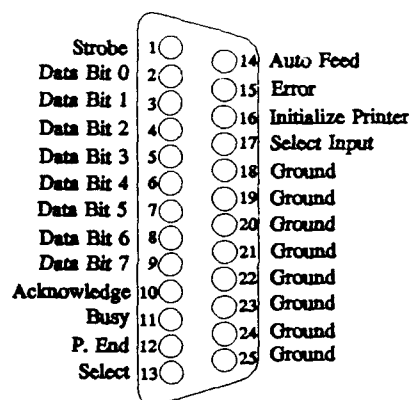


Figure 2 IBM parallel port adapter DB-25F connector pin assignments

Data Register	7	6	5	4	3	2	1	0
Base Address	Data 7	Data 6	Data 5	Data 4	Data 3	Data 2	Data 1	Data 0

2.2.1 The Data Register (378h): The data register is a read/write access port. Data can be moved into and read from the register using the standard IN/OUT instructions. It is an eight bit register which under standard operating conventions contains the data last written to the output latch. Input from the data register will return the values stored in the output latch; or if an external device is driving these lines, the data will be OR'ed with the current contents of the latch. Data input can therefore be accomplished by writing zeros to the output latch with the OUT instruction and then reading the data driven on these lines by the external device. Some caution should be given at this point due to the fact that the output of the data register uses the totem pole output structure. Since the totem pole output is active in both logical states, directly driving these gates with another totem pole output device may yield unpredictable results or eventually device damage. Modern port adapters place current limiting resistors between the output latch and connector to mitigate this problem, but older ports may not include them, therefore some method of self testing should be provided. In any case, it is a good practice to keep the register drive time to a minimum through control and handshaking.

DTIC QUALITY INSPECTED 7

Accession For	
NTIS CRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

Table 1 Parallel Port Signals

Signal Pin No.	Return Pin No.	Signal Name	Direction	Address	Bit	Description
1		STROBE	RD/WR	37A	0	The STROBE output is used to signal the receiving device that data is ready in the output latch. The Pulse width must be greater than 0.5 μ s at the receiving terminal.
2	18	DATA 1	*RD/WR	378	0	The DATA signals are Active Low signals which represent the data present in the output latch. DATA 1 is the LSB and DATA 8 is the MSB returned in Software * According to usage conventions, the data port is an out only port with loop back capability for diagnostics only; however, signals driven on these lines will be ORed with the contents of the output latch on input
3	19	DATA 2	*RD/WR	378	1	
4	20	DATA 3	*RD/WR	378	2	
5	21	DATA 4	*In/Out	378	3	
6	22	DATA 5	*RD/WR	378	4	
7	23	DATA 6	*RD/WR	378	5	
8	24	DATA 7	*RD/WR	378	6	
9	25	DATA 8	*RD/WR	378	7	
10		ACK	RD	379	6	Acknowledge (ACK) is an active low pulse approximately 5 μ s in length indicating that the external device is ready to receive data.
11		BUSY	RD	379	7	BUSY is an active high signal which indicates that the external device is busy and cannot receive or transmit data.
12		PE	RD	379	5	When connected to a printer, P. End is an active high signal used to indicate that the printer is out of paper.
13		SLCT	RD	379	4	When connected to a printer, select (SLCT) indicates the printer is in the selected state.
14		AUTO FEED	RD/WR	37A	1	When connected to a printer, auto feed (AF) is an active low signal that paper is automatically fed into the printer after printing.
15		Error	RD/WR	379	3	
16		Initialize	RD/WR	37A	2	When connected to a printer, this is an active low signal that resets the printer controller to its initial state.
17		SELECT	RD/WR	37A	3	SELECT
18-25		Ground				

Status Register

	7	6	5	4	3	2	1	0
Base Address+1	Busy	Acknowledge	Out Of Paper	Select	Error	Not Used		

2.2.2 The Status Register (379h): The status register is a read-only access port. It is an 8 bit register of which the 3 LSB bits are unused. When connected to a printer, the status register provides real-time status to the processor about the state of the printer. For instrumentation purposes, the status register will be used to return the status of the instruments such as conversion in process, device busy, etc.. The status register utilizes open collector output coupling and may be driven directly with TTL signals.

Control Register

	7	6	5	4	3	2	1	0
Base Address+2	Not Used			IRQ Enable	<u>SELECT</u>	RESET	<u>AUTO FEED</u>	<u>STROBE</u>

2.2.3 The Control Register (37Ah): The control register is a read/write access port. It is an 8 bit register, but the 3 MSB bits are unused. Data can be moved into and read from the register using the standard IN/OUT commands. When connected to a printer, this register provides control signals to the printer ie. STROBE signals the printer to read the data from the data port and Reset causes the printer to clear the print buffer and return to its initial state. Similar functions can be used for the control of instrumentation such as using the strobe line to start a measurement process and Reset to return the device to its initial state.

3.0 DESCRIPTION OF OPERATION

This purpose of this instrumentation design is to develop a minimal concept for a highly portable, flexible analog data acquisition system that can be used to monitor a single process stream at a variety of locations where PC's already reside. It is assumed that physical domain conversion and scaling are completed by an external device, and that the data acquisition system will be presented with a continuous signal ranging from 0 to 5 volts. The block diagram of this implementation is shown in Figure 3. This configuration was chosen to provide the greatest conversion throughput with the smallest amount of hardware. There are four logical blocks to this instrument. The majority of the control functions are performed under Software Control in the PC and translated to hardware signals by the parallel port adapter. The Parallel Port performs all I/O. The ADC performs data conversion under direction of the PC and the TTL Buffer is used to latch the data from the ADC and present it in TTL format to the parallel port.

3.1 Operation of Parallel Port ADC: As shown in Figure 4, the heart of the Parallel port ADC is the Texas Instruments ADC0820C high speed analog-to-digital converter. (For exhaustive technical details see the Texas Instruments Data Book)

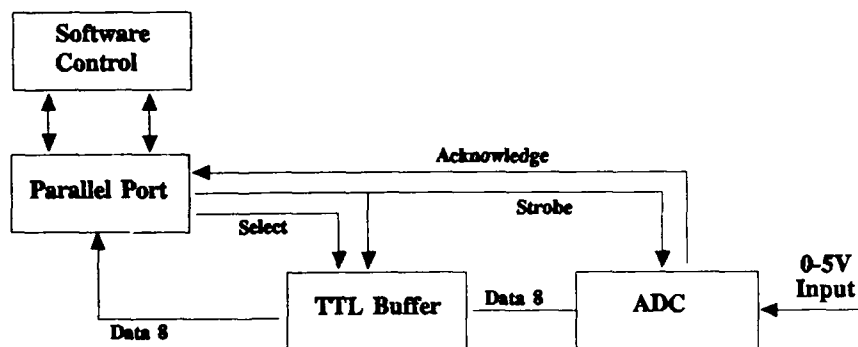
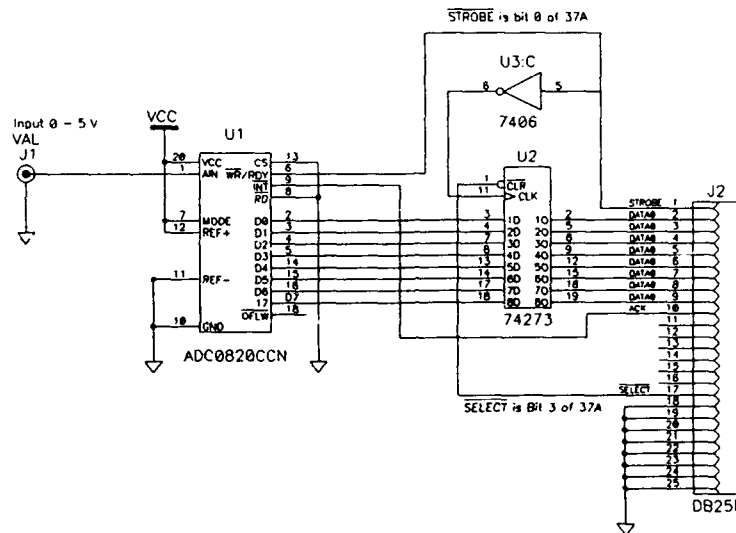


Figure 3 Parallel Port ADC Block Diagram

The ADC0820C was chosen for its high conversion speed. It's maximum conversion time is 2.5 μ s, and the on-chip track and hold circuits have a 100 ns sampling window which allows conversion of continuous signals with slew rates of up to 100 mV/ μ s. This allows continuous sampling and conversion at the throughput rate of the parallel port without requiring additional sampling components.

The ADC0820C voltage references are fixed at the power supply rails of 0 and 5 volts for this minimal implementation. A large bypass capacitor is used mitigate noise on the reference inputs. The signal input is applied directly to the converter input. Conversion begins when the



SIGNAL NAME	Mode High and CS and RD Low TIMING DIAGRAM	PIN	PORT	BIT	PORT NAME	DESCRIPTION
STROBE			37A	0	STROBE	Strobe Bit written in Software
WR/RDY		1	37A	0	STROBE	Strobe Output from PC
CLK					STROBE	ACK Input to PC
INT		10	379	6	ACK	ACK Input to PC
ACK		10	379	6 6	ACK	ACK Bit Read in Software
D0-D7						ADC Data Port
D0-D7		2-8	378	0-7	DATA	Output Latch Data Loaded Latch read with INP
SELECT			37A	3	SELECT	Select Input Written in Software
CLR		17	37A	3	SELECT	Hardware Output for SELECT

Figure 4 Parallel Port Analog to Digital Converter Schematic Diagram

parallel port STROBE line is taken low through software. STROBE is applied directly to the WR/RDY input of the ADC and is also inverted by 74LS06 TTL inverter and applied to the CLK input of a 74LS273 8-bit data latch. On the falling edge of WR/RDY, the ADC0820C begins the conversion process and the internal counter output INT goes to zero. When the ADC conversion is complete, INT will again return high and the ADC0820C data lines will change

from a high impedance state to the valid data state. $\overline{A_NT}$ is also presented to the parallel port adapter $\overline{ACK}$ line which signals that the data conversion process is complete, and the data on the ADC0820C is valid. When the $\overline{ACK}$ signal is detected, the software takes $\overline{STROBE}$ high again which loads the data into the data latch and returns the ADC0820C data outputs to the high impedance state. The control software immediately reads the data through the parallel port data register, and then clears the 74LS273 with a logic low pulse on the $\overline{SELECT}$ line (bit 3 of 37A) which returns the system to the ready state for the next conversion cycle.

3.2 Advanced Capabilities: Although this design focused on a minimal implementation, the number of unused control and status register lines suggest the parallel port is capable of supporting a wide variety of more advanced functions. The number of variations is nearly infinite and is application dependent, so this section will simply suggest an approach for a few of the most common capabilities without giving specific implementations.

3.2.1 External Triggering: External triggering is a common requirement, and it is easily implemented on this system without significant hardware changes. The trigger signal is applied to one of the unused lines of the status or control registers and is monitored in software. When a signal is detected on the selected line, the data acquisition cycle is initiated. Since control is maintained in software, the detected trigger can be used to initiate a single data point or a stream of points after the trigger.

3.2.2 Variable Voltage References: The ADC0820C does provide for variable reference voltages within the range of the power supply rails. The addition of voltage dividers at the reference voltage inputs will allow adjustment of the input range anywhere within the power supply range. This should only require modest hardware modifications, but care must be given to insure the stability of the reference levels.

3.2.3 Analog Signal Multiplexing: Analog signal multiplexing is required in some cases where multiple inputs are required. In this implementation, there are 4 unused read/write outputs which could be used for multiplexing control. Multiplexing can be easily achieved by simply using some of these lines to control one of the industry standard analog signal multiplexers. Using 3 control lines would allow the selection of 8 analog inputs lines.

3.2.4 Addressing: Addressing is used to select/activate entire devices. Three or four bit addressing can be accomplished just as it was in the analog signal multiplexing except that in this case an entire device would be enabled rather than just a single signal line. If more address space is required, the a control line could be used as an address latch enable (AEN). When AEN is active, the data register could be used to transfer the address of the desired device to the external device address register. When AEN is inactive, the commands would operate as normal on the address selected by the address register.

4. SOFTWARE DESCRIPTION

The control software for this minimal demonstration was written in Turbo C and is given in Table 2. It is meant only as a minimal implementation example - a starting point for more specific applications. As shown in the flow diagram of Figure 4, the system is configured for use with parallel port 1 (378h). STROBE and SELECT are initialized and the data register is cleared. The ADC's data latch is cleared with a pulse to SELECT, and the acquisition loop begins. Within the loop, the conversion cycle is initiated by setting STROBE to logic LO. The ACK line is then monitored in a loop until ACK is taken HI indicating the conversion is complete. When the data is ready, the STROBE line is again taken HI which loads the data from the ADC0820C in to the parallel port ADC output latch. This also readies the ADC0820C for the next conversion. The data in the latch is then read from the parallel port adapter data register and the output latch is immediately cleared with a strobe pulse on SELECT in order to keep the drive time to a minimum. The collected byte of data is stored in the array *X*. If no key has been pressed the cycle repeats; otherwise, the collected data is printed and the program terminates.

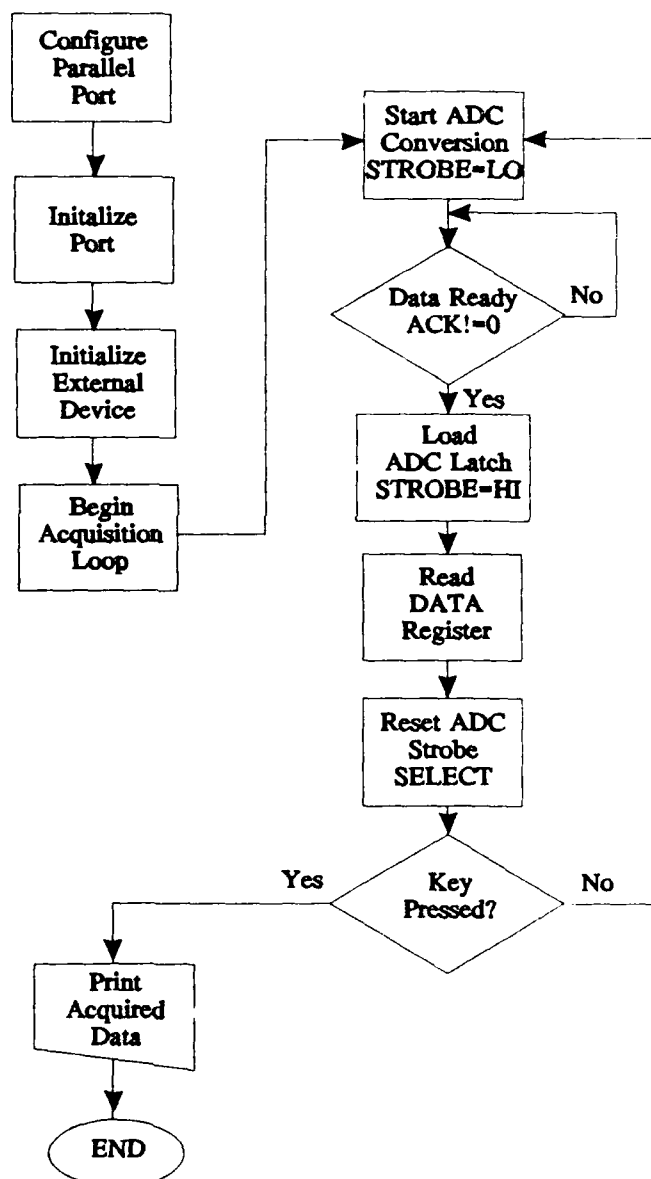


Figure 5 Parallel Port ADC Flowchart

TABLE 2 Parallel Port Analog to Digital Converter Minimal Control Program

```

#include <conio.h>
#include <stdio.h>
#include <dos.h>
#include <time.h>

/**** PARALLEL PORT ANALOG TO
DIGITAL CONVERTER MINIMAL
CONTROL PROGRAM */

int main(void){
    unsigned char x[30000]=""; /* Data space for 30 kbytes of data */

    /**** CONFIGURE FOR PARALLEL PORT 1 */
    int port = 0x378, /* Data Port */
        port1 = 0x379, /* Real Time Status Port */
        port2 = 0x37a, /* Control Port */
        strobe, /* Port2 Storage */
        busy, /* Port1 Storage */
        ack=0; /* Port 37A, bit 3 */
    long int timeout=100; /* Port read Interval */
    int i=0,k=0;

    /**** INITIALIZATION -- Get initial port settings and set initial state: STROBE BIT - HI
    DATA BITS - LO */
    strobe = inp(port2); /* Get initial value for Port1 */
    busy = inp(port1); /* Get initial value for Port2 */
    x[0] = outp(port,0x00); /* Set Data bits to 0 */
    strobe=outp(port2,strobe|0x01); /* Turn Strobe On -- WR/RDY LO */

    /**** CLEAR THE OUTPUT LATCH with strobe to SELECT */
    strobe=outp(port2,strobe|0x08); /* Turn SELECT Off to Clear 74LS273 */
    strobe=outp(port2,strobe&0xf7); /* Turn SELECT In on to Ready 74LS273 */

    /**** BEGIN ACQUISITION LOOP */
    while(!kbhit()){
        i++;
        if(i>30000)i=0;

    /**** START DATA CONVERSION in the ADC0820. Note the conversion starts on the rising edge of
    the logic low pulse WR/RDY */
    strobe=outp(port2,strobe&0xfe); /* Turn Strobe off -- WR/RDY HI */
    timeout=10000;
    ack = inp(port1)&0x40; /* Read Ack bit of Port 379h */
    while((ack!=0)&&(timeout>0)) /**** Wait until ACK signals that
    the data is ready (ACK!=0) */
        {timeout--; /* Read Ack bit of Port 379 */
        ack = inp(port1)&0x40;}
    strobe=outp(port2,strobe|0x01); /* Turn Strobe On -- WR/RDY LO */
    if(timeout==0)
        printf("*** Timeout Error **\n");
    x[i] = inp(port); /**** Read Data PORT 378h */
    /**** Clear the output latch with strobe to SELECT */
    strobe=outp(port2,strobe|0x08); /* Turn SELECT Off to Clear 273 */
    strobe=outp(port2,strobe&0xf7); /* Turn SELECT on to Ready 273 */
    }

    /**** OUTPUT the first 10000 samples collected in the run */
    for(i=0;i<100;i++) printf("x[%6d] = 0x%X\n",i, x[i]);
    return 0;
}

```

5. SUMMARY

The parallel port analog to digital converter prototype was constructed for test and evaluation as shown in Figure 6. Snap on connectors were used to provide easy connection to breadboarded components. The prototype was tested and showed stable conversions at 96,000 samples per second.

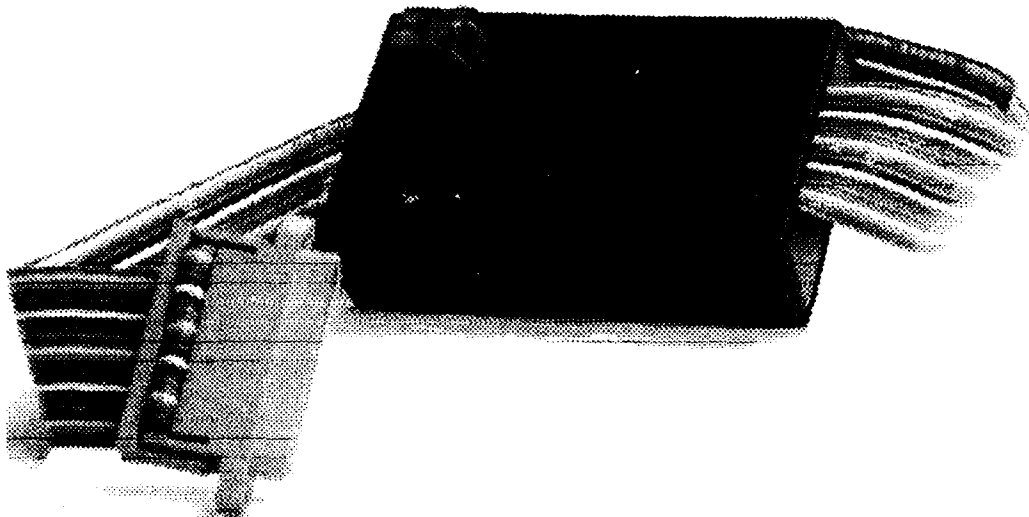


Figure 7 Parallel Port Analog to Digital Converter Prototype

The prototype demonstrates the use of the standard Centronics parallel port for instrumentation purposes through the implementation of a high speed analog to digital converter for data acquisition and process monitoring. The parallel port adapter is used because it provides simple, external connection to the PC, it provides moderate speed byte parallel data transfer rates, and it is standard equipment on virtually every PC. The design is extremely simple requiring little fabrication and minimal costs. The parallel port ADC requires only three integrated circuit (IC) packages with a total component cost much less than twenty dollars. The small size of the converter and the single, external DB-25 connection make it an extremely convenient and portable data acquisition system. The design is flexible and techniques to tailor the hardware for specific applications have been discussed. A simple Turbo C software driver has also been presented.