

MANKATO STATE UNIVERSITY  
COLLEGE OF PHYSICS, ENGINEERING,  
AND TECHNOLOGY

MECHANICAL ENGINEERING DEPARTMENT

ME-446, SENIOR LAB

DIESEL ENGINE ANALYSIS

Instructor : Dr. Jerzy Fiszdon

Reported By :

██████████  
██████████  
██████████

Date : June 6, 1994

## Problems:

- poor organization of group & distribution of tasks
- poor computer skills
- poor communication

## ABSTRACT

There were three main objectives to this lab, and they are as follows.

One, to build a permanent box which would contain all necessary electrical equipment, including the charge amplifier, the circuit board, the 15 volt power supply, an extension cord, and wires attached to a female connect.

*This was not an objective*

Two, to write a program in BASICA which would collect a data file and calculate the net work output of the engine. This program was to utilize the DAS8 data acquisition system, which would read in voltages from the pressure transducer and the corresponding voltage for the crank angle. This would give the relationship of cylinder pressure to cylinder volume. Three, to compare the power calculated by the computer, to the power calculated using the brake dynamometer.

A permanent box was constructed which greatly simplifies system hookup. Two programs were written which collect data from the engine, and calculate the power output of the engine. However some minor adjustments must still be made in order to transfer the data collected from one program to the other.

## EQUIPMENT USED

### TEST EQUIPMENT

- Thermo King Diesel Engine

Model.....TK353      Number of cylinders.....3  
Bore.....2.83 in.      Stroke.....2.83 in  
Displacement.....56.6 in<sup>3</sup>      Compression Ratio.....23:1  
Oil Filter type.....Napa Gold Oil Filter, 1344  
Oil Type.....10W40 (10 quarts)

-Stuska Absorption Brake Dynamometer

Model.....400      Serial Number.....41589-SL

### DATA ACQUISITION EQUIPMENT :

- DAS8 Data acquisition software

- Metrabyte Screw Terminal Accessory Board

Model.....STA-08PGA  
Revision.....8943      Serial Number.....94921

- PCB Laboratory Charge Amplifier

Model .....462A      Ranges.....50 to 1,000,000 units/volt  
Range tolerance....  $\pm 1\%$

- PCB Hi-Sensitivity Piezoelectric Pressure Transducer

Model.....112A      Range.....0 to 3000 psi

- Novotechnic Crank Angle Sensor

- Computer chip #8254

The equipment setup is found on page 1 appendix A

# PROJECT PROCEDURE

## I. BUILDING THE BOX

Before this project began, there were seven wires hanging out of the crank angle sensor, and all other equipment was scattered throughout the laboratory room. Our goal was to gather all the equipment needed and make a permanent box for it. Also, to make connecting the wires from the crank angle sensor to the data acquisition board one simple connection.

The First half of the project consisted of building the black box, which contained all electrical components. In order to have enough time to run tests on the engine, it was determined that the box should be made out of wood, instead of sheet metal. In order to give it more of a professional look, it was to be coated with a black laminate. This box is to be a permanent fixture, with minimal adjustments needed to be made inside of the box while in use. The construction consisted of two major parts. One, the building of the box itself; and two, making all of the appropriate wire connections. While one person worked on the electrical connections, the other two worked on the black box.

First, the dimensions of the box were determined. The charge amplifier may need to be adjusted while in use, so a section of the wall was cut out, thus leaving the front and back of the amplifier exposed outside of the box. In order to do this the width of the box

must be  $9 \frac{9}{16}$  in. In order to have room for the circuit board, power source, cords, and a little extra space, a length of  $13 \frac{11}{16}$  inches is necessary. The top of the box should be about an inch above the top of the charge amplifier to insure enough room for the coaxial cable and the computer cord. This gives a height of the box of 6 inches. The final dimensions of the box should be  $13 \frac{11}{16} \times 9 \frac{9}{16} \times 6$ .

The box was then cut out and screwed together. The dimensions of the lamination were then taken, and this was also cut out. During the cutting, some chipping on the edges occurred. This laminate was then glued to the wood.

It was desirable to have a box which could simply be plugged into from the outside. In order to achieve this, the wires coming out of the crank angle sensor were soldered to one male connect. Next a female connect needed to be soldered which would accept the male side, and distribute each voltage to its appropriate connection within the black box. Next a cord was obtained which would take in A/C power from the wall, and transfer it to the power supply. This was also soldered to the appropriate connections. Also the DC side of the power supply needed to be soldered to extension wires which could reach the circuit board. Each connection was done with caution, and covered with shrink wrap to insure that there will be no short circuits in the future.

Once the box and wiring were completed, each piece of equipment inside of the box was fastened down. An extension cord was obtained which would merge the power supply and the charge amplifier into only one power cord coming out of the box.

## II. WRITING DAS8 PROGRAM FOR DATA ACQUISITION

Before writing a program in DAS8, we needed to become familiar with this software. The first step was to read the highlights of the DAS8 manual which was given to us. This gave us some idea of how the programming was to be done. There were several example programs already installed on the hard drive, which displayed how to perform several programming tasks. These tasks consisted of loading the subroutine DAS8.BIN, setting the counter rate, making A/D conversions, and saving them onto a data file. Although the algorithm in these programs looked correct, none of these programs worked and the computer would just hang. This became a very time consuming problem. ?

The program we were most concerned with was MODE5.BAS. This was to be the program which would read the voltages coming from the crank angle sensor and the pressure transducer. We first tried debugging this program by finding out at what location the program was hanging. After determining the line number that was giving us problems, we still did not know how to resolve the problem. We then proceeded by making several calls to Metrabyte Corp. to see if

they could help, but even they could not determine what the problem was. We then asked an electrical engineering student, Kurt, who had some experience in data acquisition if he could help. After looking at the program and the computer several times, he was able to find the problem. The computer had a burnt out computer chip #8254. When the program was run, it was waiting for the counter to begin, but this chip was broken. We were able to bypass this problem by setting up a function generator which would replicate the function of the internal counter. There was also a problem with the BLOAD command, and Kurt was able to fix this as well. The program then seemed to run properly, but a total of twelve weeks was spent doing so. A few minor adjustments needed to be made from the MODE5.BAS. For example, that program only reads one channel, but our program needs to read two channels. We did all of the appropriate changes, and resaved the program as ENGINE.BAS. This DAS8 program is shown on pages 1-4 of Appendix B.

### III. WRITING A BASIC PROGRAM FOR POWER CALCULATIONS

While we were trying to solve the problems with the Metrabyte software, we wrote a program in basic which would perform the required calculations. This program is shown on pages 5-6 of Appendix B. The program may appear a bit confusing but, several REM statements were entered to clarify what the purpose of each portion of the program is. A word of caution: this is a very old version of basic, and some very familiar programming techniques



used in today's basic is not valid in this program. For instance, it is not possible to write an IF-THEN block, it must all be in one statement. It took us several hours to figure out how to write the IF-THEN portion of the program, since we did not know this at first. The major steps of the algorithm are as follows.

- 1) open the data file given by DAS8
- 2) read in all the data collected for the crank angle and the pressure
- 3) separate the combined array from DAS8 to two separate arrays of crank angle voltage and pressure voltage
- 4) search for the 1st voltage representing bottom dead center in the crank angle array (Low voltage which starts the ramp)
- 5) search for the 2nd voltage representing bottom dead center in the crank angle array (Start of the next voltage ramp)
- 6) search for the 3rd voltage representing bottom dead center in the crank angle array (Start of the next voltage ramp)
- 7) analyze data from the first bottom dead center reading, to the third bottom dead center reading, in both crank angle and pressure arrays. This will be two revolutions, and one complete cycle.
- 8) Using the trapezoidal rule, calculate the power output of one cycle for one cylinder. Then display the total horsepower output.

#### IV. ENGINE ANALYSIS AND TESTING

Once both programs were operating correctly, an attempt was made to run the program completely, but the output of the DAS8 program was not giving correct readings. After several weeks working on this problem, we asked the Trafton Electrician, John Caven, to look at our wiring. After looking at it he was able to find a connection that was not made properly. This electrical system was wired according to the 1993 senior lab report, but their report was incorrect and that was the problem in this case.

After this was solved, we ran the program again and all the data seemed to be correct. The crank angle voltages were reading from zero to five volts in a ramp like manner, just as they were supposed to. Also the voltage readings alternated from pressure to crank angle, just as they were supposed to. The next problem was that the data file being saved by the DAS8 program, was unreadable to the Basic program. We were unable to look into this problem any further, since the year was coming to an end.

## PROGRAM PROCEDURE

- 1) Before turning computer on, be sure that all necessary components are plugged in. These include:
  - Function generator
  - Power supply and charge amplifier (White extension cord from black box)
  - Computer
  
- 2) Commands to be entered in order to get into the DAS8 program
  - CD..
  - CD DAS8
  - BASICA
  - LOAD "ENGINE
  
- 3) Running the DAS8 program
  - Before running, be sure the function generator is set at a step function from 0 to +5 volts.
  - Enter desired sample rate as prompted (Note : until the counter chip is installed, the sample rate entered must be the same as the frequency setting on the function generator. i.e. 400 samples = 400 Hz.)
  - Enter length of time for data collection. (At least three seconds should be sufficient
  
- 4) Running the Basic program
  - Load "Diesel and Run
  - Observe results

Note : Since we ran out of time and were unable to write a statement which plots the data read by the basic program, we inserted a statement which enables you to print the data file and use it in a spreadsheet.

## EQUATIONS USED

The volume with respect to the crank angle position was determined using the following equation:

$$V = V_c + (V_t - [(\pi * B^2 / 4) * (l + a + s)])$$

where  $V_c$  = clearance volume = .8136 in<sup>3</sup>  
 $V_t$  = total cylinder volume = 18.615 in<sup>3</sup>  
 $B$  = Cylinder bore = 2.83 in  
 $l$  = connecting rod length = 4.4095 in  
 $a$  = crank radius = 1.145 in.  
 $s = a * \cos\theta + (l - a^2 * \sin^2\theta)^{.5}$

A figure displaying all variable is shown on page 3 of appendix A.

The crank angle sensor (CAS) gives an output ramp voltage from 0 to 5 volts in one revolution. 0 and 5 volts represents bottom dead center (BDC) and 2.5 volts represents top dead center (TDC). This gives the relationship of .01389 volts per degree, or .7958 volts per radian. This gives the following:

$$\theta = \text{CAS voltage} * 1.25664 \text{ radians/volt}$$

In order to determine the area under the curve, the trapezoidal rule was used, which gives:

$$A = [(V_2 - V_1) / 2] * [p_1 + p_2]$$

$V_1, V_2$  = volumes

$P_1, p_2$  = pressures

There were four areas evaluated, each of which represented a stroke of the engine:

$A_c$  = area under compression curve (lb in)

$A_e$  = area under the expansion curve (lb in)

$A_{ex}$  = area under exhaust curve (lb in)

$A_{in}$  = area under the intake curve (lb in)

These areas were calculated in the basic program, then used to determine the power output of the engine:

Power (compression) =  $P_c = (A_c * (1/12 \text{ ft/in}) * (\text{rpm} * .5)) / 66,000$

Power (expansion) =  $P_e = (A_e * (1/12 \text{ ft/in}) * (\text{rpm} * .5)) / 66,000$

Power (exhaust) =  $P_{ex} = (A_{ex} * (1/12 \text{ ft/in}) * (\text{rpm} * .5)) / 66,000$

Power (intake) =  $P_{in} = (A_{in} * (1/12 \text{ ft/in}) * (\text{rpm} * .5)) / 66,000$

where rpm = engine speed

The 66,000 represents  $33,000 \text{ ft} * \text{lb} / \text{min} * 2$ , since the trapezoidal rule employed the areas under three data points instead of two. This was done to reduce the error in the calculation. The use of rule is shown on page 4 of appendix A.

The total power is then:

$P_{tot} = P_e + P_{in} - P_c - P_{ex}$

This is the power produced by the combustion minus the power required to compress the air and the power loss due to friction in the cylinder bearings.

## RECOMMENDATIONS

- 1) The next step that should be done is to get the data file from the DAS8 program, readable to the Basic program.
- 2) Then the program could be modified to plot the P-V diagram for the diesel engine.
- 3) Checking to see if the program runs on Q-basic may result in easier programing in the future.
- 3) The exhaust leaks fumes into the room. This should be fixed.
- 4) Check to see that the 8254 chip was ordered, and install it.
- 5) The throttle needs to be modified to enable better speed control
- 6) Determine way to elevate the diesel fuel container above the engine in order to eliminate stalling.

## REFERENCES

- 1) Akhtar, Gohman, Krueger, Leibel : *Senior Mechanical Engineering Lab Report*, 1992
- 2) Andruskiewicz, Hanauer, Hemish, Kipfer : *Senior Mechanical Engineering Lab Report*, 1993
- 3) Thermo King Corp., *Thermo King Overhaul Manual*
- 4) Metrabyte, *Metrabyte DAS8 User's Manual*, 1988  
Phone # 1-800-348-0033 / John
- 5) Keithley/ Metrabyte, *Data Acquisition Catalog & Reference Guide*, 1993, vol 26

**APPENDIX A : DIAGRAMS AND CALCULATIONS**



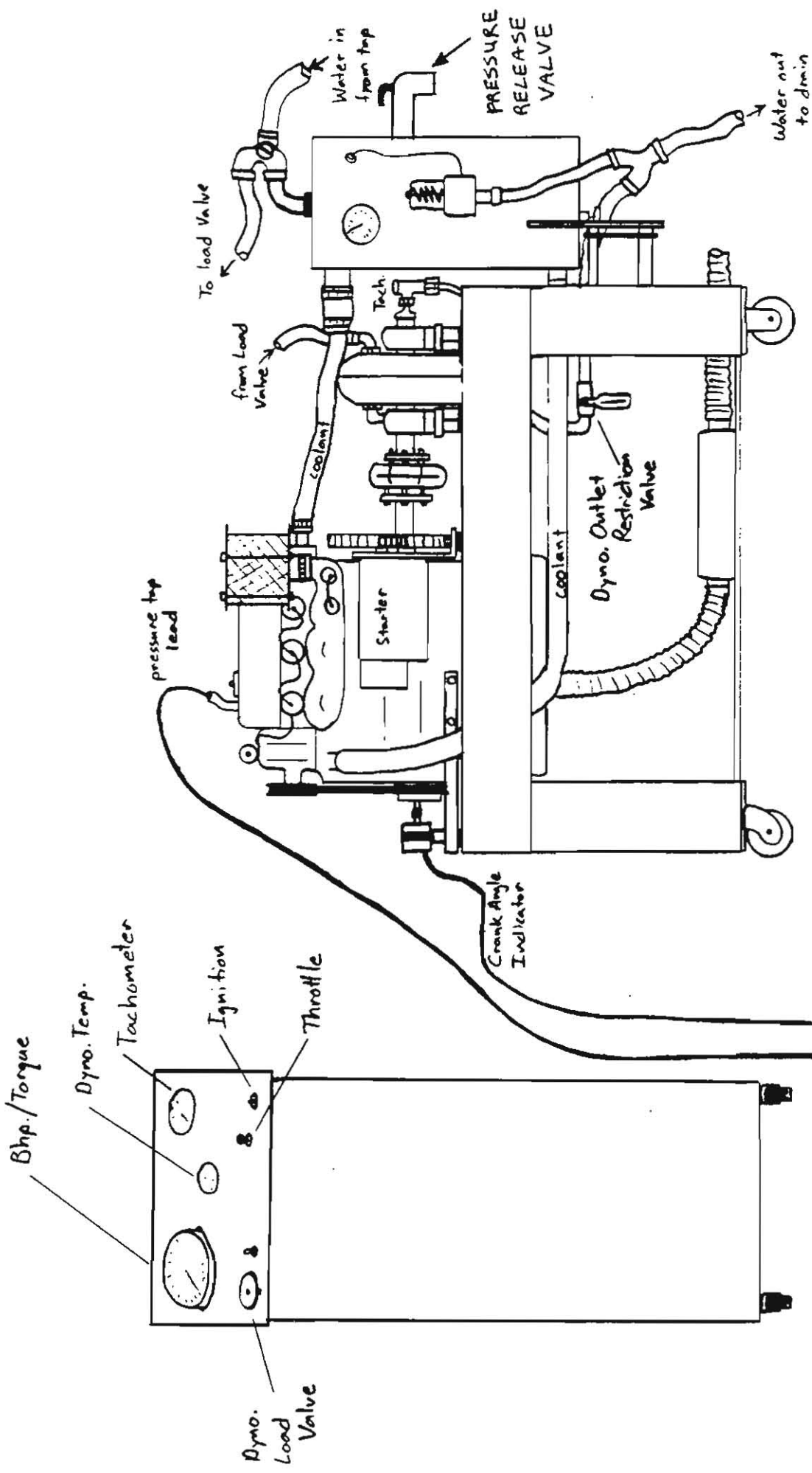
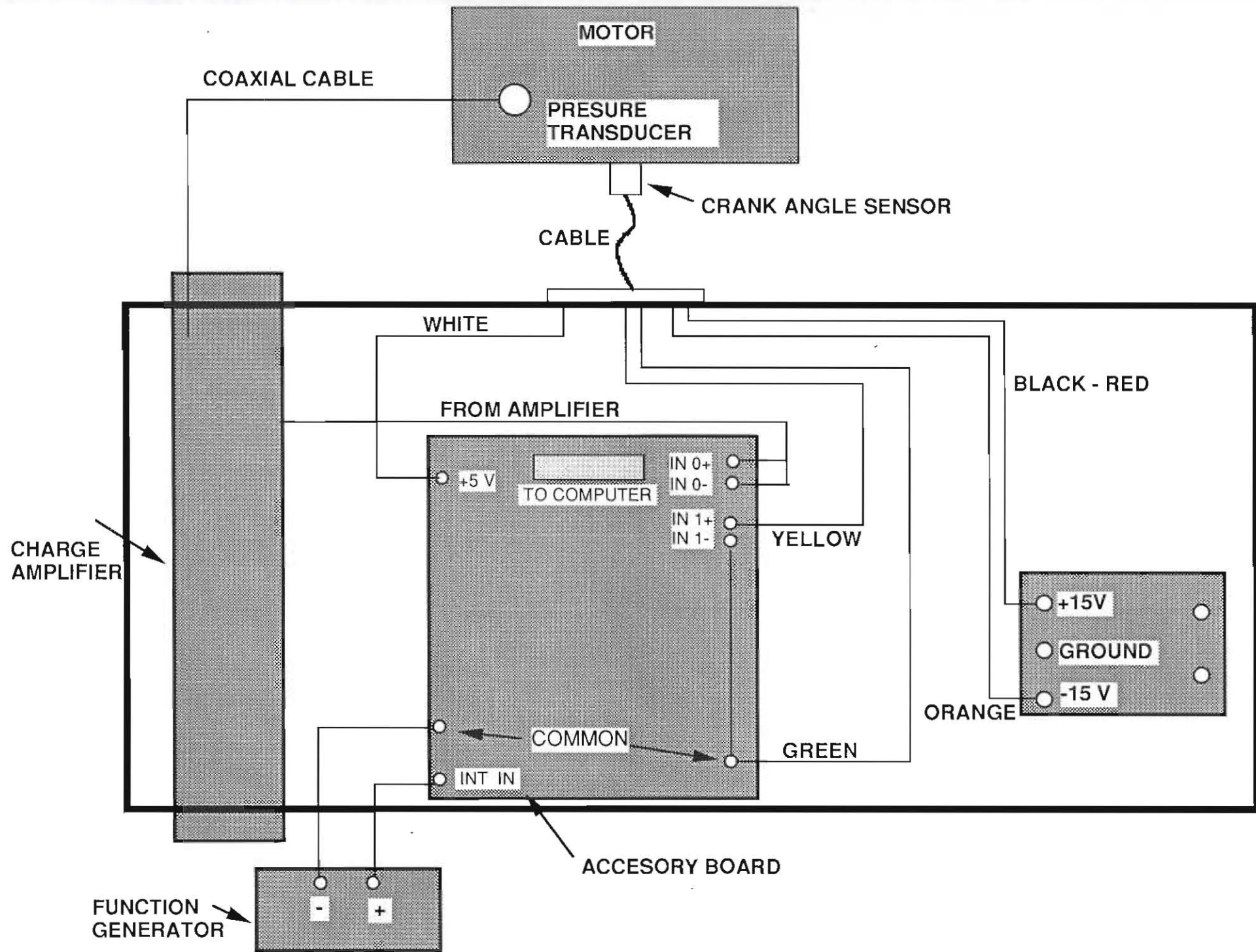
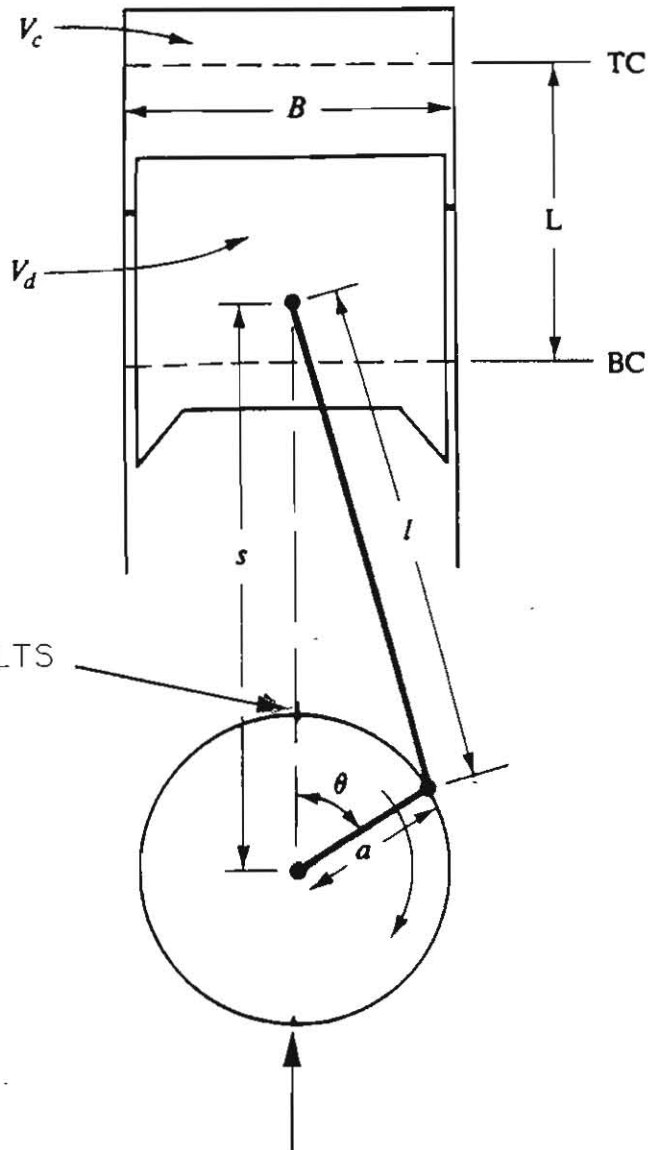


Fig. 1

DATA ACQUISITION SYSTEM  
SEE APPENDIX A Fig. 2





TOP DEAD CENTER (TDC) = 2.5 VOLTS

BOTTOM DEAD CENTER (BDC) = PEAK AT 5 VOLTS AT END OF REVOLUTION, THEN STARTS BACK AT 0 VOLTS FOR NEW REVOLUTION

VOLUME VARIATION WITH CRANK ANGLE

$$V = V_c + [ V_t - [ (\pi * B^2 / 4) * (1 + a - s) ] ]$$

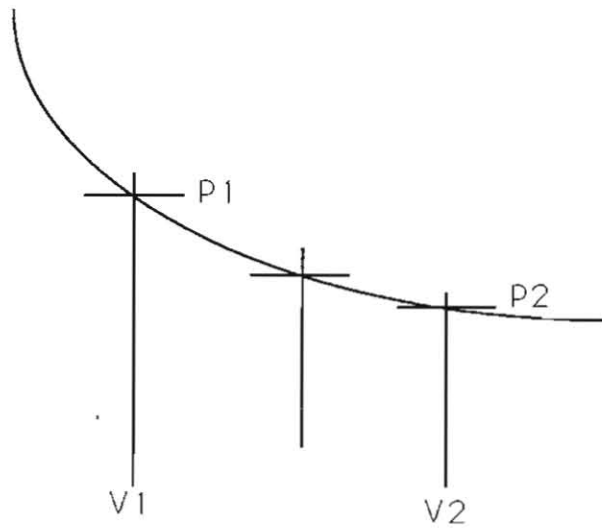
$$S = a * \cos\theta + [ l^2 - a^2 * (\sin^2\theta)^{.5} ]$$

$V_c$  = CLEARANCE VOLUME

$V_t$  = TOTAL VOLUME =  $V_c + V_d$

$V_d$  = DISPLACEMENT VOLUME

DIAGRAM AND FORMULAS ILLUSTRATING CALCULATION  
OF POWER BY USE OF THE TRAPEZOIDAL RULE



$$A_c = ( (V_1 - V_2) / 2 ) * (P_1 + P_2)$$

$$\text{POWER} = A_c * ( 1/12 \text{ ft/in} ) * ( \text{RPM} * .5 ) / ( 33000 \text{ lb.ft/min} * 2 )$$

↑  
2 revolutions  
per complete cycle

↑  
twice the area  
examined under curve

**Special Features:**

- Value-for-money Absolute Angle encoding module
- Gap-free output over all 360°
- High absolute linearity
- Interchangeable without adjustment (no zero offset and constant voltage gradient)
- Guaranteed 12 bit resolution and repeatability
- Two grades for 10 or 11 bit absolute accuracy
- Long life -  $50 \times 10^6$  revolutions
- High operating speeds 6000 r.p.m. for short periods  
1500 r.p.m. continuous use
- Rugged industrial grade construction, sealed shaft, connector to IP 65
- Maximum reliability through the combination of the Dinopot HQ5 system\* and hybrid technology
- DBP 27339-9, U.S. Pat 4,203,074 and other patents



This potentiometer has been designed to be used as an absolute 360° angle encoder.

When used in conjunction with an external A/D converter, the unit operates as an angle transducer with 12 bit resolution and offering 10 or 11 bit absolute accuracy.

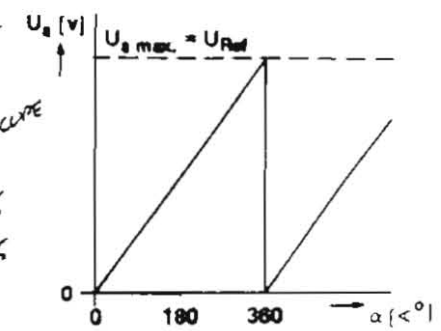
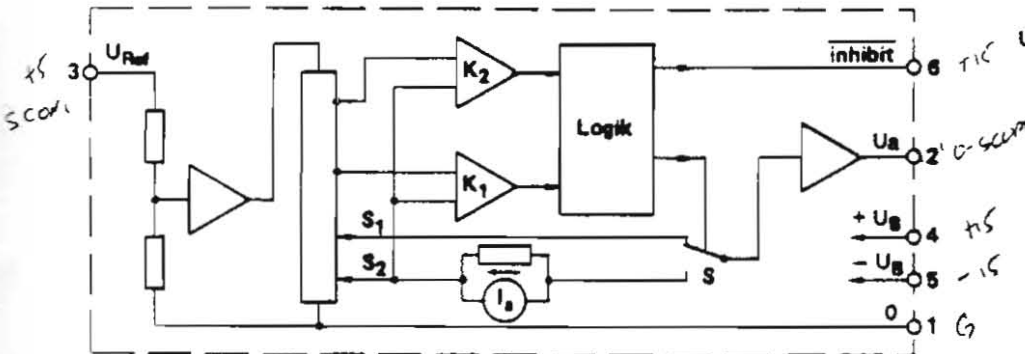
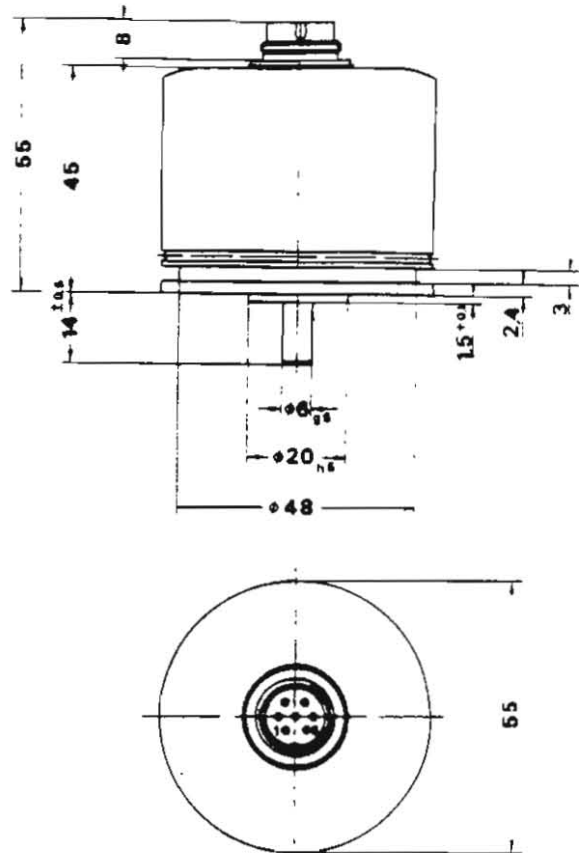
The transducer requires only 7 connections and these are made via a 7 pole, waterproof plug and socket.

The A/D converter can be mounted in an external rack, thereby, offering ease of connection with protection from noise.

We can provide examples of circuits where the unit is used in connection with A/D converters or analogue systems. If required and in order to allow the rapid construction of prototype systems we can supply circuit modules on standard Europa cards, the modules available would include A/D converters, Binary to B.C.D. converters and Digital Displays. Contrary to the limitation encountered with other potentiometers, the AW 360 ZE can be used over all 360° of rotation since it has no dead band. This limitation has been overcome by the use of two wipers and a special Hybrid circuit.

The reset voltage at the beginning and end of the sawtooth are well defined and there is no function angle tolerance as is normally associated with potentiometers. These features coupled with the fact that the maximum output voltage is determined by the applied voltage means that units may be interchanged or replaced without the need to adjust or trim. The transducer delivers a logic signal quite separately from the analogue output which can be used to inhibit the A/D conversion during the reset period.

- |                  |            |
|------------------|------------|
| 1.) Ground - red | 5.) Green  |
| 2.) brown        | 6.) Grey   |
| 3.) white        | 7.) Orange |
| 4.) yellow       |            |



## **APPENDIX B : PROGRAMS**

```

10 '*****
20 '*
30 '*   Demonstration program for DAS-8 used in array mode 5
40 '*   MetraByte Corporation
50 '*****
60 '
70 SCREEN 0,0,0 : KEY OFF : WIDTH 80
250 '
260 '--- Step 1: Initialize DAS-8 with mode 0 -----
265 CLS
270 DIM D%(6)
280 DEF SEG = &H5000
290 BLOAD "DAS8.BIN",0 'Load driver routine
300 OPEN "DAS8.ADR" FOR INPUT AS #1
310 INPUT #1, BASADR%
320 CLOSE #1
330 DAS8 = 0      'Load offset = 0
340 MD% = 0      'initialize mode
350 FLAG% = 0    'declare error variable
360 CALL DAS8(MD%, BASADR%, FLAG%)
370 IF FLAG% <> 0 THEN PRINT "Error in initialization":STOP
380 PRINT "DRIVER LOAD OK"
400 '
410 '--- Step 2: Set timer rate -----
420 MD% = 10     'Mode 10 for setting counter configuration
430 D%(0) = 2    'Operate on counter #2
440 D%(1) = 3    'Configuration #2 = rate generator
450 CALL DAS8 (MD%, D%(0), FLAG%)
460 IF FLAG% <> 0 THEN PRINT "Error in setting counter 2 configuration":STOP
470 '
480 'Prompt user for desired sample rate
490 CLS
500 INPUT "Enter desired sample rate (samples/sec) : ",F
510 ' Output frequency = 2386.4/N KHz
520 N = 2386.4*1000/F
530 IF N < 2 OR N > 65535! THEN PRINT"Warning! A sample rate of ",F," samples/sec is outside
the range of Counter 2":GOTO 500
540 MD% = 11     'Mode 11 to load counter
550 D%(0) = 2    'Operate on counter #2
560 IF N < 32767 THEN D%(1) = N ELSE D%(1) = N - 65536! 'correct for integer
570 CALL DAS8 (MD%, D%(0), FLAG%)
580 IF FLAG% <> 0 THEN PRINT "Error in loading counter 2":STOP
590 'Pin 6 (Counter 2 Out) should now be producing the selected frequency.
600 '
610 'Now fetch duration of scan
620 PRINT:INPUT "Duration of scan (in seconds) : ",DS
630 'Translate duration in a number of conversions for mode 5
640 'Number of conversions = duration x sample rate
650 NC = DS * F
660 '
670 '--- Step 3: Select the channel to scan -----

```

```

680 'Note this program only looks at one channel, but by setting LL% and UL%
690 'to the desired scan limits, the channels will be stored in array A%
700 'in the order scanned, for example if LL%= 1 and UL% = 3 then:-
710 '           A%(0) = channel 1 data
720 '           A%(1) = channel 2 data
730 '           A%(2) = channel 3 data
740 '           A%(3) = channel 1 data etc.
750 'The rest of the program can be modified to handle multiple channel
760 'graphs and data files.
770 MD% = 1           'Set scan limits, mode 1
780 PRINT:INPUT "Enter channel numbers to be scanned (0 - 7): ",LL%, UL% : D%(0)=LL% :
D%(1)=UL%
800 CALL DAS8 (MD%, D%(0), FLAG%)
810 IF FLAG% < 0 THEN PRINT "Error in setting channel scan limits":STOP
820 '
830 '-- Step 4: Dimension data array for A/D data and check sufficient memory -
840 'Check there is enough memory to hold this array
850 IF (FRE(0) - 2000 - NC*2) < 0 THEN PRINT "Warning! There is inadequate memory within
BASIC to hold this data":PRINT"Re-run program":END
860 DIM A%(NC) 'o.k. dimension an array
870 '
880 '--- Step 5: Set mode 5 going and acquire data -----
890 'Note: Counter 2 output (pin 6) should be jumpered to interrupt input
900 '(pin 24).
910 PRINT:COLOR 0,7:PRINT" Press any key to start A/D conversions ";:COLOR 7,0
920 IF INKEY$ = "" GOTO 920
930 MD% = 5           'Mode 5, do conversions direct to array
940 D%(0) = VARPTR(A%(0)) 'Starting location of array
950 D%(1) = NC           'Number of conversions
960 CLS:COLOR 0,7:PRINT" Performing A/D conversions - please wait ";:COLOR 7,0
970 CALL DAS8 (MD%, D%(0), FLAG%)
980 IF FLAG% < 0 THEN PRINT "Error in setting mode 5":STOP
990 CLS
1000 '
1010 '-- Step 6: Choose type of file required -----
1020 CLS : PRINT"Choose from the following:-"
1030 PRINT:PRINT" <1> - Generate a Lotus 1-2-3 compatible import file"
1040 PRINT:PRINT" <2> - Generate a data file & plot using the DAS-8 graphics package"
1050 PRINT:PRINT" <3> - Display data on screen"
1060 PRINT:PRINT" <4> - Exit to BASIC"
1070 PRINT:PRINT"Enter selection number (1-3) : ";
1080 A$= INKEY$: IF A$= "" GOTO 1080
1090 PRINT A$
1100 IF VAL(A$)=1 THEN GOTO 1580
1110 IF VAL(A$)=2 THEN GOTO 1160
1120 IF VAL(A$)=3 THEN GOTO 1690
1130 IF VAL(A$)=4 THEN CLS:END
1140 PRINT"[",A$,"] is not a valid entry. Please re-enter":GOTO 1070
1150 '
1160 '--- Generate DAS-8 graphics package data file & plot -----
1170 CLS

```



```

1180 LOCATE 1,1:INPUT"DATA FILE NAME [DRIVE]:NAME.EXT";FILX$
1190 OPEN FILX$ AS #1 LEN = 30
1200 FIELD #1, 15 AS X$, 15 AS Y$
1210 '
1220 'Get display mode, dot, line or no plot
1230 LOCATE 3,1:COLOR 0,7:PRINT"L";:COLOR 7,0:PRINT"ine, or ";:COLOR
0,7:PRINT"D";:COLOR 7,0:PRINT"ot plot mode?"
1240 LOCATE 3,29:INPUT"",A$
1250 IF A$="L" OR A$="l" THEN M=1:GOTO 1290
1260 IF A$="D" OR A$="d" THEN M=0:GOTO 1290
1270 LOCATE 4,1:PRINT"RE-ENTER":LOCATE 3,1:PRINT SPC(79):GOTO 1230
1280 '
1290 'Enter number of data points and plot mode in record 1
1300 LSET X$ = MKS$(NC): LSET Y$ = MKS$(M)
1310 PUT #1,1
1320 'Enter data in remaining records
1330 FOR I=2 TO NC+1
1340 LSET X$=MKS$((I-2)/F):LSET Y$=MKS$(A%(I-2)*5/2048)
1350 PUT #1, 1
1360 NEXT I
1370 'Write file
1380 CLOSE #1
1390 'Generate RLINPLT.LNK plotting file for LINPLT.BAS to use
1400 OPEN "RLINPLT.LNK" AS #1 LEN = 30
1410 FIELD #1, 30 AS RLNK$
1420 LSET RLNK$ = MKI$(1) 'one data file
1430 PUT #1,1
1440 CLS
1450 INPUT "Enter graph Y axis label : ",Y$
1460 IF LEN(Y$)>30 THEN Y$=LEFT$(Y$,30)
1470 LSET RLNK$ = Y$ 'Y axis label
1480 INPUT "Enter graph X axis label : ",X$
1490 IF LEN(X$)>30 THEN X$=LEFT$(X$,30)
1500 PUT #1,2
1510 LSET RLNK$ = X$ 'X axis label
1520 PUT #1,3
1530 LSET RLNK$ = FILX$ 'data file name
1540 PUT #1,4
1550 CLOSE #1
1560 RUN"LINPLT 'run plotting program
1570 '
1580 '---- Generate a Lotus 1-2-3 .PRN import file -----
1590 CLS
1600 LOCATE 1,1:INPUT"LOTUS .PRN FILE NAME [DRIVE]:NAME (automatic .PRN ext.) :
";F$
1610 F$ = F$ + ".PRN"
1620 OPEN F$ FOR OUTPUT AS #1
1630 FOR I = 0 TO NC
1640 PRINT #1, STR$(A%(I)*5/2048)
1650 NEXT I
1660 CLOSE #1

```

```
1670 END
1680 '
1690 '---- Display data on screen and return to menu -----
1700 CLS
1710 LOCATE 25,1:PRINT"Press any key to STOP/START display, <ESC> key to return to data
storage menu ";:LOCATE 1,1
1720 PRINT"Time   Channel data (volts)"
1730 FOR I = 0 TO NC
1740 PRINT I/F, A%(I)*5/2048
1750 A$ = INKEY$
1760 IF A$=CHR$(27) THEN I = NC+3
1770 IF A$<>" " THEN GOTO 1830
1780 NEXT I
1790 IF I = NC+3 GOTO 1010
1800 COLOR 0,7:PRINT" Press any key to return to data storage menu ";:COLOR 7,0
1810 IF INKEY$="" GOTO 1810
1820 GOTO 1010
1830 FOR K = 1 TO 50:NEXT K   'delay
1840 IF INKEY$ = "" GOTO 1840
1850 GOTO 1780
1860 END
```

```

5 DIM A(1000), CAV(1000), PRSV(1000), CRANK(1000), VOLT2(1000), PRSV2(1000),
VOL(1000), PRESS(1000)
6 REM - LINE 7 - OPENS DATA FILE THAT CONTAINS 1-D ARRAY
7 OPEN "DATA.BAS" FOR INPUT AS 1
8 REM - LINE 9 - OPENING OUTPUT FILE FOR CALCULATED DATA
9 OPEN "DATAOUT.BAS" FOR OUTPUT AS 2
10 FOR I = 1 TO 2000
11 IF EOF(1) THEN 15
12 F = F + 1
13 INPUT# 1, A(I)
15 NEXT I
16 INPUT "INPUT THE CORRECTED BAROMETRIC PRESSURE";BAR
17 REM - LINE 18 - CONVERTS PRESSURE FROM IN. OF HG TO PSI
18 BAR = BAR* 14.69/29.92
30 REM - LINE 35 - SORTING 1-D ARRAY INTO 2-D ARRAY
35 FOR N = 1 TO F
40 IF (N/2-INT(N/2))=0 THEN J=J+1 : CAV(J) = A(N) ELSE M=M+1 : PRSV(M) = A(N)
42 NEXT N
44 REM - LINES 45 THRU 55 - FINDS A STARTING POINT FOR THE BEGINNING (OR THE
MIDDLE) OF A CYCLE
45 FOR J = 1 TO 10000
50 IF CAV(J) > CAV(J-1) THEN GOTO 60
55 NEXT J
58 REM - LINES 60 THRU 80 - FINDS AND DIMENSIONS ARRAY VOLT2 (CRANK ANGLE)
FOR HALF OF ONE FULL ENGINE CYCLE (ONE REVOLUTION)
60 FOR H = (J) TO 10000
65 K = K+1
70 VOLT2(K) = CAV(H)
75 IF CAV(H) > CAV(H-1) THEN GOTO 110
80 NEXT H
105 REM - LINES 110 THRU 210 - FINDS AND DIMENSIONS ARRAY VOLT2 FOR SECOND
HALF OF CYCLE
110 FOR B = H+1 TO 10000
170 K = K+1
190 VOLT2(K) = CAV(B)
200 IF CAV(B) > CAV(B-1) THEN GOTO 215
210 NEXT B
215 K=0
218 REM - LINES 220 THRU 260 - DIMENSIONING PRESSURE VALUES TO
CORRESPONDING CRANK ANGLE VALUES FOR A PARTICULAR INSTANT IN TIME
220 FOR N = (J+1) TO H
230 K = K+1
250 PRSV2(K) = PRSV(N)
260 NEXT N
265 REM - LINES 270 THRU 350 - SEARCHING FOR LOWEST VALUES OF PRESSURE AND
CRANK ANGLE SENSOR
270 FOR I = 1 TO K
290 IF LOWP > PRSV2(I) THEN LOWP = PRSV2(I)
320 IF LOWC > VOLT2(I) THEN LOWC = VOLT2(I)
350 NEXT I
368 REM - LINE 370 - LENGTH OF CONNECTING ROD

```

```

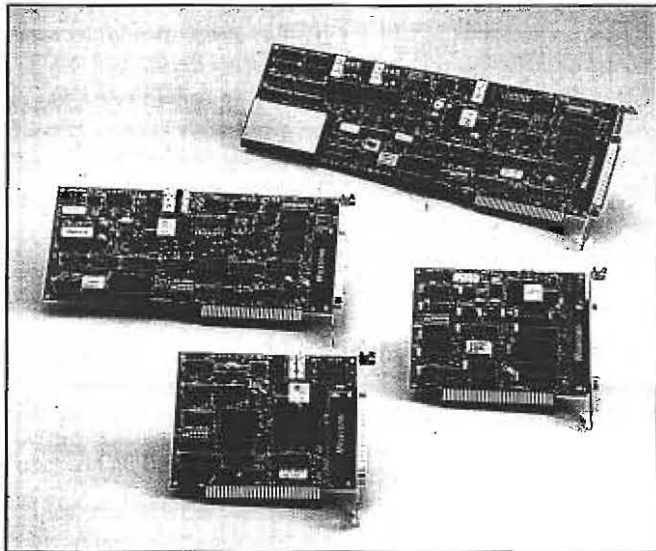
370 LTH = 4.40945
378 REM - LINES 380 THRU 500 - ADJUSTING REPRESENTATIVE VOLTAGES TO ZERO,
CONVERTING TO ACTUAL VALUES OF PRESSURE AND CRANK ANGLE, AND
CALCULATING CYLINDER VOLUME
380 FOR I = 1 TO K
390 LET PRESS(I) = 100*(PRSV2(I) + ABS(LOWP)) + BAR
400 CRANK(I) = VOLT2(I) + ABS(LOWC)
410 IF CRANK(I)>2.5 THEN A=(LTH+1.415-(1.415*COS((5-CRANK(I))*1.256))):B=((LTH^2)-
2*((SIN((5-CRANK(I))*1.256))^2)^.5:VOL(I)=(.8136+(17.89-(6.29*(A-B)))):GOTO 500
460 A = (LTH + 1.415 - (1.415*COS((CRANK(I))*1.25662)))
470 B = ((LTH^2) - 2.002 * ((SIN((CRANK(I)) * 1.25664)^2)^.5
480 VOL(I) = (.8136 + (17.88667 - (6.2902 * (A - B))))
500 NEXT I
510 WORKIN = 0
520 WORKOUT = 0
530 J=0
535 REM - LINES 540 THRU 600 - CALCULATING AREA UNDER CURVES REPRESENTING
EACH STROKE
540 FOR I = 2 TO K
560 IF CRANK(I) < 2.5 THEN WORKIN = WORKIN - (ABS((VOL(I+1) - VOL(I-1))) / 2) *
(PRESS(I+1) + PRESS(I-1)) ELSE WORKOUT = WORKOUT+(ABS((VOL(I+1)-VOL(I-
1)))/2)*(PRESS(I+1)*PRESS(I-1))
600 NEXT I
605 INPUT "ENGINE RPM :",RPM
609 REM - LINE 610 - CALCULATING TOTAL POWER FROM ALL THREE CYLINDERS
610 TOTALPOWER = 3 * (((WORKOUT + WORKIN)/12) * RPM * .5) / (2*33000!)
620 CLS
640 PRINT "INDICATED POWER FROM DAS8 DATA =", TOTALPOWER
650 PRINT "BRAKE HORSEPOWER FROM DYNAMOMETER :", BHP
690 REM - LINES 700 THRU 720 - WRITING DATA TO AN OUTPUT FILE FOR FUTURE USE
IN A SPREAD SHEET
700 FOR I = 1 TO K
710 WRITE #2, CRANK(I), PRESS(I), VOL(I)
720 NEXT I
990 END
□

```

## **APPENDIX C : DAS8 USER'S INFORMATION**

# DAS-8 SERIES

## 4 KILOSAMPLE/SEC ANALOG AND DIGITAL I/O BOARDS



### FEATURES

- 8 analog input channels
- 12-bit resolution
- 7 digital I/O bits (4 out, 3 in)
- 4000 samples per second using Call Driver
- On-board sample pacer clock
- Software-programmable input range (DAS-8PGA, DAS-8PGA/G2 & DAS-8/AO only)
- Two 12-bit analog outputs (DAS-8/AO only)
- Fully differential inputs (DAS-8PGA and DAS-8/AO only)

### APPLICATIONS

- Data logging
- Process control & monitoring
- Energy management
- Laboratory automation
- Production test
- Signal analysis

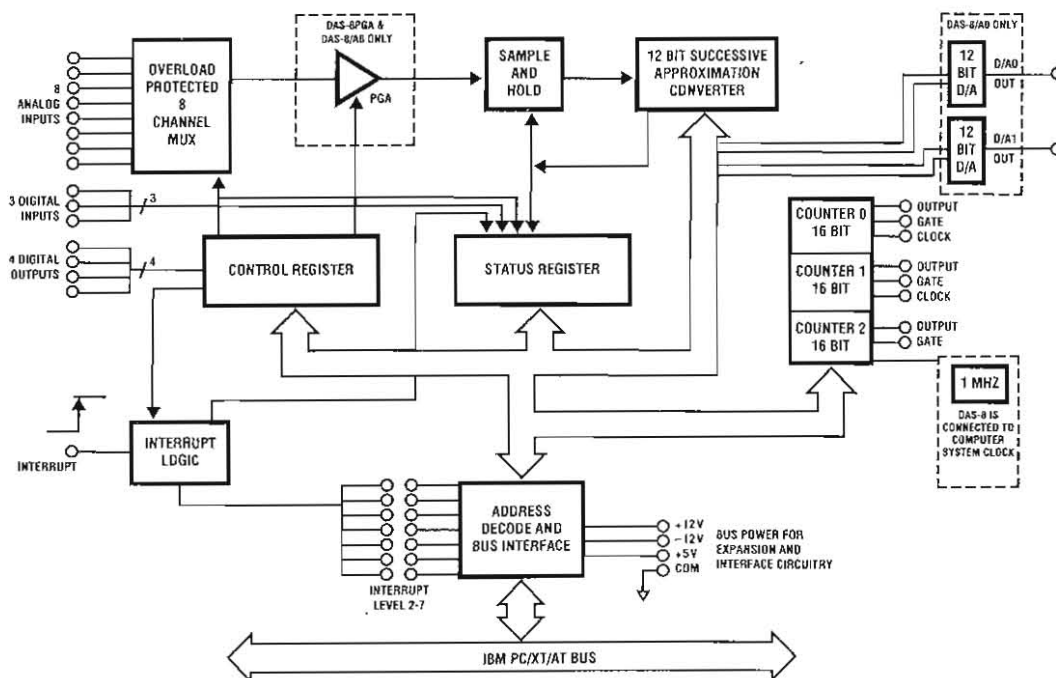
### DAS-8

The DAS-8 features an 8-channel, 12-bit successive approximation A/D converter with sample and hold. The full scale input of each channel is  $\pm 5$  volts with a resolution of 0.00244 volts per bit (2.44 millivolts). Inputs are single-ended with a common ground and will withstand a continuous overload of  $\pm 30$  volts. All inputs are fail safe: i.e., open-circuit when the computer power is off. A/D conversion time is typically 25 microseconds (35 microseconds max.). Using the software driver supplied with the board, throughputs of up to 4000 samples/sec are attainable operating in BASIC.

### FUNCTIONAL DESCRIPTION

The DAS-8 family is a series of 4 ksamples/sec analog and digital I/O boards for IBM PC/XT/AT, 386, 486 and compatible computers. The series consists of the DAS-8, the low power DAS-8LT for laptop and/or portable computers, the DAS-8PGA with software programmable input ranges and the DAS-8/AO with software programmable input ranges and two analog output channels.

### BLOCK DIAGRAM



An on-board 82C54 programmable counter timer provides periodic interrupts for the A/D sample timing. This timer obtains its input clock directly from the computer's system clock (usually 4.77, 6 or 8 MHz). Two additional counters on the 82C54 are available for event counting, pulse and waveform generation, frequency, period and pulse width measurements.

The 7 bits of TTL digital I/O consist of one 4-bit output port and one 3-bit input port. In addition, a precision +10.00 V ( $\pm 0.1$  V) reference voltage output is derived from the A/D converter reference. This output can source/sink 2 mA.

A jumper selectable external interrupt input selects any of the interrupt levels (2 - 7) and allows interrupt routines to provide background data acquisition or interrupt-driven control. The interrupt input can be externally connected to the timer/counter or another trigger source.

On the DAS-8, DAS-8LT and DAS-8PGA, +5 and  $\pm 12$  Volt computer bus power is provided along with all the I/O connections on the rear connector. This simplifies the addition of user-designed interfaces, input signal conditioning circuits, expansion multiplexers, etc. Caution should be exercised to not exceed the power supply capacity of the host computer.

### DAS-8LT

The DAS-8LT is optimized for use in a laptop or portable computer. It uses very little power (205 mW typical, 414 mW maximum) and does not require -12 Volts from the computer. The DAS-8LT is ideal for use in a laptop or portable computer that contains at least one half-card slot, or access to an expansion slot through a docking station. In addition to the standard DAS-8 features, the DAS-8LT incorporates an on-board 1MHz clock for driving the counter.

### DAS-8PGA

In addition to the standard DAS-8 features, the DAS-8PGA and DAS-8/AO provide:

- Software-programmable input ranges of  $\pm 10$ ,  $\pm 5$ ,  $\pm 0.5$ ,  $\pm 0.05$ , and  $\pm 0.01$  V or 0 - 10, 0 - 1, 0 - 0.1 and 0 - 0.02 V. The DAS-8PGA/G2 has programmable ranges of  $\pm 5$ ,  $\pm 2.5$ ,  $\pm 1.25$ ,  $\pm 0.625$  and 0-5, 0-2.5, 0-1.25 and 0-0.625.
- Fully differential analog inputs (switch selectable to Single Ended)
- Pacer clock from an on-board 1 MHz crystal oscillator rather than the computer system clock which changes from computer to computer.

### DAS-8/AO

In addition to the features of the DAS-8 and DAS-8PGA, the DAS-8/AO includes two channels of 12-bit analog output. These D/A channels are independently switch-selectable to full scale ranges of  $\pm 10$ ,  $\pm 5$ ,  $\pm 2.5$ , 0 - 10 or 0 - 5 V, and 4 - 20 mA. Simultaneous update of both channels is switch-selectable.

### SOFTWARE

There are two software alternatives for the DAS-8. A fully-integrated data acquisition software package can be used (e.g., VIEWDAC or EASYEST LX) or a custom program can be written (in BASIC, C, Pascal etc.). A guide to tradeoffs between using integrated packages and writing your own program is provided in the Software Section beginning on page 12. The software section also describes a wide variety of integrated software packages.

The DAS-8 is supported by a comprehensive set of Call Drivers and programming tools. The DAS-8 drivers are provided in two levels, the standard software package, included with each board, and the PCF-8 option.

### STANDARD SOFTWARE

The standard software with the DAS-8 includes the following:

- Mode Call Driver compatible with BASICA and QuickBASIC.
- DAS-8 installation and configuration program.
- Basic example programs demonstrating a variety of DAS-8 functions.
- Complete calibration routine.
- Thermocouple linearization routines written in BASIC.

### Mode Call Driver

The standard Mode Call Driver is designed to be used from Interpreted BASIC, Compiled BASIC or QUICKBASIC. The Call commands are a collection of functions called modes that are accessed from your BASIC program through a single-line Call statement. The various modes select all of the functions of each board, format and error check data and perform frequently used sequences of instructions. The Call Driver provides a great deal of flexibility and allows background data acquisition. Following is an example of the Mode Call Driver usage in interpreted BASIC. This program performs an analog-to-digital conversion on channel 4 of the DAS-8.

```

80  * ..... Initialize the board .....
90  D%(0) = &H300                      ' Base Address = HEX 300
100 D%(1) = 3                          ' Select interrupt level 3
110 MD% = 0                             ' Initialize mode
120 CALL DAS8(MD%, D%(0), FLAG%)
130 IF FLAG% <> 0 THEN PRINT "Error in initialization": STOP
140 * ..... Select the Input Channel .....
150 MD% = 2                             ' Mux channel select mode
160 D%(0) = 4                           ' Set channel to 4
170 CALL DAS8(MD%, D%(0), FLAG%)
180 IF FLAG% <> 0 THEN PRINT "Error in setting channel": STOP
190 * ..... Acquire the Data .....
200 MD% = 4                             ' Select A/D mode
210 CALL DAS8(MD%, D%(0), FLAG%)
220 IF FLAG% <> 0 THEN PRINT "Error in reading channel": STOP
230 PRINT "the data = "; D%(0)
240 END

```

### PCF-8

The optional PCF-8 software provides Mode Call Drivers and programming examples for Pascal, Turbo Pascal, C and Fortran.

### ACCESSORIES

Since the pin-outs of the boards are slightly different, the associated screw terminal boards are also different. The following guide shows the screw terminal interface for each of the boards. The STC-37 is compatible with each of the boards.

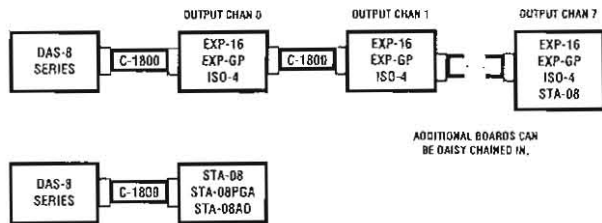
### Accessories

- DAS-8 (STA-08 or STC-37)
- DAS-8LT (STA-08 or STC-37)
- DAS-8PGA or DAS-8PGA/G2 (STA-8PGA or STC-37)
- DAS-8/AO (STA-8AO or STC-37)

In addition, there are a number of external hardware accessories compatible with the DAS-8 series: See the Signal Conditioning and Accessory Section beginning on page 155 for more details.

- EXP-16 16-Channel Expansion and Signal Conditioning Board
- EXP-GP 8-Channel Signal Conditioning Board with Wheatstone Bridge Thermocouple and RTD interface
- ISO-4 4-Channel Isolated Multiplexer for isolated thermocouple or voltage measurements.
- MB-Series Signal Conditioning Modules

## CONFIGURATION GUIDE

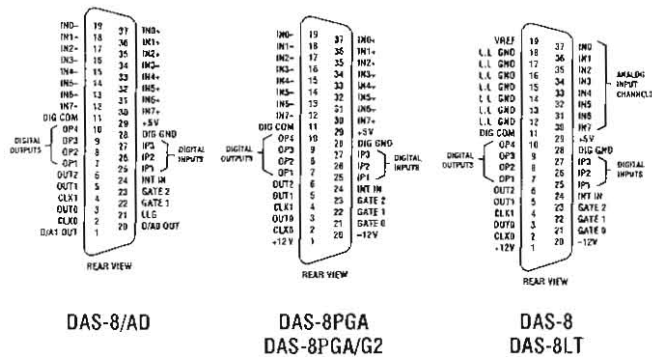


The C-1800 cable is used to daisy chain up to 8 EXP-16, EXP-GP or ISO-4 expansion boards to the DAS-8, DAS-8LT, DAS-8PGA or DAS-8/AO.

## CONNECTOR PIN ASSIGNMENT

The connector pin-outs are slightly different on each of the boards to accommodate differing I/O functions. The connector diagrams of each of the boards are shown here.

All analog and digital I/O is brought out to a standard 37-pin D male connector at the rear panel of the computer. For soldered connections, a 37-pin female connector is required (Keithley Metrabyte part # SFC-37).



## SPECIFICATIONS

### DAS-8 SPECIFICATIONS

#### Analog Specifications

Number of Channels	8, single-ended
Accuracy	0.01% of reading $\pm 1$ bit
Full scale	$\pm 5$ V
A/D Type	Successive approximation
Resolution	12-bits
Conversion	25 $\mu$ s typ (35 $\mu$ s max)
Monotonicity	Guaranteed over operating temperature range
Linearity	$\pm 1$ bit
Coding	Offset binary
Overvoltage	Continuous single channel to $\pm 35$ V

Input impedance	10 M ohms
Input current	100 nA max at 25 °C
Temperature coefficients	
Gain drift	$\pm 25$ ppm/°C max of FS
Zero drift	$\pm 10$ $\mu$ V/°C max

#### Power Requirements

+5 V supply	170 mA typ/283 mA max
+12 V supply	15 mA typ/21 mA max
-12 V supply	25 mA typ/35 mA max

Weight	4 oz (120 gm)
Dimensions	4.25 in x 5.0 in x .75 in (10.8 cm x 12.7 cm x 1.9 cm)

### DAS-8 LT SPECIFICATIONS

#### Analog Specifications

Number of Channels	8, single-ended
Accuracy	0.01% of reading $\pm 1$ bit
Full scale	$\pm 5$ V
Coding	Offset binary
Overvoltage	Continuous single channel to $\pm 35$ V
Input impedance	>10 M ohms
Input current	400 nA max at 25 °C
Temperature coefficient	
Gain drift	$\pm 15$ $\mu$ V/°C max of FS
Settling time	4.50 $\mu$ s to 0.01%, typ for full-scale step input of 10 V
Type	Successive approximation
Resolution	12-bit
Conversion	13 $\mu$ s typ (25 $\mu$ s max)
Monotonicity	Guaranteed over operating temperature range
Linearity	$\pm 1$ bit

#### Reference Voltage Output

Reference	+10.0 V ( $\pm 0.1$ V)
Load current	$\pm 2$ mA (max)

#### Power Requirements

+5 V supply	6 mA typ/30 mA max
+12 V supply	15 mA typ/21 mA max
-12 V supply	Not used
Total power	205 mW typ, 414 mW max

Weight	4 oz (120 gm)
Dimensions	4.25 in x 5.0 in (10.8 cm x 12.7 cm)

### DAS-8/PGA SPECIFICATIONS

#### Analog Specifications

Number of Channels	8 single-ended/differential
Accuracy	0.01% of reading $\pm 1$ bit
A/D Type	Successive approximation.
Resolution	12-bit
Conversion	25 $\mu$ s typ (35 $\mu$ s max)
Monotonicity	Guaranteed over operating temperature range
Linearity	$\pm 1$ bit

DAS-8PGA	Input Ranges	Bipolar	Unipolar	CMRR
		$\pm 10$ V	0 - 10 V	70 dB
		$\pm 5$ V	—	90 dB
		$\pm 0.5$ V	0 - 1 V	100 dB
		$\pm 50$ mV	0 - 100 mV	100 dB
		$\pm 10$ mV	0 - 20 mV	100 dB
DAS-8PGA/G2	Input ranges	$\pm 5$ V	0 - 5 V	78dB
		$\pm 2.5$ V	0 - 2.5 V	84dB
		$\pm 1.25$ V	0 - 1.25 V	90dB
		$\pm 0.625$ V	0 - 0.625 V	100dB
	Coding	Offset binary (Bipolar)		
		True Binary (Unipolar)		
	Input Impedance	> 100 M ohm w/100 pf		
	Temperature coefficients			
	Gain drift	$\pm 50$ ppm/°C max		
	Zero drift	$\pm 10$ $\mu$ V/°C max		



<b>Power Requirements</b>	
+5 V supply	800 mA typ/1 A max
+12 V supply	2 mA typ/5 mA max
-12 V supply	18 mA typ/30 mA max
<b>Weight</b>	
	6.5 ozs (184 gms)
<b>Dimensions</b>	
	4.25 in x 9.0 in x .75 in (10.8 cm x 22.9 cm x 1.9 cm)

**DAS-8/AO SPECIFICATIONS**

<b>Analog Input</b>			
Channels	8 single-ended/differential		
Absolute Accuracy	0.05% of full scale		
Input Ranges	<b>Bipolar</b>	<b>Unipolar</b>	<b>CMMR</b>
	±10 V	0 - 10 V	70 dB
	±5 V	—	90 dB
	±0.5 V	0 - 1 V	100 dB
	±50 mV	0 - 100 mV	100 dB
	±10 mV	0 - 20 mV	100 dB
Input common mode	6.5 V		
Input impedance	>100 M ohm in parallel with 100 pF		
Leakage current	250 nA max		
Linearity	±1 bit		
<b>Temperature coefficients</b>			
Gain drift	35 ppm of FS/°C max		
Zero drift	20 ppm of FS/°C		
Noise	30 $\mu$ V p-p typ		
A/D Type	Successive approximation		
Resolution	12-bit		
Conversion	25 $\mu$ s typ (35 $\mu$ s max)		
Monotonicity	Guaranteed over operating temperature range		
Linearity	±1 bit		

<b>Analog Output</b>	
Channels	2
Resolution	12-bits
Ranges	0 - 5 V, 0 - 10 V, ±2.5 V, ±5 V, ±10 V
Settling time	4 $\mu$ s to 0.01% for full-scale step
Input data decoding	Unipolar — True binary (right-justified) Bipolar — Offset binary (right-justified)
Output current	± 5 mA
Output resistance	<0.1 Ohm
Gain error	Adjustable to zero
Offset drift	Bipolar = 4 ppm/°C Unipolar = 10 ppm/°C
Gain drift	30 ppm of FSR/°C
4 - 20 mA current loop	
Type	4 - 20 mA constant-current sink to ground
Output resistance	> 100 M ohm
Loop excitation voltage	+6 V min +36 V max

<b>Power Requirements</b>	
+5 V supply	800 mA typ/1100 mA max
+12 V supply	8 mA typ/10 mA max
-12 V supply	Not used

<b>Weight</b>	9 oz (270 g)
<b>Dimensions</b>	4.25 in x 13.25 in (10.8 cm x 33.7 cm)

**DAS-8 FAMILY SPECIFICATIONS**

(Specs common to all boards)	
<b>Digital I/O</b>	
Outputs	LSTTL
Output Port	4 bits
Low Voltage	0.5 V max at Isink = 8.0 mA
High Voltage	2.4 V min at Isource = -0.4 mA
Inputs (IPI - 3 & Interrupts)	LSTTL
Low Voltage	0.8 V max
Low Current	-0.4 V Imax
High Voltage	2.0 V min

High Current	20 $\mu$ A max
<b>Interrupt Channel</b>	
Level	2 - 7 software selectable.
Enable	Via INTE of control register.
<b>Programmable Timer</b>	
Type	8254
Counters	3 down counter, 16-bit
Output Drive	2.2 mA at 0.45 V (5 LSTTL loads)
Input, Gate:	TTL/DTL/CMOS compatible
Clock Frequency	1 MHz $\pm$ 0.1% (DAS-8 uses computer's system clock)
Active Edge	Negative

<b>Environmental</b>	
Operating Temp.	0 to 50 °C
Storage Temp.	-20 to 70 °C
Humidity	0 - 95% non-condensing



Supported by EASYEST LX, EASYEST AG, VIEWDAC and ASYST software. See Data Acquisition & Analysis Software section starting on page 12.

**ORDER**

DAS-8	4 kS/s Analog and Digital I/O Board with software on 5.25 inch disks
DAS-8/3.5	DAS-8 with software on 3.5 inch disks
STA-08	Screw Terminal Accessory for DAS-8
C-1800	DAS-8 to STA-08 Cable
DAS-8LT	Low-Power, 4 kS/s Analog and Digital I/O Board with software on 5.25 inch disks
DAS-8LT/3.5	DAS-8LT with software on 3.5 inch disks
STA-08	Screw Terminal Accessory Board for DAS-8LT
C-1800	DAS-8LT to STA-08 Cable
DAS-8PGA	4 kS/s Analog and Digital I/O with gains of .5, 1, 10, 100 and 500, software on 5.25 inch disks
DAS-8PGA/3.5	DAS-8PGA with software on 3.5 inch disks
DAS-8PGA/G2	4 kS/s Analog and Digital I/O Board with gains of 1, 2, 4 and 8, software on 5.25 inch disks
DAS-8PGA/G2/3.5	DAS-8PGA/G2 with software on 3.5 inch disks
STA-08PGA	Screw Terminal Accessory Board for DAS-8PGA
C-1800	DAS-8PGA to STA-08PGA Cable
DAS-8/AO	4 kS/s Analog and Digital I/O Board with Analog Output, software on 5.25 inch disks
DAS-8/AO/3.5	DAS-8/AO with software on 3.5 inch disks
STA-08AO	Screw Terminal Accessory Board for DAS-8/AO
C-1800	DAS-8AO to STA-08AO Cable
PCF-8	Pascal, C and Fortran Driver for the DAS-8. Discount applied if purchased with the board on a single purchase order.
TTTOOLS	Turbo Pascal Driver package