

AD-A081 083

NAVAL POSTGRADUATE SCHOOL MONTEREY CA

F/G 14/2

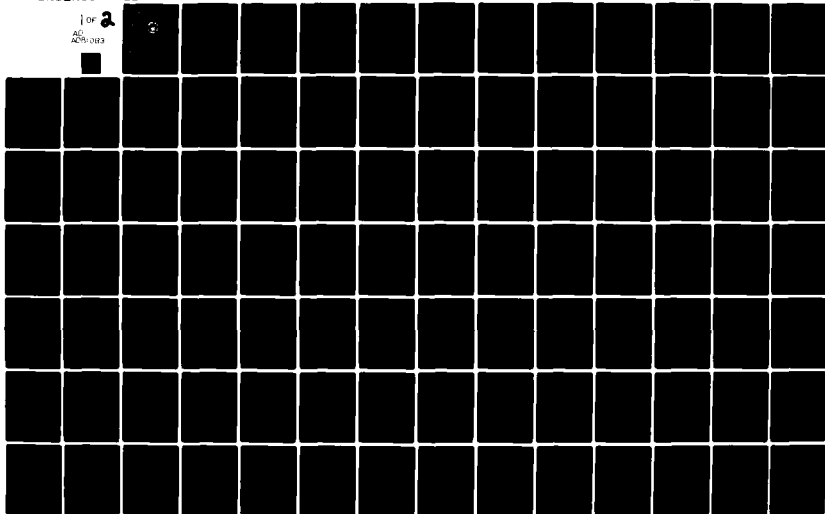
DEVELOPMENT OF A MICROPROCESSOR-BASED INSTRUMENT FOR STATIC TES--ETC(U)

SEP 79 A H BARBER

UNCLASSIFIED

NL

1 of 2
AD
A081 083



ADA 081 083

NAVAL POSTGRADUATE SCHOOL
Monterey, California



THESIS

S

FEB 26 1980

A

DDC FILE COPY

DEVELOPMENT OF A MICROPROCESSOR-BASED
INSTRUMENT FOR STATIC TESTING SMALL
ROCKET ENGINES

by

Arthur Houghton Barber, III

September 1979

Thesis Advisor:

M. L. Cotton

Approved for public release; distribution unlimited.

80 2 13 117

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle)		5. TYPE OF REPORT & PERIOD COVERED
(6) Development of a Microprocessor-Based Instrument for Static Testing Small Rocket Engines.		Master's Thesis September 1979
7. AUTHOR(s)		8. PERFORMING ORG. REPORT NUMBER
(10) Arthur Houghton / Barber, III		9. CONTRACT OR GRANT NUMBER(s)
10. PERFORMING ORGANIZATION NAME AND ADDRESS		11. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
Naval Postgraduate School Monterey, California 93940		(12) 100
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE
Naval Postgraduate School Monterey, California 93940		11 September 1979
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		13. NUMBER OF PAGES
Naval Postgraduate School Monterey, California 93940		14. SECURITY CLASS. (of this report)
		Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)		
Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)		
Solid rocket testing Microprocessor Instrument Static test stand Rocket engine testing Pyrotechnics testing		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)		
→ A microprocessor-based instrument was developed for static testing solid-fuel rocket engines having peak thrusts of 130 Newtons or less and total impulses of up to 100 Newton-seconds. It measured peak thrust, total impulse, burn time, pyrotechnic delay time, and maximum casing external temperature, all to relative accuracies of two percent of the smallest expected values. This corresponds to better than 0.1 percent of →		

DD FORM 1473
1 JAN 73
(Page 1)EDITION OF 1 NOV 68 IS OBSOLETE
S/N 0102-014-6601

1

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE/When Data Entered

full scale. The instrument was designed for minimum parts cost and for portable operation from two twelve-volt batteries. It may be easily modified to test engines with five to ten times greater thrusts and total impulses.

Accession No.	
HTB GLENN	☒
DDC TAB	☐
Unpublished	☐
Justified	☐
In _____	

A	

DD Form 1473
Jan 73
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE/When Data Entered

Approved for public release; distribution unlimited.

Development of a Microprocessor-Based Instrument
for Static Testing Small Rocket Engines

by

Arthur Houghton Barber, III
Lieutenant, United States Navy
B.S., MIT, 1973

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1979

Author

Arthur Houghton Barber III

Approved by:

Mitchell S. Cotton
Thesis Advisor

R. Paul Rogers
Second Reader

David W. Hovington, (acting)
Chairman, Department of Electrical Engineering

William M. Folles
Dean of Science and Engineering

ABSTRACT

A microprocessor-based instrument was developed for static testing solid-fuel rocket engines having peak thrusts of 130 Newtons or less and total impulses of up to 100 Newton-seconds. It measured peak thrust, total impulse, burn time, pyrotechnic delay time, and maximum casing external temperature, all to relative accuracies of two percent of the smallest expected values. This corresponds to better than 0.1 percent of full-scale. The instrument was designed for minimum parts cost and for portable operation from two twelve-volt batteries. It may be easily modified to test engines with five to ten times greater thrusts and total impulses.

TABLE OF CONTENTS

I.	INTRODUCTION-----	9
II.	DESIGN REQUIREMENTS-----	12
	A. INPUT CHARACTERISTICS-----	12
	B. CONSTRAINTS ON PARTS SELECTION-----	15
	C. ACCURACY REQUIREMENTS-----	18
	D. NOISE MINIMIZATION-----	20
III.	TRANSDUCER AND ANALOG SYSTEM-----	23
	A. THRUST TRANSDUCER DESIGN-----	23
	B. AMPLIFIER AND FILTER DESIGN-----	30
	C. TEMPERATURE TRANSDUCER DESIGN-----	35
IV.	DIGITAL SYSTEM HARDWARE-----	43
	A. INTERFACE CIRCUITRY-----	43
	B. MICROPROCESSOR AND MEMORY SYSTEMS-----	48
	C. INPUT/OUTPUT CIRCUITRY-----	54
V.	SYSTEM SOFTWARE-----	60
	A. STRUCTURE AND GENERAL FEATURES-----	60
	B. SOFTWARE A/D CONVERSION-----	65
	C. MAIN OPERATING PROGRAM-----	67
	D. SUBROUTINES AND MINOR PROGRAMS-----	75
VI.	CONCLUSION-----	79
	APPENDIX A. CIRCUIT SCHEMATICS-----	81
	APPENDIX B. INSTRUMENT OPERATING INSTRUCTIONS-----	88
	COMPUTER PROGRAM-----	90

LIST OF REFERENCES-----103

INITIAL DISTRIBUTION LIST-----104

LIST OF TABLES

I.	Characteristics of Design Inputs to System-----	12
II.	Characteristics of Transducer Beams with Center Mass-----	29
III.	Amplifier Performance for Gain of 1000-----	33
IV.	System Memory Map-----	51
V.	Decoding of LED/Comparator Select Lines-----	57
VI.	Decoding of Data Routing Control Lines-----	58
VII.	Functions of Input Keys-----	61

LIST OF FIGURES

1.	System Block Diagram-----	11
2.	Example Shapes of Thrust-time Inputs-----	13
3.	Thrust Transducer-----	26
4.	General Thermocouple Circuit Characteristics-----	39
5.	Four-bit Straight-binary Successive Approximation A/D Conversion-----	68
6.	Flowchart of Main Operating Program-----	70
7.	Thrust Transducer Amplifier and Filter Circuit-----	81
8.	Thermocouple Amplifier and Filter Circuit-----	82
9.	Thermocouple Compensation and Peak Detection Circuit-----	83
10.	Digital Interface Circuit-----	84
11.	Microprocessor and its Support Circuitry-----	85
12.	Memory System-----	86
13.	Digital Input/Output Circuit-----	87

I. INTRODUCTION

The object of this thesis effort was to develop a minimum-cost, field-transportable instrument for static testing small solid-fuel rocket engines and accurately measuring several of their performance parameters. The system was specifically tailored to test model rocket engines for compliance with various state and Federal safety requirements and international standards for use in record attempts. For these purposes, measurements of total impulse, peak thrust, thrust duration, pyrotechnic delay duration, and casing external temperature were required, all to an accuracy of two percent of the peak value in each test. This instrument could be used with little modification to test any type of rocket engine having a thrust of 150 pounds or less and a total impulse of less than 200 pound-seconds.

The various parameters of interest here could certainly all have been measured with purely analog electronics. This could, for example, have been done by use of a multi-channel fast-response chart recorder. Such a recorder is very expensive and not easily portable, and its accuracy depends on proper selection of the scales to be used, which requires some advance knowledge of the performance expected from the test item. A microprocessor-based digital system has none of these disadvantages, and this is why the more complex digital design was used.

At the expense of some complexity in software, the system developed for this thesis delivered the required accuracy without advance scale selection when used to test a wide variety of rocket engines. In order to minimize parts costs, the system was developed as a single-purpose instrument rather than as an adaptation of an existing microcomputer.

The instrument consisted of a mechanical force transducer and a thermocouple temperature sensor, followed by analog amplifying and filtering circuits and a digital processing system. This microprocessor digital system performed analog-to-digital conversion, detection or computation of the values of the five parameters of interest, storage of the digital data representing thrust-vs.-time history, and display (under operator control) of the measured data. Figure 1 shows the relationship among the various components of the system.

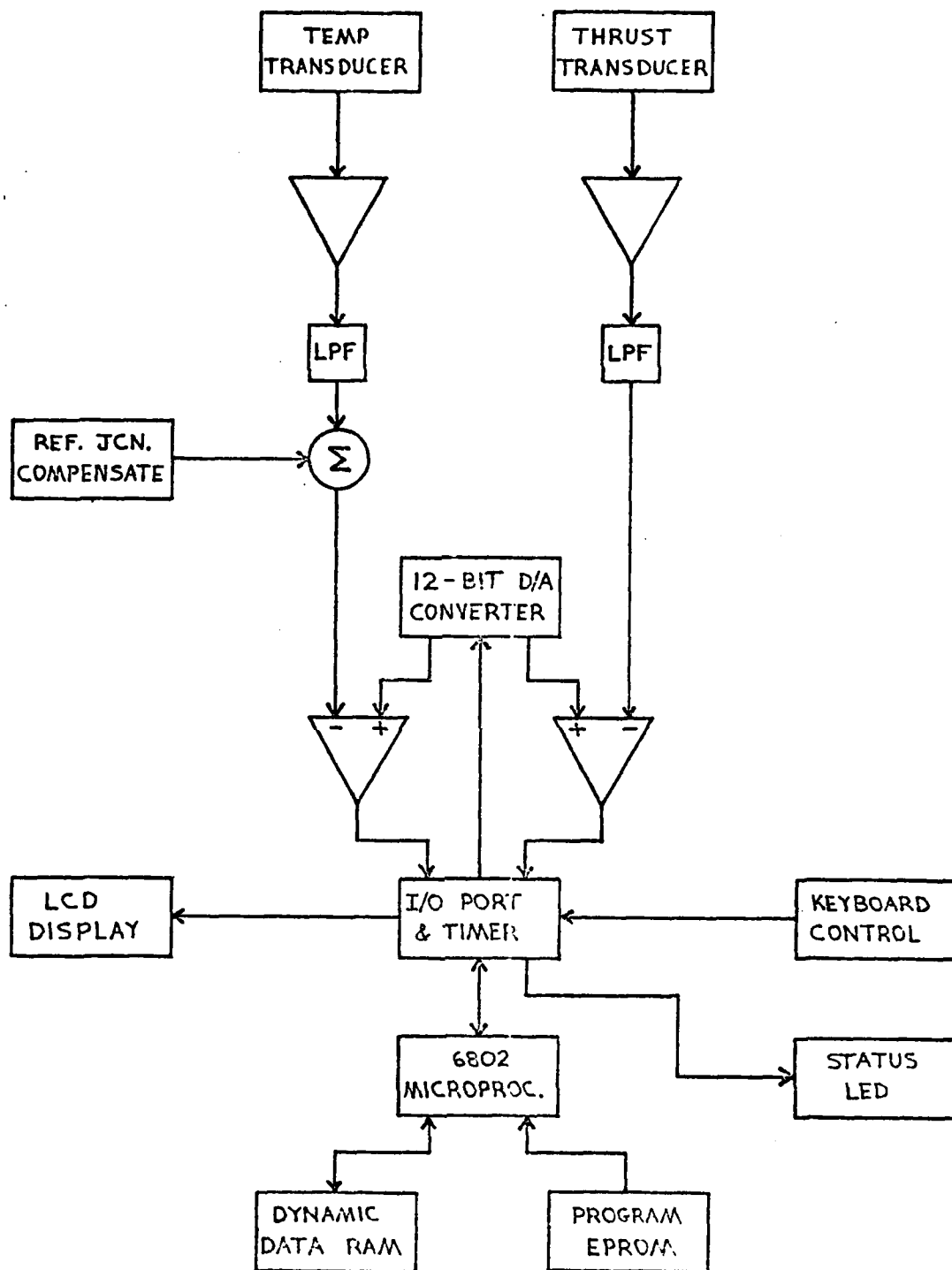


Figure 1. System block diagram.

II. DESIGN REQUIREMENTS

A. INPUT CHARACTERISTICS

The engines which this system was designed to test come in a wide variety of powers, sizes, and durations, and have many different variations of thrust with time. Information on the general values of these parameters is not always available before a test, and even if it is not the parameters must be measured to an accuracy of two percent of their value. Consequently, this system was designed to have a maximum error of roughly two percent of the smallest value expected for each parameter. This gave it a performance on larger engines much better than the minimum requirements. The range of variation of each parameter which was used in establishing the design is given in Table I. Some typical shapes for the variations of thrust with time are given in Figure 2.

TABLE I
CHARACTERISTICS OF DESIGN INPUTS TO SYSTEM

Parameter	Units	Expected Values	
		Minimum	Maximum
Peak Thrust	Newtons	6.0	130.0
Average Thrust	Newtons	2.0	75.0
Total Impulse	Newton-seconds	0.50	99.99
Thrust Duration	seconds	0.20	9.50
Delay Duration	seconds	0.00	15.00
Casing Temperature	degrees C	25.0	250.0

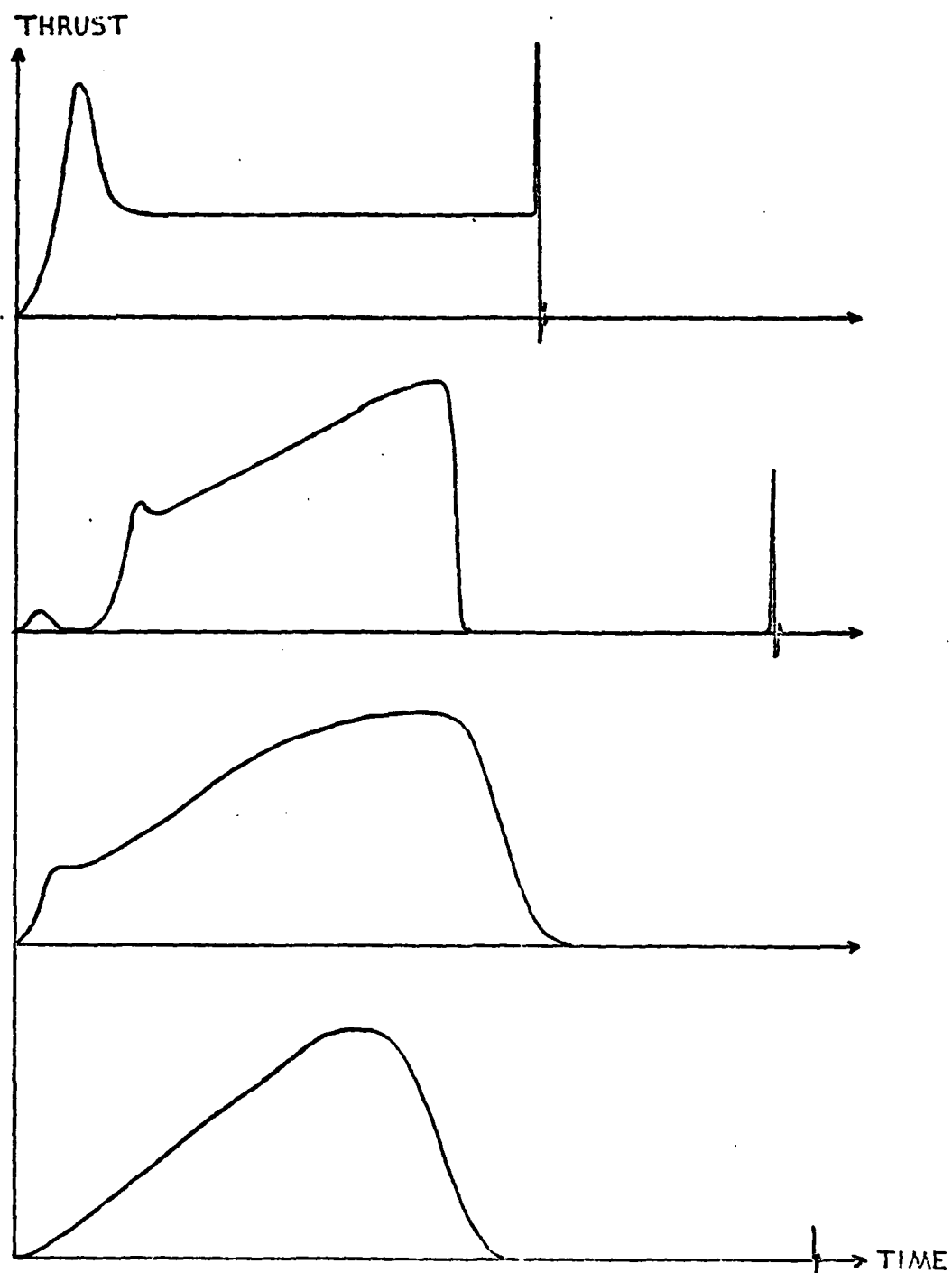


Figure 2. Example shapes of thrust-time inputs.

Model rocket engines include a small gas-generating charge for activation of recovery systems, which may go off from 0 to 15 seconds after the propellant has burned out. The "thrust" spike which this produces on a test stand must not be considered when computing total impulse or peak thrust. With some types of engines there is an igniter-produced "thrust" spike prior to actual propellant ignition, and this must be ignored when computing thrust duration and total impulse. The system software had to be designed to perform both of these tasks.

A major consideration in the design of a digital data-collection system is the choice of a sampling rate. This rate must be at least twice the highest frequency component in the analog signal being sampled to avoid aliasing. The sampling rate has a great effect on the amount of memory required to store the samples and on the technique and hardware used in analog-to-digital conversion. Consequently, its selection was the first step in the design of the digital portion of this system. The natural frequency of the mechanical transducer used in this system was about 500 Hz, permitting sampling rates of up to 1000 Hz.

In order to determine the minimum acceptable sampling rate, the thrust transducer (described later) was built and was connected to an amplifier circuit and a variable-frequency four-pole low-pass filter. The direct output of the amplifier was connected to one channel of a Honeywell 906C

Visicorder recording oscillograph, and the filtered output was connected to a second channel. The recorder had flat frequency response to beyond 1000 Hz, and hence did not affect the results in this test. Various engines were fired in the transducer, and the recorder outputs before and after the filter were compared visually. The filter was set at various values from 500 Hz to 200 Hz. No significant change in the shape of the output by filtering was detected for any engine at any filter setting. Clearly the engine thrust variation contained no important information above 200 Hz and could safely be low-pass filtered at this value. A sampling rate of 500 Hz was chosen.

Casing temperature changes much more slowly than the other parameters measured by this system. Typically, it does not begin to rise until after the propellant has burned out, and does not reach its peak value until 30 to 70 seconds later. The sampling rate required for this parameter is not easily compatible with the high rate required for thrust sampling, so an analog peak detection circuit was used to hold the voltage corresponding to peak temperature for later one-time sampling by the digital system.

B. CONSTRAINTS ON PARTS SELECTION

This system was designed to be field-portable, that is, operable from a pair of batteries connected to provide +12 V and -12 V with no more than one ampere of current drain on either battery. In addition, it was desired to minimize

the total parts cost even if this required a modest increase in software complexity and in hardware assembly time.

The voltage requirements posed no great difficulty, except in the selection of the analog-to-digital conversion hardware. Most A/D and D/A converters with the speed and number of bits required use either a +15V or a -15V power supply, or both. A suitable and inexpensive D/A converter was eventually obtained from Burr-Brown Research.

The current requirements dictated that Schottky low-power transistor-transistor logic (LS-TTL) be used in place of standard TTL in the digital circuit. CMOS logic was not used because of its incompatibility with many of the microprocessor system and other major special-purpose chips. The difference in price and speed between LS-TTL and standard TTL was too small to outweigh the major differences in current consumption. Current requirements also determined the choice of a liquid crystal display (LCD) for output rather than light-emitting diodes (LED). Although an LED display would have been substantially cheaper (\$9. versus \$20.) and slightly less complex, a four-digit display of reasonable size would have drawn at least 0.5 ampere and would have been unreadable in sunlight during field use. The reflective LCD that was chosen uses less than three milliamperes and is not affected by sunlight.

There were three major constraints on the selection of the microprocessor: cost, development system support, and complexity. Performance was not a major issue, since this

application was neither exceptionally fast nor dependent on large arithmetic computations. Nearly any eight-bit microprocessor would have had adequate performance. At the time that this project was undertaken, microprocessor development system (MDS) support was available at the Naval Postgraduate School for only three CPU chips: the Intel 8080A, the Zilog Z-80, and the Motorola 6800. Because of the complexity of the software required for this project, selection of a CPU not supported by a sophisticated MDS would have been unwise.

The features required from the microprocessor system were: 1) 13 or more I/O ports; 2) a programmable timer; and 3) 128 x 8 or more of static RAM (memory) for scratchpad and stack use. It was determined that such a system using the Z-80 would have used five chips and cost about \$55., while one using the 8080A would have used six chips and also cost \$55. A system using the 6802 (almost pin-for-pin compatible with the 6800) could be assembled with two chips for \$48., and this design was selected.

Parts cost was also a factor in the design of the analog amplifier section of this system, and in the selection of the analog-to-digital conversion technique. In both cases, single modules existed which would have met all of the performance requirements. Instead of a monolithic instrumentation amplifier for \$17., a set of three operational amplifiers (total parts cost \$5.) was used for each transducer. Rather than a fast monolithic 12-bit A/D converter module costing \$85., a good 12-bit D/A converter and a comparator were used

(parts cost \$30.) with idle microprocessor computing capacity being employed to generate successive approximation A/D logic. In both cases, the result was a system which met every performance requirement. The amount of effort expended in wiring and in software development to achieve this, however, was so great that the savings in parts cost was not justified.

Design of the RAM system for storing the thrust data from the analog-to-digital conversion was a final major area where cost and current requirements determined the design. A 16K x 4 or 8K x 8 memory was required. A static memory of this size using the most economical memory chips available would have cost about \$100. and consumed one ampere. The dynamic memory that was chosen cost \$55. and consumed 0.15 ampere. Once again, these savings were realized at the expense of added software complexity.

C. ACCURACY REQUIREMENTS

Once the magnitudes of the expected inputs were defined, it was possible to calculate the accuracy required of the system as a fraction of full-scale values, and thus the number of bits of analog/digital conversion required.

The accuracy needed in peak thrust measurement was two percent of the minimum expected peak thrust (6.0 N), or 0.12 Newtons. Full-scale was 130 N, so on this basis the least significant bit in A/D conversion had to represent one part in $130/0.12 = 1083$. This is 10.1-bit accuracy.

Total impulse is simply the integral (digitally, the sum) of thrust with time, so its error is just the error in thrust multiplied by the duration of the thrust. A small thrust error in a long-burning engine can add up to a large total impulse error. However, if the thrust during this time is much larger than the error, the total impulse error will be only a small percentage. The error effect here depends on the average thrust of the engine. If thrust is measured to an accuracy of two percent of the minimum expected average thrust, then assuming no timer or arithmetic errors the total impulse error can never exceed two percent. Referring back to Table I, 2 percent of 2.0 N is 0.04 Newtons. This is the accuracy of thrust measurement that must be achieved to guarantee accuracy of total impulse to two percent of its minimum expected value, or, $.02 \times .50 = .01$ N-sec. As a fraction of peak expected thrust, this requirement is $130/.04 = 3250$, or 11.7-bit accuracy. This more stringent requirement supersedes the 10.1 bits needed for peak thrust. A 12-bit conversion was chosen for thrust measurement.

For the temperature system, the maximum acceptable error was 3°C. There were nonlinearities and errors in the sensor itself (discussed later) which could introduce at least 2°C of error, so it was desired to hold A/D quantization error to 1°C or less. Since the maximum value was 250°C, this was one part in 250, or eight bits. For this system, only half of the -5V to +5V range of the A/D conversion input was used,

so relative to the full range nine bits of accuracy were required. Actually, the same twelve-bit system was used here as was used for thrust, and the last three bits were ignored.

Event duration timing in this system was done with a programmable timer driven by the system clock, which was crystal-controlled. The only error here was the 0.002-second quantization error, this being the interval between timer-produced interrupts. This was negligible.

D. NOISE MINIMIZATION

The system developed in this project was intended to be a highly accurate instrument. The accuracy requirements placed on it were such that its analog portion had to be capable of resolving microvolt signals from the transducers, and its digital portion had to have less than one-half bit of error in a 12-bit A/D conversion. This meant that careful attention had to be paid to noise minimization from the beginning of the design process. Reference 1 was a particularly useful source of design techniques for noise reduction. The design goal was a true-RMS noise voltage output of no more than 1/4 of the least significant bit value. For the thrust system, this was 0.6 millivolts. For the temperature system, it was 4.8 millivolts.

The ground system of the transducer and analog portion of the instrument was designed before the rest of the circuit. The entire instrument was built inside a covered metal chassis

to shield the circuits from external electric fields. This chassis was connected at only a single point to the other parts of the instrument's ground system. Ground conductors to all of the elements of the analog system were run from this central grounding point, with no more than two circuit elements being connected in series on the same conductor. This kept noise voltage from being induced in the ground of one element by the return current from another element flowing through the resistance of a shared ground wire.

Within all of the high-gain first-stage amplifier circuits, metal film one percent resistors were used to minimize thermal noise and drift. In addition to their anti-aliasing function, the low-pass filters at the outputs of the transducer amplifiers were used to eliminate high-frequency noise.

The cables between the circuit chassis and the transducers were twisted-pair conductors with braided metal shields. The shields were not used as signal conductors or grounds. They were isolated from system ground at the transducer end and connected to it at the chassis end, inside the chassis. When connected in this manner, the shields minimized noise coupling into the signal conductors from both ground loops and external electric fields.

The external +12V and -12V power supply inputs were bypassed to system ground with several parallel capacitors ranging from 10 microfarad electrolytic to .001 microfarad ceramic, to attenuate power supply noise over a wide frequency range. Within the system, separate heavily-bypassed

+5V and -5V power supplies were used for the digital and analog systems to prevent TTL switching noise from coupling into the analog elements.

In the digital portion of the instrument, a copper-clad circuit board was used to provide a good ground plane, which was connected to the system ground at one point with a large, flat conductor. Flat metal strips were used as digital power buses to provide maximum capacitive coupling to the ground plane, and hence minimum characteristic impedance $Z_0 = \sqrt{L/C}$. These buses were also bypassed to ground with large capacitors. The power leads of the TTL chips were all bypassed to ground with .015 microfarad capacitors to provide current for switching. Each TTL chip was connected to a power bus through its own unshared conductor.

III. TRANSDUCER AND ANALOG SYSTEMS

A. THRUST TRANSDUCER DESIGN

The key step in the design of this instrument system was the selection of a technique for converting rocket engine thrust to an electrical signal. Each of the many types of device that are used as force-to-signal transducers has different requirements for output processing circuitry, so the design of this circuitry must await selection of the transducer. The transducer for this system had to meet the following requirements:

- 1) sufficient output to permit the .04 N minimum detectable thrust element to be converted to one bit in a 12-bit, 10-volt A/D conversion (2.44 millivolts) using an amplifier gain of no more than 1000,
- 2) ability to survive a force of 260 Newtons in case of an engine malfunction (explosion) during a test,
- 3) natural frequency of not less than 500 Hz with a 45-gram engine in place, to guarantee that transducer frequency response effects would not obscure the performance of engines under test,
- 4) ability to withstand prolonged exposure to corrosive engine exhaust gases;
- 5) low power consumption and minimum cost.

Although a wide variety of techniques are available for force transduction, relatively few of them could meet the frequency response and dynamic range requirements of this system. The high cost of suitable commercially-available units made it desirable to use a transducer that could be

built locally. Most such transducers contain a mechanical element which is deflected by the force, with this displacement leading to a change in the resistance, capacitance, or inductance of some attached device or to the deflection of a light beam. Of these techniques, strain gauges (which change resistance) are the least complex and the cheapest. It was decided to try these first.

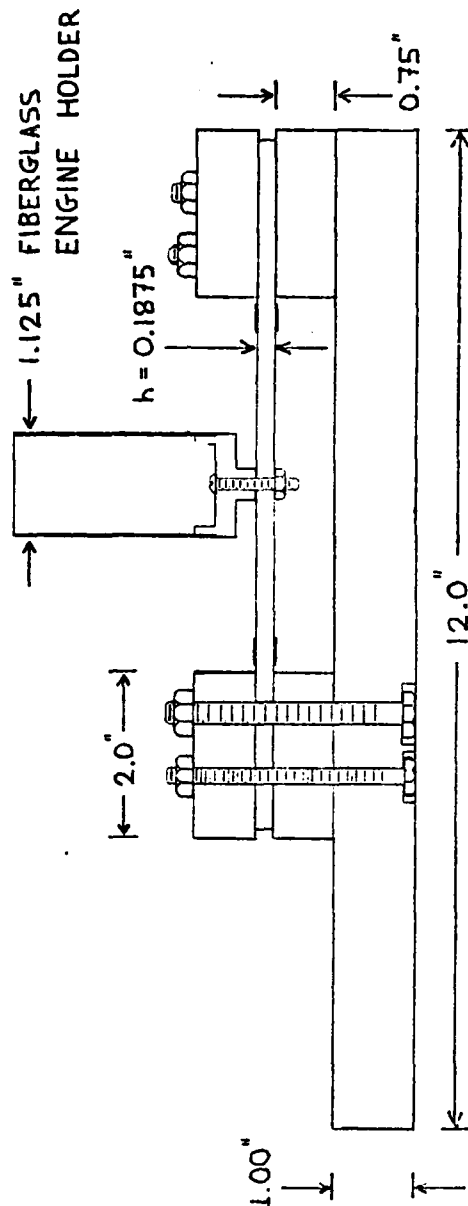
Many commercial and laboratory force-sensing systems are based on strain gauges, so the techniques for their use are well developed. These gauges are extremely thin grids of copper or constantan which are adhesively bonded to a surface. As the surface deforms under the application of a stress, the gauge metal deforms and changes resistance. The fractional change in resistance, and hence in the output, is proportional to the mechanical strain, $\Delta R/R = g\epsilon$, where g is the gauge factor of the strain gauge (usually approximately 2.1) and ϵ is the fractional elongation of the material, or the strain, $\epsilon = \Delta l/l$. As long as the thermal expansion coefficient of the gauge material is matched to that of the surface to which it is applied, thermal gradients will have little effect on gauge accuracy. By arranging two or four gauges on the test item so that equal numbers are in tension and compression, and by wiring them in a Wheatstone bridge arrangement, the electrical output of the gauges is maximized and thermal expansion errors are further reduced. This is discussed in detail in Ref. 2.

The mechanical element to which the gauges are bonded totally determines the linearity, magnitude, and frequency response of their output. The shapes that can be used for this element include rings, tubes, rods, octagons, and cantilevers. The rectangular cross-section cantilever is the easiest to analyze and fabricate, so it was the first choice. Cantilevers may be rigidly supported either on just one end (clamped-free) or on both (clamped-clamped). Both types were examined, and the clamped-clamped design was chosen because it had a higher natural frequency for a given level of strain output.

The equation for the stress in a cantilever as a function of its dimensions and material properties and the force F applied at its center was obtained from Ref. 3. Of particular importance is the linear relationship between input (force) and output.

$$\sigma = \frac{3Fl}{4bh}$$

The dimensions l , b , and h are illustrated in Figure 3. Equations were also required which predicted the natural frequency of the beam, since this was a major design parameter. These are given in handbooks such as Ref. 3 only for beams without attached masses, whereas the transducer beam had an engine holder and engine in its center. An approximate solution was developed from the known no-mass natural frequency formula and from the general principle that



ALL STRUCTURAL PARTS ARE ALUMINUM

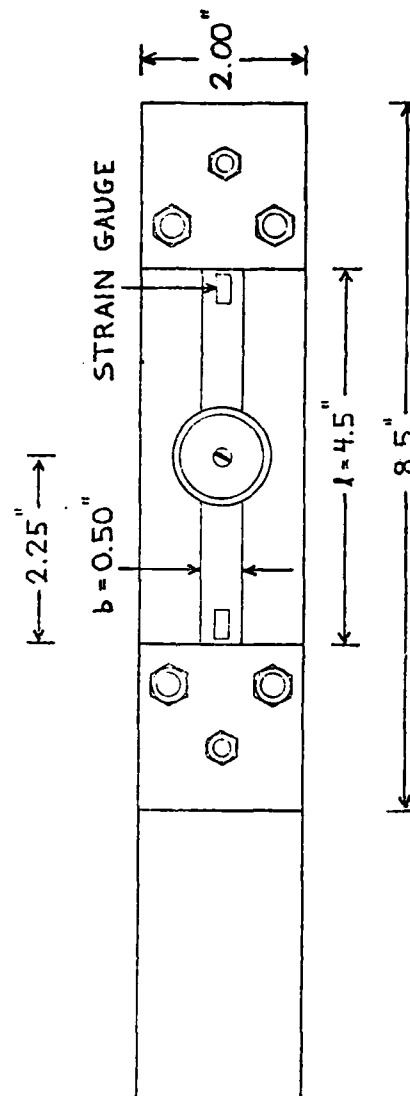


Figure 3. Thrust transducer.

NOT REPRODUCIBLE.
 THIS DOCUMENT CONTAINS INFORMATION WHICH DO NOT
 REVEAL THE DESIGN OR OPERATION OF THE THRUST TRANSDUCER.

natural frequency ω_n is related to a beam's distributed mass m_b and spring constant K by:

$$\omega_n = \sqrt{\frac{K}{\alpha m_b}} \quad , \quad m_b = \rho b h l, \quad K = \frac{16 E b h^3}{l^3}$$

where α is a constant. The exact equation for the fundamental frequency of an unloaded clamped-clamped cantilever was obtained from Ref. 3 as:

$$\omega_n = \frac{127.6 h}{l^2} \sqrt{\frac{E}{\rho}}$$

All units are pounds and inches. E is Young's modulus for the material used and ρ is its density.

These equations were set equal, and the value of α was found to be $\alpha = .0009827$. It was then assumed that when a point mass m_c was added at the center of the beam, the natural frequency would become:

$$\omega'_n = \sqrt{\frac{K}{\alpha m_b + m_c}}$$

By examining the equations for stress and natural frequency together, it is apparent that tradeoffs are required in getting both the desired strain $\epsilon = \sigma/E$ and natural frequency ω'_n . Increasing Young's modulus by choice of the beam material will increase ω'_n and reduce ϵ . Increasing beam length l will increase ϵ but sharply reduce ω'_n . Increasing beam thickness h will increase ω'_n but

sharply reduce ϵ . Increasing beam width b will reduce ϵ and not affect ω'_n . In general, high strain ϵ means low natural frequency ω'_n .

Clearly, a minimum width and length beam was desired. However, enough width had to be provided to permit strong bolt attachment to supports at the ends. Also, enough length had to be allowed for development of a reasonable strain at the strain gauges, whose centers could be no closer than about 0.25" to the ends of the beam. A width $b = 0.5$ " and a length $l = 4.5$ " were selected on this basis. Using four strain gauges in a Wheatstone bridge with two in tension and two in compression, the output is:

$$\frac{\Delta V}{V_0} = \frac{\Delta R}{R} = g\epsilon = g\frac{\sigma}{E} = \frac{3Fgl}{4Eb^3h^2}$$

In order to avoid excessive current drain and reduced gauge lifetime from heating, it was decided to limit the DC bias voltage applied to the bridge to $V_0 = 12V$. Assuming that an amplifier with gain $G = 1000$ was to be used on the bridge output, values of thickness h were found for various materials which gave the design condition of $G\Delta V = 2.44$ millivolts for $F = 0.04N = 0.009$ pounds.

$$\Delta V = \frac{3FV_0Ggl}{4Eb^3h^2} = \frac{1530.9}{Eh^2} = .00244$$

Only standard thicknesses (1/8", 3/16", etc.) were considered. Each design was then checked to ensure that the stress was

below the yield stress of the material for $F = 260 \text{ N}$, and the natural frequency was calculated for a mass $m_c = 60 \text{ grams}$ at the center of the beam. The results are summarized in Table II.

TABLE II
CHARACTERISTICS OF TRANSDUCER BEAMS WITH CENTER MASS

Material	Yield Stress psi	E psi	ρ lb/in ³	Thickness in	Frequency Hz
Aluminum (7075-T6)	67,000	10.4×10^6	.101	.1875	630
				.250	953
Steel (C1020)	48,000	30.0×10^6	.272	.125	559
Magnesium (AZ31B)	24,000	6.5×10^6	.064	.250	773
Titanium (Alloy 16)	160,000	16.8×10^6	.164	.1875	775

Although magnesium and titanium beams both were highly desirable because of their natural frequency, it was impossible to obtain a supply of either material. The final design used 0.1875" thick 7075-T6 aluminum, this thickness being chosen rather than 0.25" despite its lower natural frequency to ensure adequate sensitivity. The construction of the transducer is shown in Figure 3. Four Micro-Measurements EA-13-125BB constantan foil strain gauges were used, one on each side of the beam at each end, as close as possible to the clamping points to maximize the strain they saw. The beam could withstand an applied force of 1500 N (350 pounds) before yielding.

After the beam was built, a 48-gram mass was placed in the 12-gram engine holder to give $m_c = 60$ grams, and the holder was tapped with a hammer. The output (after amplification) was observed on an oscilloscope to measure the natural frequency, which is approximately the frequency of the oscillatory response to this "impulse" input. This frequency was 526 Hz, or 83% of the predicted 630 Hz.

As a final test of the beam transducer, it was loaded with varying amounts of calibration weights (up to 18 pounds) and its amplified DC voltage output was measured for each weight. This output was exactly linearly related to the input force.

B. AMPLIFIER AND FILTER DESIGN

The output signal from the transducer strain gauge bridge was a very small voltage--on the order of microvolts normally--superimposed on a 6 VDC common-mode voltage. The analog-to-digital conversion system could handle inputs over the range of -5V to +5V, so to take full advantage of its accuracy considerable amplification of the differential output across the bridge was necessary. The literature on instrumentation-type DC amplifiers [Refs. 4-6] recommends that gains of greater than 1000 be avoided, if possible. Since maximum gain was desirable to permit a high transducer natural frequency, the design problem was reduced to finding the most cost-effective 1000-gain DC amplifier.

There are many figures of merit used to judge the performance of an instrumentation-type amplifier. In this relatively high-gain, low-frequency application, the important ones were the following:

- 1) input impedance (Z_{in})
- 2) input noise voltage
- 3) common-mode rejection ratio (CMRR)
- 4) input offset voltage (V_{os})
- 5) input offset current (i_{os})
- 6) input offset current and voltage thermal drift.

High input impedance was desired to minimize the current drawn by the amplifier from the transducer. This current could induce an error voltage in the resistance of the strain gauges. Input noise voltage is amplified by the gain of the amplifier and appears at the output, where no more than 0.6 millivolts (RMS) of noise could be tolerated. Assuming that the output was to be limited to a noise bandwidth of 250 Hz, then:

$$V_{n \text{ out}}(\text{RMS}) = \sqrt{\text{Gain} \cdot \text{Bandwidth} \cdot G_n} =$$

$$V_{n \cdot in} \sqrt{\text{Gain} \cdot \text{BW}}$$

$$0.0006 = V_{n \cdot in} \sqrt{1000 \cdot 250}$$

$$V_{n \text{ in}} \leq 1.2 \mu\text{V}/\sqrt{\text{Hz}}$$

Here G_n is input noise power spectral density. The maximum acceptable noise input voltage was 1.2 microvolts per $\sqrt{\text{Hz}}$.

Input offset voltage is the difference between inverting and non-inverting input voltages seen by the amplifier internally when both input terminals are grounded. When multiplied by the gain, it appears as an output DC voltage offset. In order to avoid large output offsets, it was desired that this input offset be no more than two millivolts. More important than the absolute value of this voltage was its variation with temperature. The lowest possible variation was desired, to avoid large drifts in output resulting from small short-term fluctuations in temperature inside the instrument circuit. Input offset current, when multiplied by the equivalent DC resistance of the circuit's inverting input and by the gain [Ref. 4], becomes an output offset voltage, so a minimum value and drift of this quantity was also desirable. As discussed in the calculations for noise input voltage, a gain-bandwidth product of at least 250,000 was required from the amplifier. Since an output voltage range of ten volts was needed, the slew rate S had to be [Ref. 4]:

$$S \geq 2\pi V_{out} F_{max} = .016 \text{ volt}/\mu\text{sec}$$

A high common-mode rejection ratio was desirable to minimize the effects on the amplifier of fluctuations in the DC offset of the transducer output.

Once the figures of merit for selection were established, manufacturers' data was consulted to find devices which met the performance requirements. Those which did so fell into

three categories: operational amplifiers with FET inputs; op amps with supergain bipolar transistor inputs; and monolithic instrumentation amplifiers. An amplifier system made from discrete op amps requires three of these devices, while monolithic amplifiers do the same job with a single (more expensive) chip.

Most of the acceptable op amps and amplifiers were extremely expensive and had long delivery times. On the basis of cost alone, the field was quickly narrowed to the National LM308A bipolar op amp (\$1.25) and the Analog Devices AD521J monolithic amplifier (\$13.). Performance figures for these two devices are summarized in Table III. The LM308A was selected because of its superior noise and offset voltage performance and because it gave a lower system parts cost, even though three chips plus ten resistors and capacitors were required for this approach compared to one chip and two resistors for the AD521J.

TABLE III
AMPLIFIER PERFORMANCE FOR GAIN OF 1000

Device	Z_{in} M Ω	Output Noise mV	Bandwidth Hz	V_{os} nV	V_{os} Drift μ V/ $^{\circ}$ C	CMRR dB
LM308A	40	.02	600	0.73	2.0	110
AD521J	3000	1.20	6000	2.00	7.0	110

Once the device to be used was selected, design of the amplifier circuit was straightforward. A standard op amp circuit was selected from Ref. 5 and the appropriate values of resistances were calculated to deliver the required gain. The circuit is shown in Figure 7, Appendix A. This design had the advantage that its input impedance was virtually infinite. Since the inverting and non-inverting inputs go to separate op amps, Z_{in} is twice the impedance from one input to system ground. The circuit amplifies only differential input voltages, $V_1 - V_2$, not any common-mode voltage. The gain equation is:

$$A_v = \frac{V_0}{V_1 - V_2} = - \left(1 + \frac{2R_1}{R}\right) \left(\frac{R_0}{R_2}\right)$$

It was known that a low-pass filter with DC gain of 2.57 would be used on the output, so the instrumentation amplifier gain needed was $1000/2.57 = 389$. Because of the limited selection of resistance values available, a theoretical gain of 402 had to be used in the final design. Metal film 1% resistors with relatively low values of resistance were used for noise minimization, accuracy, and thermal stability.

The amplifier circuit was followed by a four-pole, 200-Hz Butterworth low-pass filter, which provided anti-aliasing for the 500 Hz A/D conversion sampling and attenuation of higher-frequency noise. This filter was designed, using the techniques described in Ref. 7, as a cascaded pair

of two-pole active filters. The design DC gains of the two filter stages were 2.235 and 1.152. The filter was made using LM308 op amps, a less expensive version of the LM308A with higher voltage and current offsets. Carbon resistors were used. The precision components used in the instrumentation amplifier were not necessary here because of the low gain and high input voltage levels involved. Based on the measured values of the resistors actually used in this filter, its expected gain was 2.535, giving an overall DC gain for the amplifier system of 1019. The low-pass filter is also illustrated in Figure 7 of Appendix A.

Once the amplifier and filter were built, the system's gain and frequency response were measured by applying a variable frequency sinusoidal signal to the input through a 200:1 voltage divider and measuring the signal and output voltages with an AC voltmeter. An overall gain of 992 was measured, and the 6 dB rolloff point of the four-pole filter was 205 Hz. The strain gauge bridge was then connected through the shielded cable system, and a true-RMS voltmeter was used to measure the noise output voltage after the filter. This was 0.6 millivolt. All values were quite close to the desired performance.

C. TEMPERATURE TRANSDUCER DESIGN

The requirement of measuring the surface temperature of a rocket engine casing to an accuracy of 3°C over the range 25-250°C demanded a second transducer system. There are

only two reasonably simple and accurate techniques for converting a surface temperature to a voltage signal: resistance thermometers (thermistors) and thermocouples. The procedures for using both are highly developed and their errors are well understood. Reference 8 contains a great deal of data on surface thermometry and was used extensively in the development of this transducer.

Most types of thermocouples and thermistors have outputs (voltages or fractional change in resistance, respectively) which are not linearly related to temperature, except over fairly narrow ranges. This makes their implementation in an accurate, wide-range digital system fairly difficult; the conversion from transducer signal to displayed temperature requires some sort of table look-up or a nonlinear conversion equation. The primary selection criterion for this transducer was linearity of output over a wide temperature range. All types of thermistors and thermocouples require analog processing circuitry of roughly equal complexity, and their accuracy and response times depend more on their size and the technique of installation than on inherent properties, so these factors were not considered in the selection.

The data for numerous commercial thermistors and the standard voltage-vs.-junction temperature tables [Ref. 9] for the popular combinations of thermocouple metals were examined for linearity over the desired temperature range. Of these, the chromel-alumel thermocouple was by far the

most nearly linear. It produces an average output of $40.725 \mu\text{V}/^\circ\text{C}$ over the range $25\text{-}250^\circ\text{C}$, with a maximum deviation from linearity of $31.0 \mu\text{V}$ at 120°C . Thus, by assuming that its output and input were linearly related, a maximum error of less than one degree was introduced.

Since it was desired to make the A/D conversion of temperature nine bits with respect to a 10 V output range, the least significant bit (corresponding to one degree C) was $10/2^9 = 19.531$ millivolts. This meant that the thermocouple output signal had to be amplified by $19.531 \times 10^{-3} / 40.725 \times 10^{-6} = 479.6$ before conversion. An op amp amplifier system identical in design to that on the thrust transducer was used, with a two-pole low-pass Butterworth filter. Since temperature was to be sampled only one time, anti-aliasing was not a major concern; the principal purpose of the filter was noise reduction and a less expensive one with only two poles was acceptable. The DC gain of a two-pole Butterworth filter is 1.586, so the design gain of the amplifier was $479.6/1.586 = 302.4$.

The amplifier/filter circuit is shown in Appendix A, Figure 8. Once it was built, its gain was tested in the same manner as described for the thrust amplifier. A trimming potentiometer was placed in parallel with one of the filter resistors so that overall gain could be adjusted to achieve exactly the required value of 479.6. This was necessary to achieve the desired temperature accuracy. Once this was

done, the output noise after the filter was measured with the thermocouple and cabling connected, using a true-RMS voltmeter. The maximum acceptance noise was 4.8 millivolts; the measured value was 0.3 millivolts.

Thermocouples work on the principle that a junction between dissimilar metals produces a contact voltage which is related to temperature. This occurs at every such junction in a thermocouple system, not just at the sensor junction. Figure 4 shows the connection of a general thermocouple circuit. Each junction may be modeled as a voltage source whose value depends on the materials involved and on the temperature. If intermediate junctions J_3 and J_4 between the thermocouple wires and the copper conductors of the amplifier system are at the same temperature, then their voltages cancel. If reference junction J_2 is not explicitly provided, then since J_3 and J_4 are invisible a virtual reference junction between materials A and B will exist at the amplifier leads, at their unknown temperature.

The input to the amplifier is proportional to the difference in temperature between sensor and reference junctions, not to the sensor temperature alone. Clearly, the reference junction temperature must be known very accurately. This is usually achieved by immersing it in an ice bath, but this was considered impractical for a field-portable system. An alternate solution is to add a system to the circuit which changes output voltage with temperature at exactly the same rate as the reference junction, but in the opposite direction

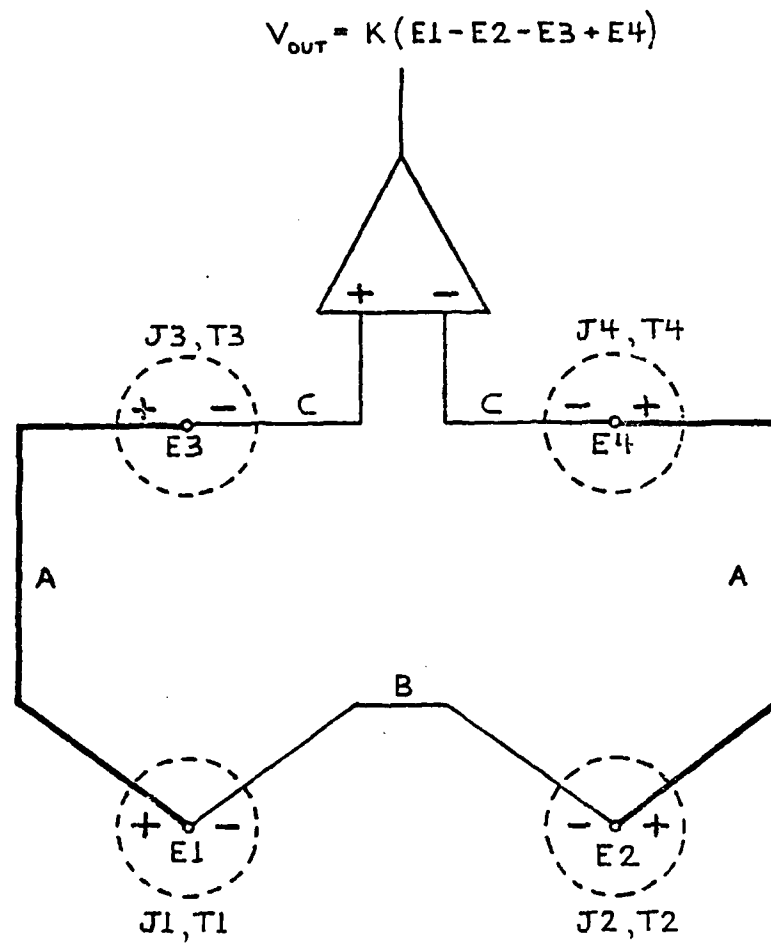


Figure 4. General thermocouple circuit characteristics.

so that the sum of the two voltages is a constant as ambient temperature varies. This technique was chosen, with the compensation being placed after the amplifier.

The compensation system used was a Yellow Springs Instrument Co. YSI 44202 precision thermistor network, whose output voltage was:

$$V_{out} = 0.805858 V_{in} - 0.0056846 V_{in} T_T$$

where T_T is the thermistor temperature. The amplified output of the thermocouple was:

$$V_{out} = 0.019531 T_S - 0.019531 T_R$$

where T_S is the sensor thermocouple temperature and T_R is the reference thermocouple temperature.

The thermistor network was connected to a $V_{in} = +5$ VDC power supply, and was combined with the thermocouple system output and with a DC source in an op amp summing network. By appropriate choice of the fixed scaling resistors in this summer, these three signals were made to produce the sum:

$$V_{out} = -0.01953 T_S + 0.01953 T_R - 0.01953 T_T$$

The reference thermocouple and the thermistor were then placed close together in a remote section of the circuit chassis so that $T_T = T_R$, leaving the output voltage directly proportional to the sensor thermocouple temperature. This circuit was checked by immersing the sensor in an ice bath (0°C) and adjusting a trimmer potentiometer on the fixed input to the summer until the output was -0.40 V. The sensor

was then immersed in boiling water (100°C). The output was -2.35 V, as predicted. The response was quite rapid.

A non-inverting negative peak detector circuit [described in Ref. 4] was placed at the output of the summer to catch and hold the most negative output value, corresponding to the highest temperature. After some experimentation, it was found that a 25 μ F holding capacitor was required in this circuit to minimize the voltage droop from leakage through the op amp circuit during a 75-second hold time. For most rocket engines, the maximum casing temperature is not reached until 60 to 70 seconds after the start of a test and in these normal cases voltage droop during the remaining 5 to 15 seconds of the holding period was negligible with this capacitor. The compensation system and peak detector are shown in Appendix A, Figure 9.

Following the guidelines in Ref. 8 for minimizing the error in surface temperature measurement, the thermocouple sensor was made from small-diameter (28 AWG) wire. The chromel and alumel wires were soldered close together and parallel for more than 100 wire diameters on a very thin piece of brass shim stock 0.25" square. This piece was formed to the shape of the engine casing surface with the wires paralleling what were estimated to be isotherms, and was covered with insulation before being taped tightly to the surface in a test. No measurements of error were conducted, but despite the care taken in the design an error

of two degrees was the minimum that was expected. This was in addition to the one degree error from A/D conversion.

IV. DIGITAL SYSTEM HARDWARE

A. INTERFACE CIRCUITRY

The signals developed by the thrust and temperature transducers and their associated amplifier, filter, and compensation systems were analog voltages with well-determined characteristics. The primary processing and decision-making done by this instrument was done with digital logic on the digital representation of these signals. A sample-and-hold circuit, a digital-to-analog converter, and a group of comparators, collectively referred to here as the interface circuitry, was used to link the analog and digital systems. Reference 10 was particularly helpful in designing this circuitry.

There were two independent transducer analog signals in this system that required analog-to-digital conversion. These conversions did not have to be simultaneous, and could have been done with two independent A/D converters, with one converter using multiplexed analog inputs, or with one converter using multiple inputs and outputs. The use of multiple A/D converters or even single monolithic A/D converters was rejected on the basis of parts cost. Multiplexing analog signals to the accuracy required by this system is quite difficult and was not considered desirable. Once the decision was made to use surplus microprocessor computing capacity

to perform A/D conversion with a D/A converter, it became possible to use the last method and put together a conversion system with multiple independent inputs and nonsimultaneous outputs.

The first step in designing the conversion system was selection of a D/A converter. Since only one converter was to be used for both transducers, the specifications were determined by the more demanding application, the thrust conversion. In this case, 12 bits of resolution with 11.7 bits of accuracy were required. This conversion could have taken as long as one millisecond or so; some part of the two-millisecond interval between conversions (500 Hz sampling rate) had to be left for the microprocessor to do other computation. With this amount of time available, the settling time of the D/A converter was not important. The accuracy requirement meant that the analog output of the converter could not deviate from a straight line between its minimum and maximum values by more than 30 percent of the least significant bit voltage, or 0.73 mV. This is called "0.3 LSB linearity."

The combination of 0.3 LSB linearity and operability from 12 V power supplies proved to be impossible to find among reasonably-priced 12-bit D/A converters. The linearity requirement was relaxed to 0.5 LSB, making one converter acceptable: the Burr-Brown DAC-80Z-CBI-V. This device had the additional advantages of being a monotonic converter

rather than a multiplying type, meaning that it did not require an external precision reference voltage, and of being a voltage rather than current output device, meaning that no external buffer op amps were required on its output. It used complementary binary input, with an all "0" digital input giving the most positive analog output and an all "1" input the most negative output. The converter could be connected to produce a variety of voltage output ranges; -5 V to + 5 V was the one used. Gain and offset triming potentiometers were required to obtain exactly the correct output voltage range. The settling time was three microseconds.

The A/D conversion technique employed was successive approximation. This is commonly implemented in hardware in fast A/D converter chips, but it is quite susceptible to slower software implementation as well. Basically, the microprocessor was used to generate a 12-bit test digital value which was then sent to the DAC. The resulting analog output from the DAC was sent to one input of an analog comparator while the signal voltage being converted was held at the other input. The microprocessor then sampled the output of the comparator and modified the test value according to whether the output was a digital "1" or a digital "0", i.e., whether the test value was greater than the signal or less. The program to do this is described in Chapter V. Twelve such successive tests are needed to perform a 12-bit conversion, requiring about 700 microseconds with the program used in this system.

Multiple independent outputs were obtained by sending the DAC output voltage in parallel to several comparators, with the second inputs to each comparator going to different analog signals. The microprocessor selected the output of one comparator for monitoring at the beginning of each conversion by sending the appropriate digital select code to a gated R-S flip-flop. The R output of this flip-flop caused the thrust comparator output to be transmitted through an AND-OR-INVERT circuit to the microprocessor input port, and blocked the temperature comparator output. The S output did the reverse.

The accuracy of the A/D conversion scheme was measured by applying DC voltages to a comparator input and recording the digital output of the system, then converting this digital value back to its analog equivalent. The difference between the actual voltage and its analog equivalent was the error. Over the full ± 5 V range of inputs, the standard deviation of error was 5 mV. This corresponds to 11 bits of actual delivered accuracy. While less than the desired 11.7 bits, this was considered still acceptable. The basic reason for this inaccuracy was the width of the comparators' switching transition.

The comparators were LM308 op amps operated open-loop and connected to ± 5 V power supplies. These devices have an open-loop gain of 3×10^5 . This means that their output voltage is greater than the difference between their inverting and non-inverting input voltages by this factor, up to

the limiting point where the output equals a power supply voltage. The output will thus swing over its full range of ten volts in response to a change of $10/3 \times 10^5 = 0.33$ mV in the differential input voltage. Since these comparators were connected to TTL-compatible circuitry, a feedback diode was added to limit the negative output to -0.3 V. The slope of this diode's forward-bias characteristic caused a broadening of the comparator's transition region between output levels. These comparators were connected with the non-inverting input grounded through a resistor and the inverting input connected to the two voltages being compared through identical resistors. This is equivalent to connecting each input to one of the voltages, and it eliminates the error effects of offset currents, as discussed in Ref. 4. The switching time of the comparator was about ten microseconds.

Successive approximation A/D conversion is extremely dependent on having the voltage under conversion held constant to within one or two LSB throughout the process. If this is not done, gross errors may result. This was not a problem with the slowly-changing peak detector output from the temperature transducer, but the thrust transducer voltage could change so rapidly that a sample-and-hold circuit was necessary between it and the A/D conversion system. What was required was a sample-and-hold having less than one LSB (2.44 mV) droop during the two-millisecond interval between thrust voltage samples, and having acquisition and settling times of a few microseconds or less. A Datel SHM-LM-2 was

available and met all of these requirements, so it was used. With a 1000 pF holding capacitor, this device had an acquisition time of six microseconds (to 0.01%), a settling time of 0.8 microseconds, and a droop of 0.1 mV in 2 milliseconds. A polystyrene holding capacitor was used to minimize errors due to dielectric absorption.

The interface circuitry is illustrated in Appendix A, Figure 10.

B. MICROPROCESSOR AND MEMORY SYSTEMS

The key elements of the digital portion of this system were the Motorola 6802 microprocessor and the MOS Technology 6530-002 input/output/timer chip. The remaining major elements of the digital system provided either memory or input/output support for these two devices.

The 6802 is an eight-bit microprocessor with an address space of 64K bytes (16 address lines), an on-chip RAM memory of 128 bytes, and an on-chip clock oscillator which permits clock rates of up to one megahertz, depending on the value of an external crystal. It has the same instruction set as the Motorola 6800 and is almost pin-for-pin compatible with it. It requires only a +5V power supply. The microprocessor and its supporting circuitry is shown in Appendix A, Figure 11.

Proper operation of the 6802 was found to depend on close adherence to the correct power-up sequence. The RESET input had to be held below 0.8 V for at least 20 milliseconds after

the chip's V_{cc} power supply went above 4.75 V, and then had to transition sharply and without oscillation to +5 V. This was accomplished with a comparator circuit identical to those described in the previous section of this chapter. A constant -2.5 V was obtained from a resistive voltage divider connected to the analog circuit - 5V supply and applied to one comparator input, while the second input was obtained from a series RC circuit connected between the digital +5V supply and ground. This circuit took 30 milliseconds to charge to +2.5 V after digital power was turned on, at which point the comparator output went high, setting the 6802 RESET.

The 6530 is a multipurpose microprocessor support chip. It has 64 bytes of RAM and 1024 bytes of mask-programmed ROM (neither of which were used in this system), and a programmable timer. It also has 16 input/output ports, arranged in two groups of eight called peripheral registers A and B (PA and PB). Each individual port of each register may be set up as a direct input or as a latched output by writing a "0" or a "1", respectively, to the corresponding bit of two registers called data direction registers A and B (PAD and PBD). Each of these four registers has a unique address derived from the chip's ten address lines and two chip select lines. Three of the I/O ports (PB5-7) were used as chip selects or as interrupt outputs, leaving 13 ports available for this system.

The 6530's timer is a series of eight registers located at adjacent addresses. The timer counts down for a number

of clock cycles equal to 1, 8, 64, or 1024 times the 8-bit value loaded in the appropriate register, and can generate an interrupt on the IRQ pin (PB7) when the count reaches zero. It was used in this system to provide a master event-timing reference by generating a non-maskable interrupt (NMI) to the microprocessor every two milliseconds. This 6530 IRQ pin was connected to the NMI input of the 6802 through a one-shot set to provide a five-microsecond low pulse. Considerable difficulty was experienced when interrupts were applied to the microprocessor directly rather than through a short-pulse one-shot. The 6802 would often execute the register-stacking operations of its built-in interrupt service routine, reach the first step of the software interrupt service program and then repeat the register stacking if the interrupt was still set. If the interrupt source was a type that remained set until cleared by a command in the service program, the 6802 remained locked in this cycle indefinitely.

The 6802 and 6530 are designed to drive a maximum of one TTL unit load on each pin. Those 6530 I/O pins which were used for output, and several of the 6802 address and control signal pins, had to be connected to larger loads than this. These pins were buffered with 74LS367 non-inverting buffers to increase their drive capacity to five unit loads.

The program memory for this system was quite simple. The 128 bytes of RAM which were built into the 6802 were used for stack and for program working storage. This RAM was located on "page zero" of the microprocessor's address space,

at addresses 0000-007F. The overall map for all of this system's memory, and for the various memory-mapped I/O and timer registers, is presented in Table IV. The operating program was stored in two 2708 UV-erasable programmable read-only memory (EPROM) chips, each having a capacity of 1024 8-bit bytes. These were wired to be at the top of the address space, at F800-FFFF. Their requirement for +12 V, +5V, and -5 V power supplies was not a problem since these voltages were already required by several other circuit elements. The memory circuit for this system is illustrated in Appendix A, Figure 12.

TABLE IV
SYSTEM MEMORY MAP

Address		Chip	Function
START	END		
0000	004F	6802	RAM; working memory
0050	007F	6802	RAM; stack
1340		6530	Peripheral register A; output
1341		6530	Data direction register A
1342		6530	Peripheral register B; input
1343		6530	Data direction register B
1344	134F	6530	Timer registers
13C0	13FF	6530	RAM; not used
2000	5FFF	2117	Dynamic RAM; data storage
F800	FBFF	2708	EPROM-1; program storage
FC00	FFFF	2708	EPROM-2; program storage

The original design of this instrument included a requirement that all of the thrust A/D conversion output data be stored in memory. This data was to be used in several ways. The first was to compute corrections to the thrust and total impulse data to account for the effects of engine propellant weight loss during a test. When the thrust transducer is placed flat, engine weight and thrust act in the same direction and are indistinguishable. The weight loss during a test can be as great as 0.6 Newton for large engines, and this can integrate to a substantial but predictable total impulse error. The error can be reduced by placing the transducer on its side, so that the weight and thrust vectors are perpendicular and the weight loss changes only a torsional force, to which the strain gauge bridge is insensitive. A more flexible approach, and the one which was desired, is to measure the amount of weight loss by comparing the transducer zero outputs before and after a test. Using this, a linearly-varying correction (zero for initial thrust, maximum for final thrust) can be applied to each stored data point and the total impulse and peak thrust can be recalculated from the corrected data.

Other plans for the stored thrust data included using it to generate a video output of thrust versus time on either an oscilloscope or a television-type display, or both. None of these uses for the data were implemented by the time that this report was written, although all were still under development. Storage of thrust data was not necessary for

performance of the other instrument functions described in this report.

Since this system was required to take 500 samples of thrust per second for up to 9.5 seconds, 4750 12-bit data points had to be stored. The reasons for using dynamic rather than static memory for this were discussed previously. The most economical dynamic RAM available was the Intel 2117 16K x 1 chip. Since the thrust data was already available in four-bit words as a result of the A/D conversion process, it was decided to use four of these chips to form a 16K x 4 memory. Accomplishing the required data storage used $3 \times 4750 = 14,250$ bytes of this.

Dynamic memory requires periodic refreshing to maintain its data. The 2117 chip is described in Ref. 11 as being arranged internally in a 128 x 128 array, and a refresh operation requires only that each of the 128 row addresses be written to every two milliseconds. The chip is a 16-pin package with only seven address input lines. These are multiplexed, serving as row address inputs during the first part of a read or write cycle and as column address inputs during the second part, giving effectively the 14 address lines required by a 16K memory. The refresh control and the address multiplexing was accomplished with a single dynamic RAM control chip, the Intel 3242. This took in 14 address lines, a clock, and a refresh enable line, and provided all the necessary address data to the 2117 chips for both refresh and normal read-write operations.

The 2117 required two timing inputs not provided by the 3242: row address strobe (RAS) and column address strobe (CAS). These enabled the chip to distinguish whether its address inputs were to be interpreted as row or column addresses. Both must appear with the proper time relationship during each read or write operation. They were derived from 6802 address and clock outputs, and effectively served as chip selects since they were gated so that they reached the memory chips only when the memory address was on the address bus. The 2117 has separate data inputs and outputs, and it was found that an extraneous signal appeared on the output in response to an address input regardless of the state of the $\overline{WE}$ read-write control. Consequently, the data input and output could not simply be connected directly to the system data bus; each had to be isolated by a tri-state buffer which was enabled by $\overline{WE}$ (for input) or by $\overline{WE}$ complemented (for output).

C. INPUT/OUTPUT CIRCUITRY

At an early stage of this system's design it was decided to make every effort to squeeze the required input and output (I/O) functions into the 13 ports available on a single 6530 chip. This decision was made to limit parts cost; it led to software complexity and debugging problems that more than offset the small savings in parts. The 13 I/O ports were allocated as follows:

PB0	input from comparators
PB1-PB5	input from 16-key keyboard
PA0-PA2	routing control for output data
PA3	sample-and-hold control
PA4-PA7	output data to DAC, LED, LCD

The input from the comparators, PB0, was simply wired to the output of the comparator AND-OR-INVERT circuit discussed in section A of this chapter. The PA3 output for sample-and-hold control was wired directly to the appropriate pin on the SHC chip. The remaining I/O functions require more explanation. The I/O circuit is shown in Appendix A, Figure 13. The IC numbers used hereafter refer to this drawing.

A 16-key unencoded, undebounced hex keyboard was used to select among the various operating modes and output data displays of this system. These keys were each SPST momentary switches, with one side of each tied to +5V and the other to one of the 16 inputs of a Harris HD-0165 keyboard encoder. Depressing a key pulled the corresponding encoder input high; the required pull-up resistors were built into the encoder. It also caused the encoder STROBE output to go low, setting a flip-flop (IC34) which was connected through a one-shot to the IRQ interrupt on the 6802 microprocessor. The 6802 looked at the keyboard output only in response to this interrupt, and cleared the interrupt flip-flop as part of its response. The four outputs of the encoder were held in latch IC35, whose enable circuitry allowed it to reject keyboard

outputs resulting from the simultaneous depression of two keys. This latch was reset only by the depression of a new keyboard key. Key debouncing was accomplished with a software wait loop.

Three single light-emitting diodes (LED) were used in this instrument to indicate its present status, and a four-digit liquid crystal display (LCD) was used to read out test data. The LED's were labeled TEST, CLEAR, and CALIBRATED, and their use is explained in the next chapter. They were driven by three of the seven outputs of a standard common-anode open-collector decoder-driver chip (IC32). The hex inputs to the driver required to achieve each desired combination of status lights are listed in Table V. These hex inputs were held for the decoder by a four-bit latch (IC36) which was reset only when a new sequence of status lights was commanded by the microprocessor.

The output data from the system's software was routed in four-bit words on I/O lines PA4-PA7 to eight possible destinations. These destinations were the four LCD digits, the 12-bit (three-word) input to the D/A converter, and a mixed destination which included a gated flip-flop for comparator selection and the LED decoder-driver. The selection of which of these destinations was to receive the data was made by output lines PA0-PA2, which were decoded into eight lines by a three-to-eight line decoder (IC40) and were then used as digit or chip selects or latch enables.

TABLE V
DECODING OF LED/COMPARATOR SELECT LINES

<u>HEX Inputs</u>	<u>Test</u>	<u>LED Selected</u>		<u>Comparator Selected</u>	
		<u>Clear</u>	<u>Calibrated</u>	<u>Thrust</u>	<u>Temperature</u>
1,3,7	off	on	off	off	off
2	on	on	off	off	off
4,9	off	on	on	off	off
5	off	off	on	off	off
6	on	off	on	off	off
8	on	on	on	off	off
A	on	off	off	off	off
B	off	off	off	off	off
C	off	on	on	off	on
D	off	off	on	on	off
E	on	off	on	off	on
F	off	off	off	on	off

A Schottky TTL decoder was used to ensure that the enable/select lines were stable before the data was sent to its destination. The data and select outputs left peripheral register A of the 6530 simultaneously, and the data lines were delayed by two 74LS367 non-inverting buffers to avoid a possible race condition. The decoding of lines PA0-PA2 is described in Table VI.

TABLE VI
DECODING OF DATA ROUTING CONTROL LINES

PA2	PA1	PA0	Data Destination
0	0	0	low word of D/A converter
0	0	1	middle word of DAC
0	1	0	high word of DAC
0	1	1	LED/comparator select
1	0	0	LCD digit 2
1	0	1	LCD digit 1 (LSD)
1	1	0	LCD digit 4 (MSD)
1	1	1	LCD digit 3

The liquid crystal display was a Timex T1001A reflective type with four 0.5" seven-segment characters. LCD's require excitation of each segment and a common backplane with a 30-100 Hz square wave alternating between 0 and 5 volts. A segment that is to be "on" is fed a signal 180° out of phase with the backplane excitation, so that the net field across it varies from +5V to -5V. An "off" segment is excited in phase with the backplane. A Siliconix DF411 four-digit LCD decoder-driver was used to provide the proper AC excitation to the display. This chip had four BCD data input lines, plus four digit-select inputs to select which of the four internal seven-segment output latches were to receive the decoded result of the input data.

The twelve bits of digital data required as an input by the D/A converter were sent to the converter in three

four-bit words because of the limited number of I/O ports available. These words were held by three four-bit latches (IC37-39), each of which was enabled separately by the appropriate output of the enable decoder. This arrangement saved using a second 6530 I/O chip to provide the 12 bits in parallel, but made the successive-approximation A/D. software somewhat more complex.

V. SYSTEM SOFTWARE

A. STRUCTURE AND GENERAL FEATURES

The software which this instrument used to perform its basic functions, not including video displays or weight loss corrections, required approximately 1450 bytes. It was organized into a main operating program, seven subroutines, two interrupt service routines, a power-up routine, and nine minor programs. Each of these was written in the 6800 mnemonic assembly language and was compiled and debugged separately on a Tektronix 8002 microprocessor development system (MDS). Reference 12 was used extensively in developing the software. A listing of the programs is provided at the end of this report.

Each of the 16 keys on the input keyboard commanded the instrument to take a particular action or display a particular piece of data, as shown in Table VII. Three of these actions were not yet implemented at the time this report was written. Of the remaining 13 keys, two called the main operating program (with different input data), one was a "stop" key, one called the "zero-memory" subroutine, and each of the other nine called a different minor program. In general, functions which were used in several different operating modes of the system, such as averaging, A/D conversion, and hex-to-BCD conversion, were implemented as subroutines. Only one of these was used directly as a response to a

key input. Key response was accomplished with individual programs, some of which did little more than set up data for a subroutine, then call it. These programs could actually be thought of as subroutines themselves, since each ended with an unconditional jump back to the keyboard input routine.

TABLE VII

FUNCTIONS OF INPUT KEYS
Programs Whose Names are in Parentheses not yet Written

Key	Program Called	Function
0	KEYIN	Stop program in progress and wait
1	OPER	Start an engine test, no temperature
2	OPER	Start test, measuring temperature
3	CALSET	Set up for calibration
4	CALIB	Accept calibration
5	DDTHST	Display one thrust A/D output
6	DDTEMP	Display one temperature A/D output
7	DTHST	Display peak thrust to 0.1N
8	DIMP	Display total impulse to 0.01N-sec
9	DBTIM	Display burn time to 0.01 sec
A	(DTV)	Display thrust curve on TV
B	DTEMP	Display peak temperature to 1°C
C	(APCOR)	Apply weight-loss correction
D	(DOSC)	Display thrust curve on oscilloscope
E	DDTIM	Display delay time to 0.01 sec
F	ZERO	Clear all RAM

When power is applied to a 6802 microprocessor and the proper timing is followed on its RESET pin, the chip goes to address locations FFFE and FFFF for its startup routine vector. This vector is simply the address of the first step of the power-up program routine. This routine, entitled PWRUP in this system, must start with a CLI (clear interrupt flag) command and should set the initial value of the stack pointer. In this system, it also temporarily disabled timer interrupts, set the data direction registers (PAD and PBD) of the 6530 for input or output as appropriate, and cleared all static and dynamic RAM. Dynamic memory requires an initial clearing or refresh before it can be used after power is first applied to it. The PWRUP routine turned on the CLEAR LED as its last step (leaving the other LED's off), indicating that the RAM was clear of all test data. The system then entered a loop where it waited for a keyboard input.

Depressing any key on the instrument keyboard caused an IRQ interrupt to be sent to the 6802. In response, the 6802 went to addresses FFF8 - FFF9 and found the vector for the IRQ service routine, which was entitled IRQRES. This routine began with a one-millisecond wait loop to give the key switch "ringing" time to damp out. It then stored the four 6530 input bits PB1 - PB5 which contained the output from the keyboard encoder in memory location 0046. If the key input was the stop command (key 0) or the zero-memory command (key F), the routine forced an immediate return to the KEYIN

keyboard response routine to do this and terminated any other action that the instrument had been doing. Otherwise, IRQRES simply returned the system to whatever it was doing before the interrupt, and the new mode action was not taken until the next time the system returned normally to KEYIN.

The KEYIN routine used an action pointer table technique [Ref. 12] to develop responses to the key input commands. This routine compared the contents of address 0046 (filled by IRQRES) to those of address 0049, the last command executed by the system. If they were the same, it did nothing. This meant that in order to repeat a particular key action, some other key (generally the "stop" key, key 0) had to be used in between. If the two memory location contents were different, indicating that an unexecuted command was pending, then the key command (multiplied by 3) was used as an offset for an indexed-address jump into an action pointer table entitled VECTOR, which started at address FC00. This table contained unconditional jump instructions, directing the system to the appropriate programs for response to each key. For example, if key 5 was pressed, the system jumped to address $FC00 + (3 \times 5) = FC0F$, where it found the command JMP FBA0 telling it to jump unconditionally to FBA0, the starting address of the program responding to that key.

The accuracy of the time-based measurements made by this instrument (burn and delay times and total impulse) depended on having an accurate timing reference available to trigger an A/D conversion exactly every two milliseconds.

The easiest way to do this was to use the 6530 programmable timer to generate a non-maskable interrupt to the 6802 at this interval. The NMI interrupt sent the 6802 to addresses FFFC-FFFD, where it found the vector sending it to the response routine for this interrupt, NMIRES. This routine began by reloading the counter so that another interrupt would be generated in 2.00 milliseconds. It then enabled the RAM refresh input to the 3242 dynamic RAM controller for 128 microseconds, giving it time to automatically generate all the signals necessary for a refresh cycle. If a mode was being executed which required A/D conversion (keys 1 through 6), the program jumped to the ADC12 A/D conversion subroutine before returning from the interrupt. Otherwise it returned from interrupt directly at this point.

Because this system had only 48 bytes of RAM available for stack and these bytes were located just above important program memory, considerable care was taken to ensure that the stack did not overflow. The 6802 uses seven bytes of stack in responding to an interrupt and three bytes to jump to a subroutine, so this limited the depth to which subroutines could be nested, particularly in the interrupt service routines. Several early versions of this system's software overflowed the stack. Such overflows were not possible with the final design.

Although many sections of this system's software were debugged individually on the Tektronix MDS, several of the major interactions between them were not. These interactions

were the ones which depended on the occurrence of NMI timer interrupts and real-time thrust inputs. MDS emulation provides line-by-line traceouts of the status of every CPU register as each program instruction is executed, but to provide this and other valuable debugging services it must slow the execution speed by a large factor. Interrupt inputs to the MDS are enabled for only a small fraction of each instruction execution cycle, so the MDS seldom sees the negative-going edge it requires to sense an NMI interrupt. This meant that the real-time NMI-dependent processes of the system could not be emulated. A vast amount of time was required to debug these subroutines and interrupt interactions using a Paratronics 532 Logic State Analyzer. This device permitted real-time operation of the system with its own CPU in place, rather than the emulator plug required by the MDS, but it provided access only to the contents of the external pins of the CPU (address and data bus) and not to internal registers. It allowed no interaction for modifying the contents of memory, and could provide only 256 clock cycles of traceout at once.

B. SOFTWARE A/D CONVERSION

The hardware required to accomplish a 12-bit successive-approximation A/D conversion using software and a D/A converter has been described previously. The function of the software was to generate the proper 12 bits of data to send to the converter, working with one four-bit word at a time

while leaving the other eight bits unchanged. The software also generated control signals for the sample-and-hold circuit and packed the A/D conversion output from the last four bits of three separate bytes into one and a half consecutive bytes.

All of the A/D software was contained in one subroutine entitled ADC12, which occupied 131 bytes and required 715 microseconds to perform a conversion. This program operated in straight binary, converting the digital test values to the complementary binary required by the DAC before sending them out, then changing the final result back to the straight binary used in the rest of the system. ADC12 could be called repeatedly only by the NMIRES interrupt response program, so the timer NMI interrupt had to be enabled by any program which required more than one A/D conversion. These programs generally used a WAI (wait for interrupt) command where the A/D data was required so that their execution was stopped at this point until the data was available from ADC12 after the next timer interrupt.

The ADC12 subroutine began by sending a "sample" command to the SHC for 14 microseconds, then a "hold" command. Next it set the initial digital signal to the DAC to 01111..., corresponding to a 0.00V analog output since the DAC operated over a -5V to +5V output range. It then entered a loop where it worked on only one four-bit word (starting with the most significant four bits), leaving the other two words set at

their initial values. Straight binary will be used hereafter to describe the functioning of this program.

The ADC12 program set the initial value of the current four-bit test word to 1000, and set a rotating-bit word to 0100. It sent the initial test word to the appropriate latched DAC input by adding the necessary final three bits (PA2 - PA0) to enable the proper latch. After 45 microseconds of executing other instructions, it examined the output of whichever comparator had been enabled by the program which had called ADC12. An output of "1" indicated that the DAC analog output was less than the signal being converted. In this case, the rotating bit was ORed to the test word, increasing its analog equivalent voltage. If the comparator output was "0", the rotating bit was subtracted from the test word instead. In either case, the rotating bit was then shifted one position to the right and another conversion cycle was started. After every fourth cycle, a new word was begun.

The branches possible in a four-bit straight binary successive-approximation A/D conversion are illustrated in Figure 5.

C. MAIN OPERATING PROGRAM

The main operating program of this instrument, entitled OPER, was the program used during an engine static test. Its function was to recognize the occurrence of such events as start of thrust, thrust termination, and gas-generation

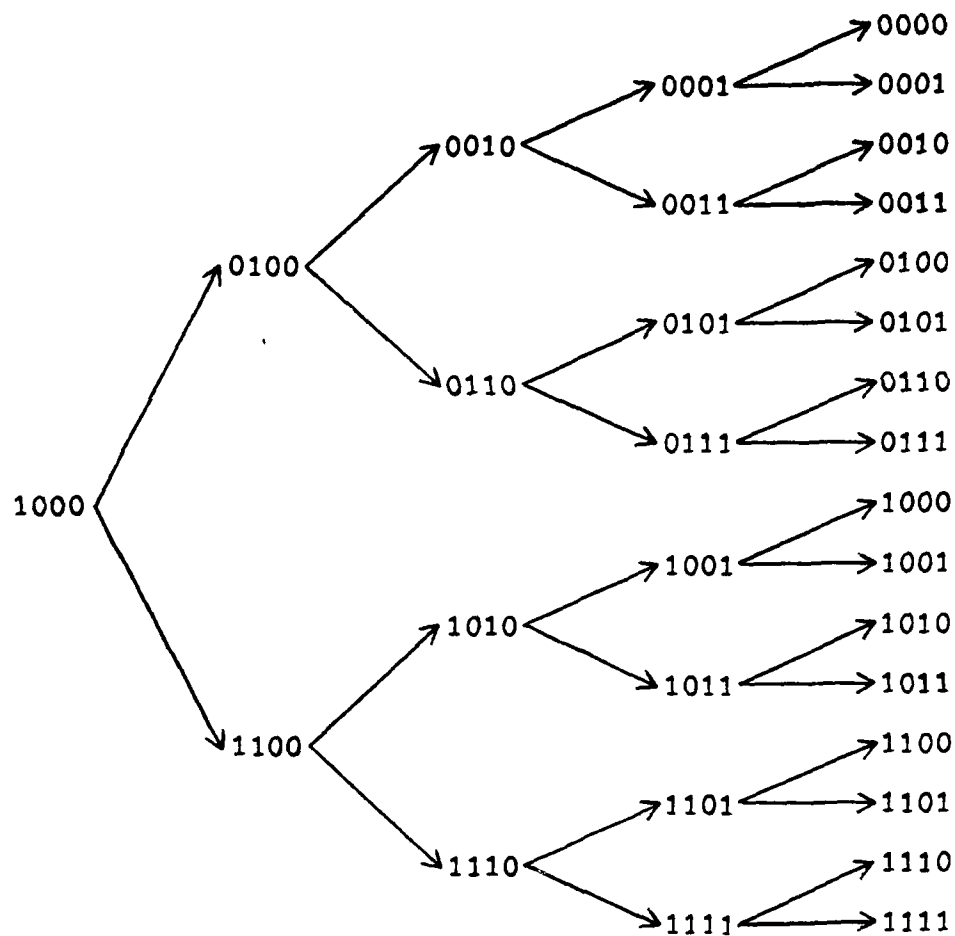


Figure 5. Four-bit straight-binary successive approximation A/D conversion. Upper branches taken when comparator output is 0.

charge actuation regardless of the shape of the input thrust variation. It computed the value of burn and delay times and peak thrust and added thrust outputs to accumulate total impulse, all based strictly upon the input values it received every two milliseconds from the ADC12 subroutine. Depending on whether it was called by depressing input key 1 or key 2, it either ignored the temperature peak detector or accepted its output once at the end of a test, respectively. This program was 365 bytes in length. Its operating time varied widely during the course of a single engine test because of its numerous conditional branching instructions, but it was always short enough that its operation (including an A/D conversion) was completed in the two-millisecond interval between NMI timer interrupts.

The flow diagram for OPER is shown in Figure 6. The first action of the program was to check memory location 004E to see if a calibration had been performed since the system had been turned on. A calibration, which used the CALSET and CALIB routines described in the next section of this chapter, caused a flag value to be placed in this memory location. Only if a calibration had been performed would the program permit initiation of a test. It indicated that a test was in progress by turning off the CLEAR LED and turning on the TEST LED. It then enabled the thrust comparator for the A/D conversion, and averaged the result of eight conversions to obtain a value of thrust transducer output corresponding to zero thrust input. This feature

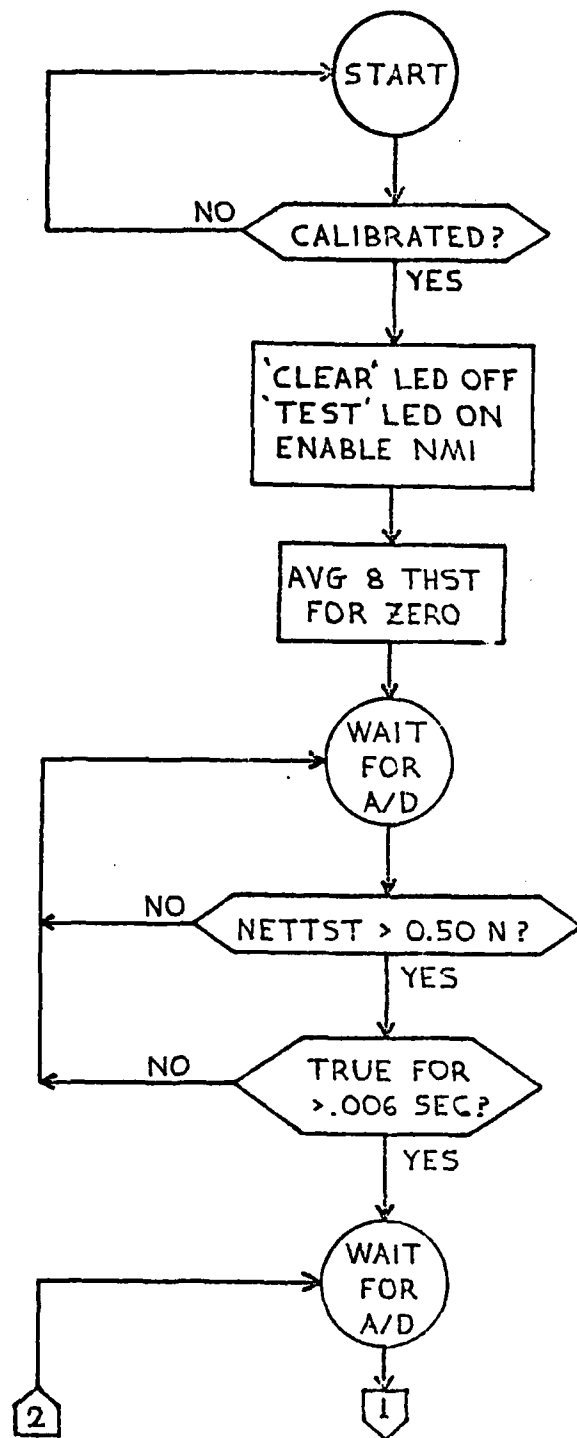


Figure 6. Flowchart of main operating program.

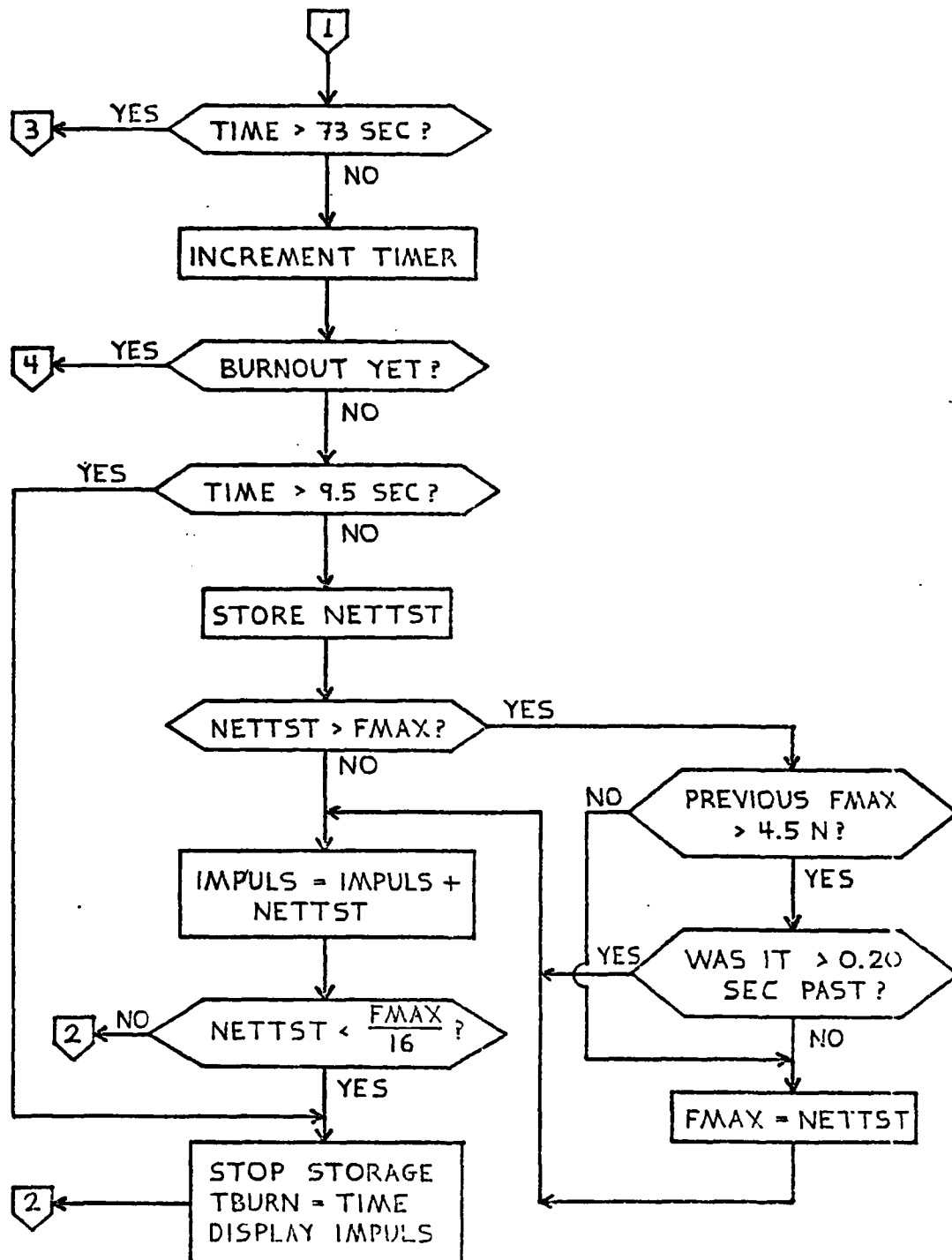


Figure 6. (Continued)

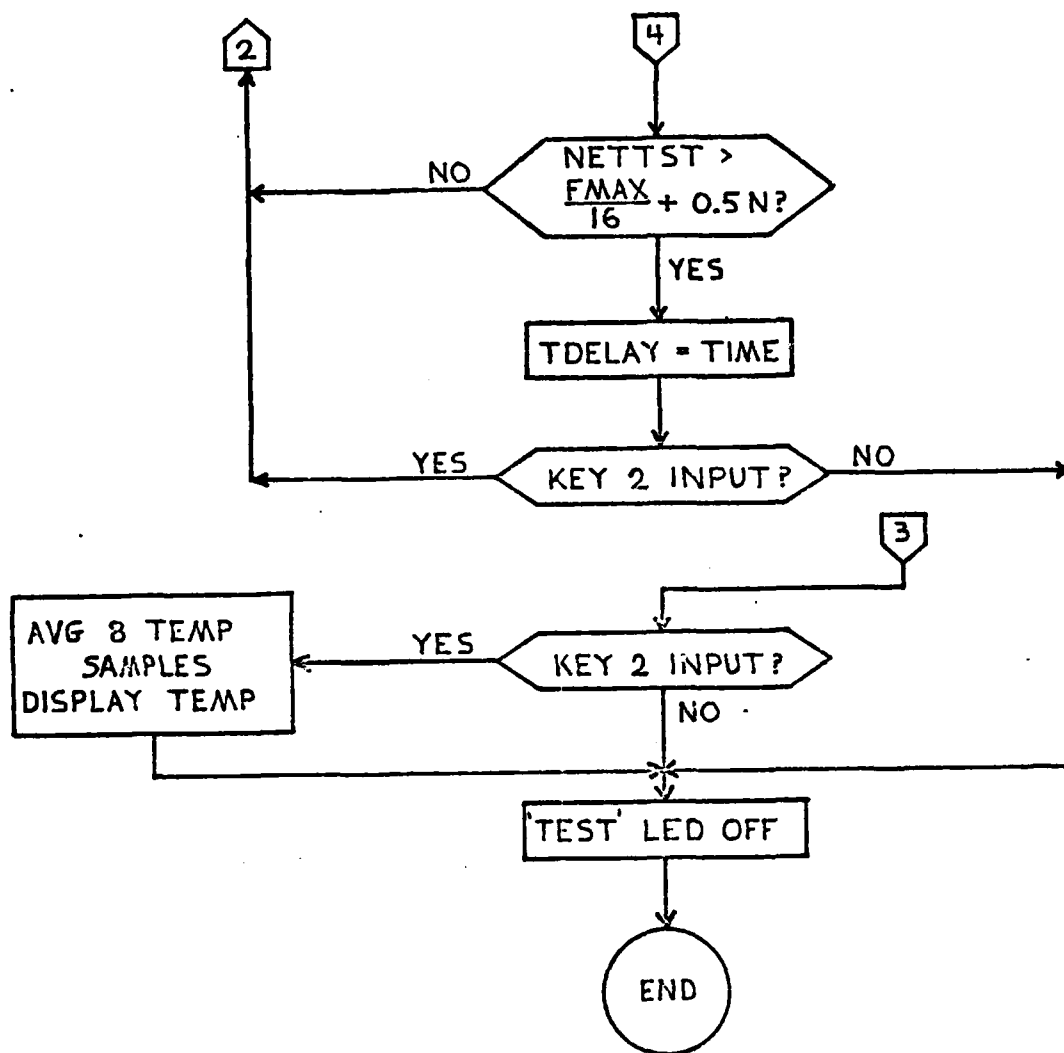


Figure 6. (Continued)

eliminated the need to carefully trim the transducer analog output to -5.000 V before each test, since throughout the rest of the program this zero value was subtracted from the A/D output to obtain the actual net thrust.

After the preliminary preparations for a test were completed, the OPER routine waited for a thrust input of 0.50 N or greater lasting at least three timer periods (0.006 seconds) before it decided that a valid test had begun. Setting a threshold in this manner prevented a single large noise spike before engine ignition from erroneously causing the program to begin accumulating durations and total impulse. The time at which this threshold was satisfied was declared by the program to be time zero for computing durations, and for the next 9.50 seconds or until it decided engine burnout had occurred (whichever came first) it accumulated thrust outputs for total impulse and examined them to detect a valid peak value.

The peak value detection logic rejected any peaks which occurred more than 0.20 seconds after a previous peak of 4.5 Newtons or greater, but accepted all others. "Peaks" rejected by this criterion were almost always just spurious output spikes, caused by ejection of an engine casing by its gas generation charge when there was no delay between burnout and actuation of this charge. The most recent valid value of peak thrust was placed in addresses 002C and 002D during each cycle of OPER until burnout, at which time the updating was stopped.

During each cycle of the program between the times when it recognized ignition and burnout, the net thrust was added to a three-byte memory location where total impulse was accumulated. The design limit of this system was 99.99 N-sec. of total impulse. Since 99.99 Newtons was 3150 A/D counts and one second was 500 A/D cycles, the maximum value which this accumulation could reach was $500 \times 3150 = 1.575 \times 10^6$, or 21 bits. The net thrust was obtained by sending the ADC12 direct output to the NETTST subroutine, where the zero reference value was subtracted.

The burnout detection logic defined burnout as that moment when the thrust output of an engine first dropped below 1/16 of the peak value recorded by that engine. This is a more flexible criterion than the fixed threshold value used to define ignition. At the moment when burnout was recognized, the total impulse value accumulated was sent to the DIMP subprogram to be converted to a value in N-sec. and then to be displayed. The value of the A/D conversion cycle counter was also sent to a memory location to be remembered as the burn time.

After burnout was sensed, the program waited for the occurrence of a thrust spike at least 0.50 N greater than the threshold used to determine the burnout. This was presumed to have come from the activation of the gas-generation charge on the engine, and its time of occurrence minus the burn time of the engine was stored as the delay time. At this point, if the without-temperature mode of OPER was the

one in progress, the test was considered to be terminated and the TEST LED was turned off.

If the with-temperature mode was in progress, or if no post-burnout thrust spike occurred, OPER continued until 73 seconds after engine ignition before terminating. Temperature data, if required, was taken just before termination by averaging eight samples of the output of the peak detector circuit. This data was then converted to degrees Centigrade and displayed on the LCD by the DTEMP subprogram.

After each test the peak thrust, burn time, delay time, total impulse, and (if used) peak temperature were available for display on the LCD by pressing the appropriate input key. The instrument could then be prepared for a new test by using key F to clear all RAM locations except those five where the calibration data was stored. Doing this caused the CLEAR LED to come back on and left the CALIBRATED LED on.

D. SUBROUTINES AND MINOR PROGRAMS

Up to this point only the two principal programs used by this system, plus the interrupt response and power-up routines and a few subroutines, have been described. The remaining subroutines and minor subprograms were either quite straightforward or very short and will be discussed only briefly.

The CALSET subprogram was entered in response to key 3. It enabled the NMI interrupt in order to gain access to the A/D subroutine, then averaged eight A/D outputs using the

AVG8 subroutine. This result was used as the zero-thrust reference value in the CALIB subprogram.

The CALIB subprogram was entered in response to key 4, and was meant to be preceded by CALSET. Before selecting this routine, but after executing CALSET, a 10.00 Newton calibration weight was placed on the thrust transducer. CALIB then averaged eight A/D outputs using AVG8 and subtracted from this result the zero-thrust reference value developed in CALSET. The difference was the net output for 10.00 N of thrust/weight. This calibration value was stored in a section of RAM that was not erased between static tests, and was used to convert A/D outputs from the operating program into units of thrust or impulse. Upon completion of CALIB, the CALIBRATED LED was turned on.

The HEXBCD subroutine accepted two bytes of hexadecimal data as an input and, using the binary-to-BCD conversion algorithm developed in Ref. 12, converted this data to a four-digit BCD result. The total impulse display program (DIMP) could conceivably generate a hex quantity which would overflow four BCD digits, so HEXBCD included a provision to generate an output of 9999 if too large a hex input were provided. It put the BCD output in the high four bits of four data words, then added the proper data routing control bits to the end of each word to send it to the appropriate LCD digit.

The DTHST subprogram was entered in response to key 7. It took the peak thrust A/D hex value developed by the

operating program during a static test and multiplied it by 100 with the MULT subroutine, then divided it by the calibration value with the DVID subroutine. The result was the hex value of peak thrust in tenths of Newtons. This was then displayed using HEXBCD.

The DTEMP subprogram was entered in response to key B. It divided the 12-bit A/D count resulting from conversion of the peak detector output during a test by approximately 8. This made each bit of what remained equal to one degree C., so the quotient was converted by HEXBCD and displayed.

The DIMP subprogram was entered in response to key 8. It started by doubling the hex value accumulated by the operating program in the three-byte total impulse memory. This converted the value to units of bit-milliseconds. This was divided by the calibration, which had units of bits per 10 N, putting the quotient in units of 0.01 N-sec. This was sent to HEXBCD for conversion and display.

The DBTIM (key 9) and DDTIM (key E) subprograms were combined since their function was virtually identical. They converted the burn time and delay time, respectively, that had been accumulated by the operating program into units of 0.01 second and sent them to HEXBCD. This conversion was accomplished by dividing five into the respective counts of NMI interrupts, which had been incremented by OPER every 0.002 second.

The DDTHST (key 5) and DDTEMP (key 6) subprograms were also so similar that they were combined. They provided an

LCD output of the direct, unconverted result of one A/D conversion of the thrust and temperature transducer outputs, respectively. These programs simply enabled the appropriate comparator, called the ADC12 subroutine once, then sent the result to HEXBCD.

The MULT and DVID subroutines provided 16 x 24-bit integer multiplication and 24/24-bit integer division, respectively, for use throughout the program. Both used standard algorithms taken from Ref. 12. Considerable care was taken to set up the arithmetic operations required by this system so that they could be performed in integer arithmetic rather than floating-point.

VI. CONCLUSION

The objective of this project was to develop a microprocessor-based instrument for accurate static-test measurement of five performance parameters of small solid-fuel rocket engines. Maximum and minimum design values for these parameters and maximum allowable errors were established to define the performance required from the system. It was then designed and built with the additional constraints that it have minimum parts cost and that it must operate from two 12-volt batteries and draw no more than one ampere from either. The final design drew 0.69 ampere from +12V and 0.18 ampere from -12V, and had a 1979 parts cost of about \$350.

In the process of building this instrument and debugging its 1450 bytes of software, it became apparent that major simplifications of the software could have been achieved by using slightly more expensive and sophisticated parts. Nevertheless, the system was eventually made to operate exactly as desired, displaying the values of the five parameters one at a time on a four-digit liquid crystal display after each static test.

A wide variety of solid-fuel model rocket motors were static tested to verify the instrument's performance. Every feature of the system functioned as designed on every test.

Static calibrations proved that the system thrust and temperature transducers and its A/D conversion routine delivered satisfactory static accuracy. There is no such thing as a precision rocket engine which could be used as a realistic dynamic reference source, so the real-time accuracy could only be checked approximately. This was done by recording the analog transducer output signals on a chart recorder during a static test and measuring them to estimate the values of the five parameters. These values were in excellent agreement with the values then presented by the system on its LCD.

The instrument developed in this project made extensive use of the arithmetic and decision-making capabilities of a microprocessor to deliver accurate measurements with relatively inexpensive hardware requiring minimal adjustment by the operator. These measurements were accurate regardless of the shape of the input signal as long as this signal was within the design limits of the system.

This instrument may be modified to test engines of up to 1000 Newtons of peak thrust and 1000 Newton-seconds of total impulse by replacing resistor R in the thrust amplifier circuit (Figure 7) by a 2000 ohm resistor and by replacing the 10.0 N calibration weight with a 100 N weight. In this case, the thrust data output will be in whole Newtons and the impulse data in tenths of Newton-seconds. This will degrade the accuracy for tests of small engines.

APPENDIX A. CIRCUIT SCHEMATICS

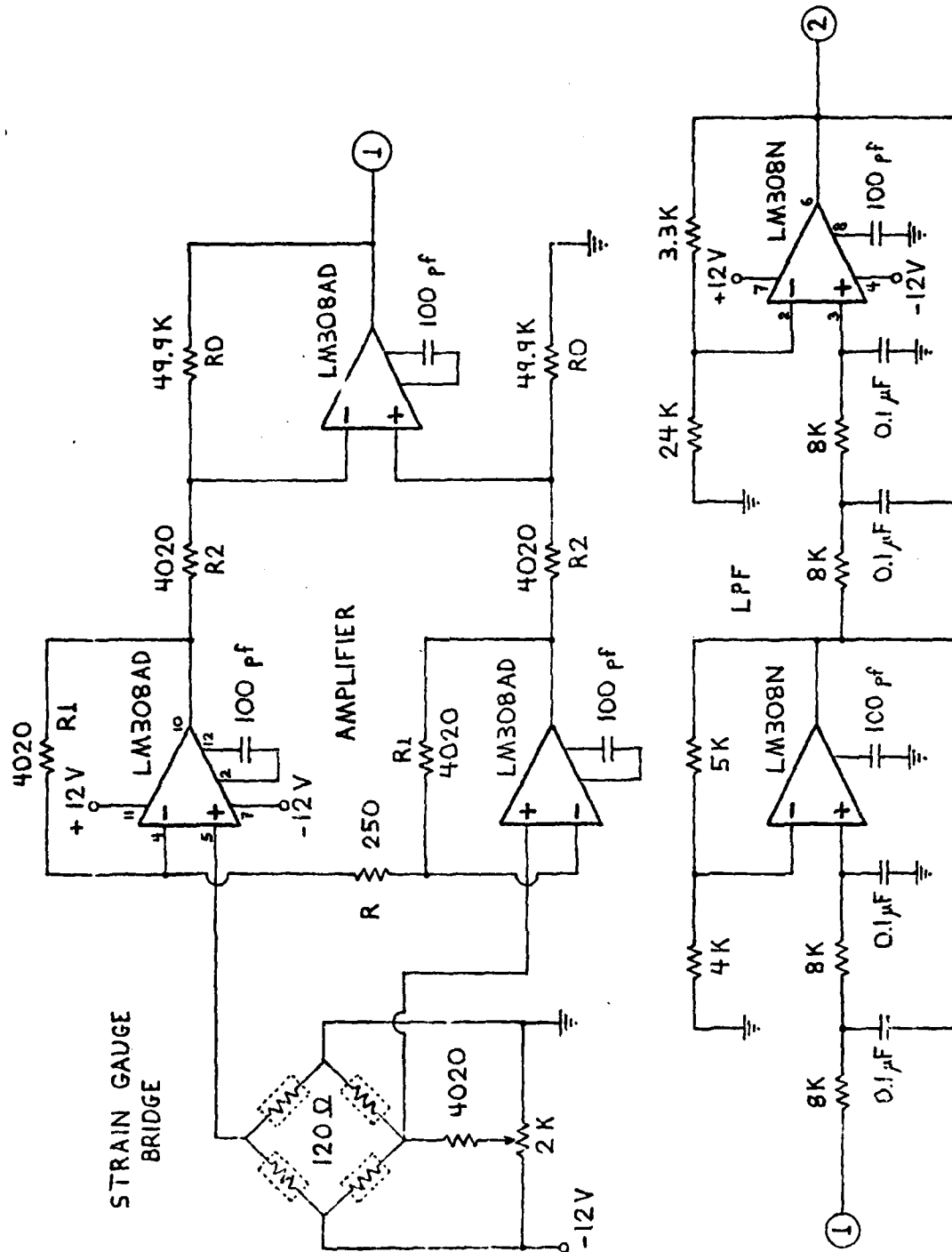


Figure 7. Thrust transducer amplifier and filter circuit.

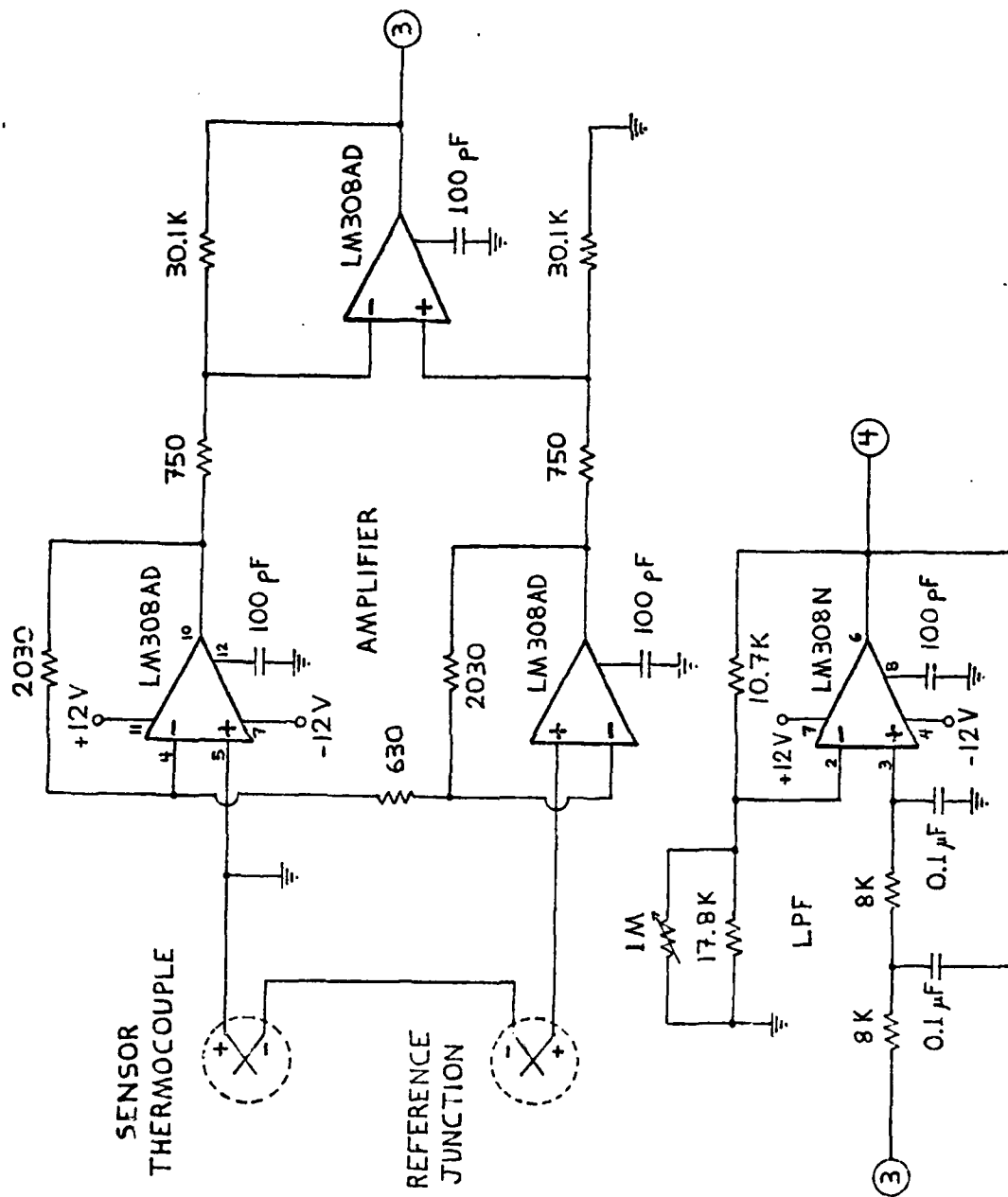


FIG. 1. Thermocouple amplifier and filter circuit.

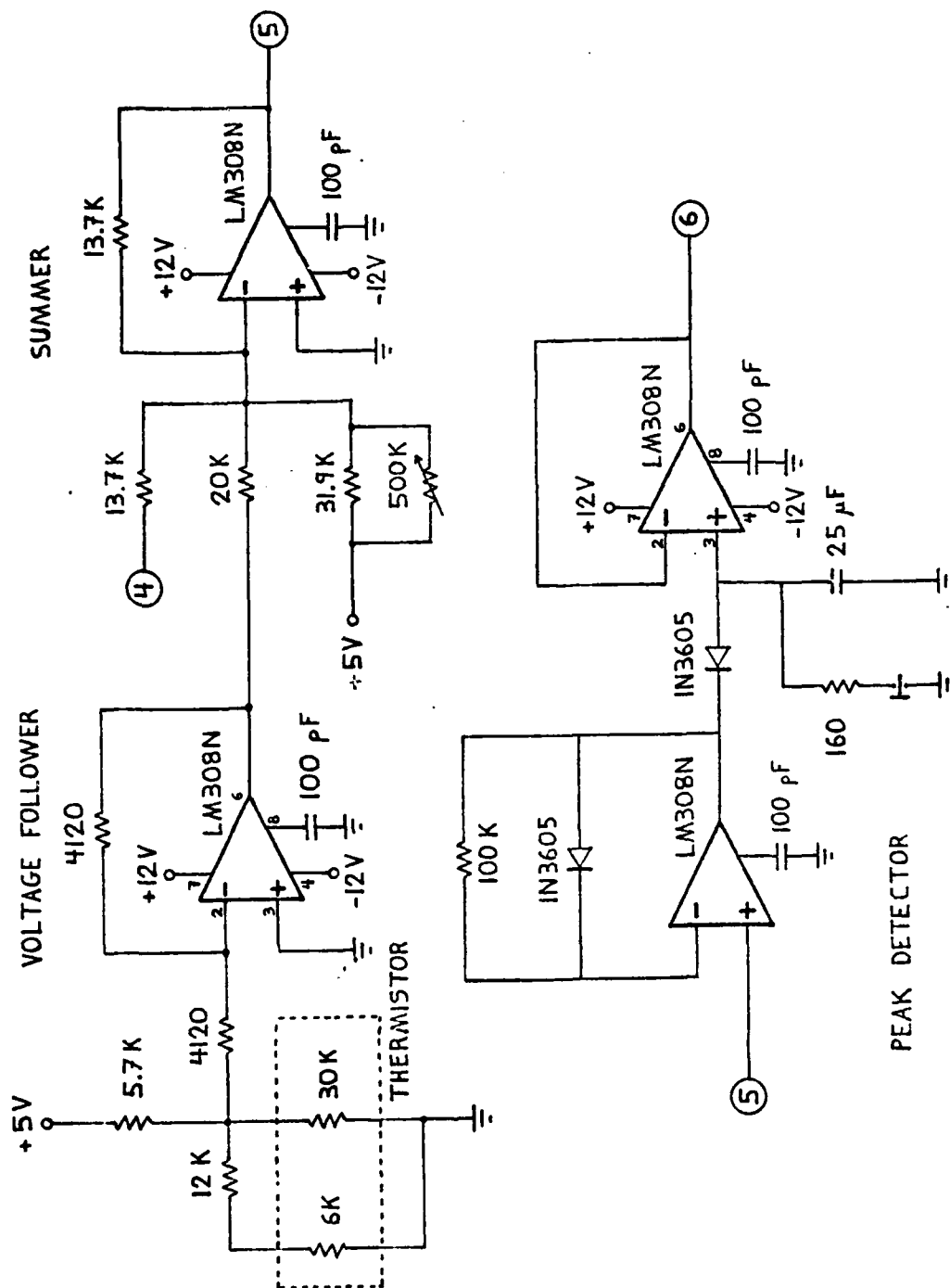


Figure 9. Thermocouple compensation and peak detection circuit.

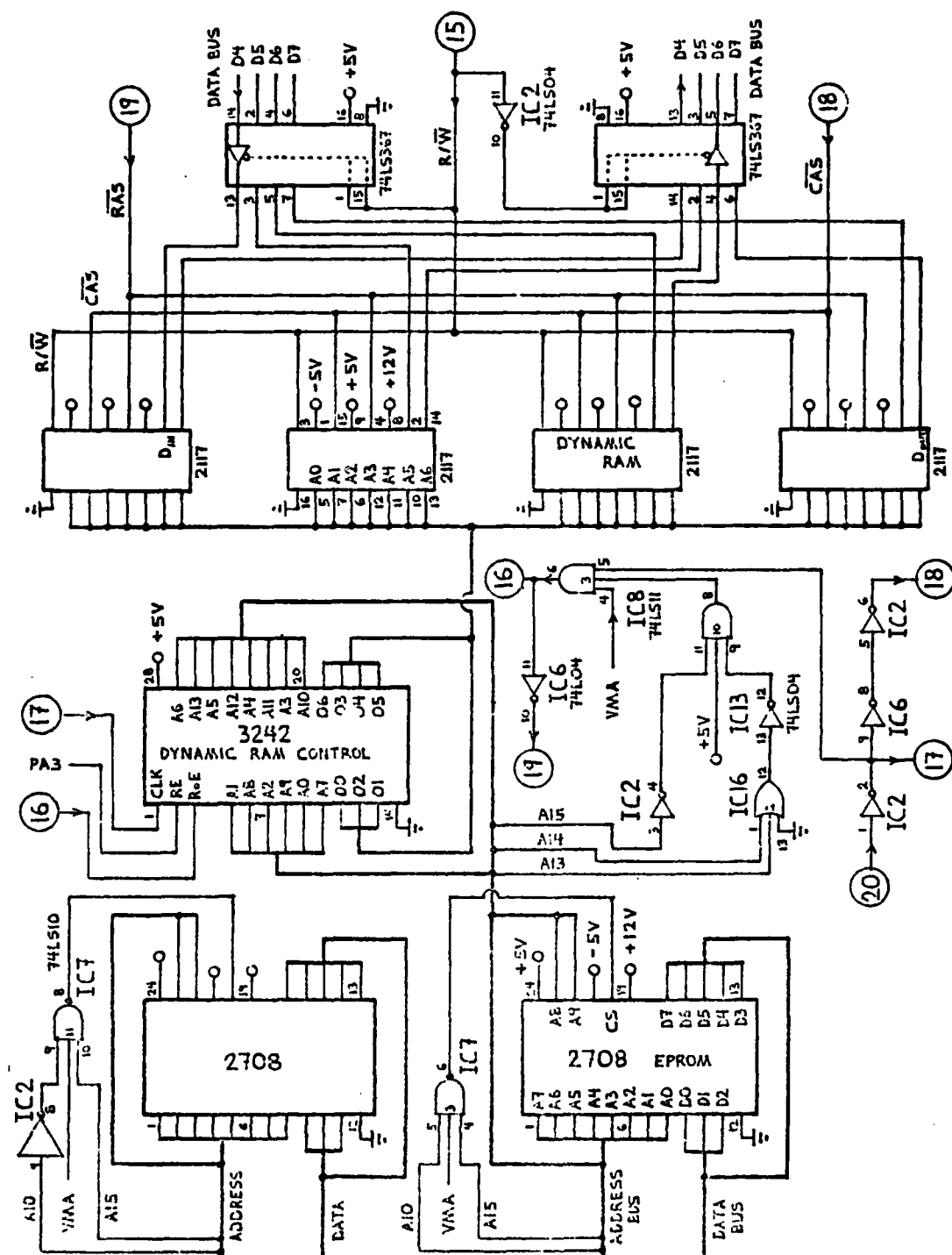


Figure 12. Memory system.

APPENDIX B. INSTRUMENT OPERATING INSTRUCTIONS

1. Apply external +12V and -12V power to the system.
2. Turn on the switch to provide internal +5V power to the digital circuits. The CLEAR LED should come on and the LCD should read 0000.
3. Trim the BRIDGE NULL potentiometer while monitoring thrust transducer output voltage until this voltage is slightly above -5.00 V.
4. Place the temperature sensor in an ice bath and trim the TEMP NULL potentiometer while monitoring temperature transducer output voltage until this voltage is -0.400 V.
5. Place the thrust transducer flat and press key 3 to set up for calibration.
6. With the thrust transducer still flat, place the 10.0 N calibration weight in the engine holder, wait a few seconds, and press key 4. The CALIBRATED LED should come on.
7. Place the thrust transducer on its side and clamp it securely to an extremely rigid and sturdy support.
8. Place the engine to be tested in the engine holder. Ensure that the ignition leads exert no force on the engine along its thrust axis. Zero the peak detector circuit. Attach temperature sensor to the engine.

9. Press key 1 or key 2 to initiate a test. The TEST LED should come on and the CLEAR LED should go out. Press key 0 if it is necessary to cancel a test before the engine is fired.
10. Fire the engine. The measured total impulse should appear on the LCD at burnout.
11. When the TEST LED goes out, indicating the end of the test, read out the performance parameters of interest by pressing keys 7, 8, 9, B, and/or E.
12. Clear the memory to set up for a new test by pressing key F. The CLEAR LED should come on and the CALIBRATED LED should remain on. Return to step 8 to conduct a new test.

COMPUTER PROGRAM

FI00	>	ORG	3F100H	
FI03 CE0003		LDX	#03	
FI03 8C02		LDA A	#02	;SET MUX WORD CONTROL
FI05 971C		STA A	1CH	
FD07 86F8		LDA A	#0F8H	
FI09 B71340		STA A	PAI	;SEND "SAMPLE" TO SHC
FI0C 86F9		LDA A	#0F8H	
FI0E B71340		STA A	PAI	;SET LOW 11 IAC BITS TO '1'
FD11 8672		LDA A	#072H	
FI13 B71340		STA A	PAI	;SEND "HOLI" TO SHC
FI16 8C40	WNEX	LDA A	#40H	;BEGIN NEW WORD A/D
FI18 971C		STA A	11H	;SET ROTATING BIT
FI1A 8C05		LDA A	#05	
FD1C 971B		STA A	1BH	;SET BIT COUNTER
FI1E 8600		LDA A	#00H	;SET IAC INITIAL VALUE
FD20 2026		BRA	BAKER	
FI22 961B	START	LDA A	1EH	;BEGIN NEW BIT A/D
FD24 8001		SUB A	#01	;DECREMENT BIT COUNTER
FI26 971B		STA A	1EH	
FD28 272B		BEQ	NEXT	;GO TO NEXT WORD AFTER 4 BITS
FI2A 01		NOP		
FI2B B61342		LDA A	PBI	
FD2E B61342		LDA A	PBD	
FI31 01		NOP		
FI32 01		NOP		
FD33 8401		AND A	#01	;EXAMINE COMPARATOR
FI35 260A		BNE	PAST	;BRANCH IF SAMPLE > IAC
FD37 961E		LDA A	1EH	
FI39 1611		LDA B	11H	
FI3B 12		SBA		;SUBTRACT ROT BIT FROM TEST
FD3C 74001D		LSR	1EH	;SHIFT ROTATING BIT
FI3F 2007		BRA	BAKER	
FD41 961E	PAST	LDA A	1EH	
FI43 9A11		ORA A	11H	;OR ROT BIT TO TEST
FD45 74001D		LSR	11H	;SHIFT ROTATING BIT
FI48 84F0	BAKER	AND A	#0F0H	;MASK OFF LAST 4 BITS
FD4A 971E		STA A	1EH	
FI4C 86F0		EOR A	#0F0H	;COMPLEMENT OUTPUT TO IAC
FI4E 9A1C		ORA A	1CH	;ADD MUX CONTROL
FI50 B71340		STA A	PAI	;SEND TO IAC
FI53 20C1		BRA	START	
FD55 961E	NEXT	LDA A	1EH	
FI57 A71F		STA A	1FH.X	;STORE ADC VALUE UNPACKED
FD59 7A001C		DEC	1CH	;ANTI UNCOMPLEMENTED
FI5C 09		DEX		
FI5L 2EB7		BGT	WNEX	;GO TO NEXT WORD
FI5F 9620		LDA A	20H	;PACK ADC VALUE IN 2 BYTES
FI61 CE0004		LDX	#04	
FI64 44	PACK	LSR A		
FI65 09		DEX		
FD66 26FC		BNE	PACK	
FI68 9A21		ORA A	21H	
FI6A 9724		STA A	24H	;LOW ADC BYTE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE.
THE COPY FURNISHED TO YOU CONTAINED A
SIGNIFICANT NUMBER OF PAGES WHICH DO NOT
REPRODUCE LEGIBLY.

FL6C 9622		LLA A	22H	
FD6E C30004		LIX	#04	
FL71 44	PACK2	LSR A		
FL72 09		DEX		
FD73 26FC		BNE	PACK2	
FL75 9725		STA A	25H	;HIGH A/D BYTE
FD77 86FC		LDA A	#0FCH	
FL79 9024		SUB A	24H	
FD7B 9724		STA A	24H	
FL7D 8607		LEA A	#0FH	
FL7F 9225		SBC A	25H	
FL81 9725		STA A	25H	;CHANGE SIGN OF RESULT
FL83 39		RTS		
FC37		OR3	0FC37H	
FC37 8609	AVG8	LDA A	#09	;AVERAGES 8 A/D OUTPUTS
FC39 9716		STA A	16H	
FC3B 8600		LDA A	#00	
FC3D 9714		STA A	14H	
FC3F 9715		STA A	15H	
FC41 9616	AVG	LEA A	16H	
FC43 8001		SUB A	#1	
FC45 9716		STA A	16H	
FC47 2F12		BLE	RUN1	
FC49 01	WAIT	NOP		;WAIT FOR A/D OUTPUT
FC4A 01		NOP		
FC4B 3E		WAI		
FC4C 9614		LEA A	14H	
FC4E D624		LDA B	24H	
FC50 1B		ABA		
FC51 9714		STA A	14H	
FC53 9615		LDA A	15H	
FC55 9925		AIC A	25H	
FC57 9715		STA A	15H	;ALL 8 A/D OUTPUTS
FC59 2036		BRA	AVG	
FC5B C30003	RUN1	LIX	#03	
FC5E 740015	RUN	LSR	15H	
FC61 760014		ROR	14H	
FC64 09		DEX		;DIVIDE SUM BY 8 FOR AVG
FC65 26F7		BNE	RUN	
FC67 39		RTS		
1340	PAI	EQU	1340H	
1342	PBI	EQU	1342H	
		ENE		

THIS DOCUMENT IS NOT QUALITY PRACTICABLE.
 THE INFORMATION CONTAINED HEREIN IS UNCLASSIFIED
 EXCEPT WHERE SHOWN OTHERWISE WHICH DO NOT

FE00	FE00	>	ORG	0FE00H	
FE00 9C4E			LDA A	4EH	
FE02 910F			CMP A	#2FH	
FE04 26FA			BNE	OPER	;DO NOT PROCEED IF UNCALIBRATED
FE06 3F			SWI		
FE07 8602			LTA A	#2	
FE09 9726			STA A	2EH	;SET INTERVAL COUNTER
FE0B 8620			LDA A	#20H	
FE0D 9729			STA A	29H	;SET MEMORY POINTER
FE0F 66D3			LDA A	#0D3H	
FE11 B71340			STA A	PAL	;ENABLE THRUST COMPARATOR
FE14 8663			LDA A	#63H	
FE16 F71340			STA A	PAL	;SET UP LED
FE19 9742			STA A	42H	
FE1B B1FC37			JSR	AVG8	;AVG PRESENT OUTPUT FOR ZERO REF
FE1E 3E			WAI		;WAIT FOR TIMER INTERRUPT
FE1F B1F628	STRT		JSR	NETTST	;SUBTRACT ZERO REF FROM A/D
FE22 9C46			LTA A	4EH	
FE24 260F			BNE	STRT1	;BRANCH IF TEST STARTED
FE26 9617			LDA A	17H	
FE28 8012			SUB A	#19	
FE2A 9616			LDA A	19H	
FE2C 8200			SBC A	#0	
FE2E 21EE			BLT	STRT	;WAIT FOR THRUST > 0.50N
FE30 7A0026			IEC	2EH	
FE33 2CE9			BGE	STRT	;WAIT FOR TEST HI FOR 3 COUNTS
FE35 8601	STRT1		LTA A	#1	
FE37 9748			STA A	4EH	
FE39 9B27			ADD A	27H	
FE3B 9727			STA A	27H	;TIME COUNTER LOW BYTE
FE3D 9628			LDA A	2EH	
FE3F 8900			ADC A	#0	
FE41 9728			STA A	2EH	;TIME COUNTER HIGH BYTE
FE43 8187			CMP A	#0FH	
FE45 2607			BNE	TEST1	
FE47 8602			LTA A	#02	
FE49 973C			STA A	3EH	
FE4B 7EFF2C	>		JMP	DELAY	;STOP TIMING AFTER 73 SEC
FE4E 962B	TEST1		LTA A	2EH	
FE50 2703			BEQ	TEST	
FE52 7EFF17	>		JMP	TEST3	;BYPASS IF BURNOUT PAST
FE55 9627	TEST		LTA A	27H	
FE57 808E			SUB A	#0EH	
FE59 9626			LTA A	2EH	
FE5B 8212			SBC A	#12H	
FE5D 2D04			BLT	BURN	;STOP STORING TEST AFTER 9.52 SEC
FE5F 8601			LTA A	#01	
FE61 972B			STA A	2EH	
FE63 962B	BURN		LTA A	2EH	
FE65 2703			BEQ	STORE	
FE67 7EFFE5	>		JMP	TEST2	;BYPASS IF BURNOUT PAST
FE6A 8603	STORE		LTA A	#3	
FE6C 9B2A			ADL A	2AH	;MEMORY POINTER LOW
FE6E 972A			STA A	2AH	
FE70 9629			LDA A	2EH	
FE72 8E00			ADC A	#0	;MEMORY POINTER HIGH
FE74 9729			STA A	2EH	;INCREMENT MEMORY POINTER BY 3
FE76 1E29			LTX	2EH	;PUT MEMORY POINTER IN X
FE78 9620	STOR1		LTA A	20H	
FE7A A700			STA A	X	
FE7C 09			DEX		
FE7E 9621			LTA A	21H	
FE7F A700			STA A	X	

FE81 09		DEX	
FE82 9622		LDA A	22H
FE84 A700		STA A	X
FE86 9617	FMAX	LIA A	17H
FE88 9021		SUB A	21H ;SUBTRACT FMAX FROM THRUST
FE8A 9618		LDA A	18H
FE8C 922C		SEC A	2CH
FE8E 2132		BLT	IMPULS ;BRANCH IF FMAX > THRUST
FE90 9621		LIA A	21H
FE92 C087		SUB B	#135 ;WAS PREVIOUS FMAX > 4.5N?
FE94 D62C		LDA B	2CH
FE96 C200		SEC B	#00
FE98 2C12		BGE	TIMT ;BRANCH IF YES
FE9A 9617	NEWMAX	LIA A	17H
FE9C 9721		STA A	21H
FE9E 9618		LDA A	18H
FEA0 972C		STA A	2CH ;SET FMAX = THRUST
FEA2 9627		LDA A	27H
FEA4 972E		STA A	2EH
FEA6 9628		LDA A	28H
FEA8 972F		STA A	2FH ;REMEMBER TIME OF FMAX
FEAA 2016		BRA	IMPULS
FEAC 9627	TIMT	LIA A	27H
FEAE 902E		SUB A	2EH
FEB0 9730		STA A	30H
FEB2 9628		LDA A	28H
FEB4 922F		SBC A	2FH
FEB6 9731		STA A	31H ;FIND TIME-TMAX
FEB8 9630		LDA A	30H
FEBA 8064		SUB A	#100
FEBC 9631		LDA A	31H
FEBE 8200		SEC A	#00 ;LAST FMAX MORE THAN 2.2 SEC FAST?
FECE 2116		BLT	NEWMAX ;BRANCH IF NO
FECE 2116	IMPULS	LIA A	32H ;ADD THST TO TOTAL IMPULSE
FECE 2116		ADD A	17H
FECE 2116		STA A	32H
FECE 2116		LIA A	33H
FECE 2116		ADC A	18H
FECE 2116		STA A	33H
FECE 2116		LIA A	34H
FECE 2116		ADC A	#00
FECE 2116		STA A	34H ;TOTAL IMPULSE HIGH BYTE
FECE 2116		LIA A	2CH ;DIVIDE FMAX BY 16
FECE 2116		LIA B	21H
FECE 2116		LIX	#04
FECE 2116	DIVBY	LSR A	
FECE 2116		ROR B	
FECE 2116		DEX	
FECE 2116		BNE	DIVBY
FECE 2116		ADC B	#00
FECE 2116		STA B	35H ;LSD OF FMAX/16
FECE 2116		ADC A	#00
FECE 2116		STA A	36H ;MSD OF FMAX/16
FECE 2116		LDA A	35H
FECE 2116		SUB B	17H
FECE 2116		LIA B	36H
FECE 2116		SBC B	18H ;FMAX/16 - THST
FECE 2116		BGE	TEST2 ;BRANCH IF THST < FMAX/16
FECE 2116		JMP	STRT
FECE 2116	TEST2	LIA A	2BH
FECE 2116		BNE	TEST3 ;BRANCH IF BURNOUT PAST
FECE 2116		LDA A	27H
FECE 2116		SUB A	#100

THIS DOCUMENT IS BEST QUALITY PRACTICABLE.
 THERE ARE SOME UNRECORDED CHANGES WHICH DO NOT
 AFFECT THE RESULTS OF THE TESTS.

FFFD 9626		LDA A	28H	
FFFF 8200		SBC A	#00	;IS TIME > 0.20 SEC?
FF01 2C03		BGE	TBURN	;BRANCH IF YES
FF03 7EFE1E	>	JMP	STRT	;WAIT FOR NEXT A/D OUTPUT
FF06 8601	TBURN	LDA A	#01	
FF08 972B		STA A	21H	
FF0A 9627		LIA A	27H	
FF0C 9737		STA A	37H	;BURN TIME LOW BYTE
FF0E 9626		LDA A	28H	
FF10 9738		STA A	38H	;BURN TIME HIGH BYTE
FF12 7EFE08		JMP	LIMP	
FF15 2012		BRA	GOBCK	
FF17 9635	TEST3	LDA A	35H	
FF19 8B0F		ADC A	#15	
FF1B 9636		LDA A	36H	
FF1D 8900		ADC A	#00	
FF1F 9635		LIA A	35H	
FF21 9017		SUB A	17H	
FF23 9636		LDA A	36H	
FF25 9218		SBC A	18H	;IF MAX/16) - 0.5N - THST
FF27 2103		BLT	DELAY	;BRANCH IF EJECTION OCCURS
FF29 7EFE1E	>	JMP	STRT	
FF2C 9639	GOBCK	LIA A	39H	
FF2E 2610	DELAY	BNE	TEMP1	;BRANCH IF EJECTION PAST
FF30 8601		LDA A	#01	
FF32 9739		STA A	39H	
FF34 9627		LIA A	27H	
FF36 9637		SUB A	37H	
FF38 973A		STA A	3AH	;DELAY TIME (LO)
FF3A 9628		LIA A	28H	
FF3C 9238		SEC A	38H	
FF3E 973B		STA A	3BH	;DELAY TIME (HI)
FF40 9649	TEMP1	LIA A	49H	
FF42 8101		CMP A	#01	
FF44 2602		ENE	TEMP	;BRANCH UNLESS IN NO-TEMP MODE
FF46 201E		BRA	FINIS	
FF48 9639	TEMP	LDA A	39H	
FF4A 8102		CMP A	#02	
FF4C 2702		BEQ	TEMP2	;BRANCH IF TIME > 73 SEC
FF4E 2019		BRA	GOBCK	
FF50 86C3	TEMP2	LDA A	#0C3H	
FF52 B71340		STA A	PAD	;ENABLE PK LET COMPARATOR
FF55 8600		LIA A	#00	
FF57 9714		STA A	14H	
FF59 9715		STA A	15H	
FF5B B1FC37		JSR	AVG8	
FF5E 9614		LDA A	14H	
FF60 9717		STA A	17H	;STORE PEAK TEMP (LO)
FF62 9615		LIA A	15H	
FF64 9723		STA A	23H	;STORE PEAK TEMP (HI)
FF66 8653	FINIS	LIA A	#53H	
FF68 B71340		STA A	PAD	;TURN OFF TESTING LED
FF6B 9742		STA A	42H	
FF6D 7EFB65		JMP	KEYIN	
FB08	LIMP	EQU	0FB08H	
FA40	TEMP	EQU	0FA40H	
FC37	AVG8	EQU	0FC37H	
F828	NETTST	EQU	0F828H	
1340	PAD	EQU	1340H	
FB65	KEYIN	EQU	0FB65H	
		END		

NOT

AD-A081 083

NAVAL POSTGRADUATE SCHOOL MONTEREY CA
DEVELOPMENT OF A MICROPROCESSOR-BASED INSTRUMENT FOR STATIC TES--ETC(U)
SEP 79 A H BARBER

F/G 14/2

UNCLASSIFIED

2 of 2
AD
ADN-083



NL
END
DATE
FILMED
3-80
DDC

FAB0		ORG	0FAB0H
FAB0 CE01F4	> IRGRES	LDI	#500
FAB3 01	WAIT	NOP	
FAB4 09		LEX	
FAB5 26FC		BNE	WAIT ;DEFOUNCE KEYBOARD
FAB7 86F3		LIA A	#CF3H
FAB9 B71340		STA A	;RESET IRQ FLIP-FLOP
FABC 9642		LIA A	42H
FABE B71340		STA A	;RESTORE LED OUTPUT
FAC1 B61342		LIA A	;INPUT KEYBOARD ENTRY
FAC4 841E		AND A	#1EH ;MASK OFF KEY BITS
FAC6 44		LSR A	
FAC7 9746		STA A	4CH
FAC9 9646		LDA A	4EH
FACB 2706		PEQ	TAKIN ;IF NOT STOP OR ZERO
FACI 810F		CMP A	#CFH ;COMMANDS, RETURN TO
FACF 2702		BEQ	TAKIN ;PROGRAM
FAD1 2021		ERA	BACK
FAD3 9642	TAKIN	LDA A	42H
FAL5 81E3		CMP A	#C3H ;IS TEST LED ON?
FAD7 2607		BNE	TAK
FAL9 8653		LIA A	#53H ;IF YES, TURN OFF
FALB B71340		STA A	
FADE 9742		STA A	42H
FAE0 32	TAK	PUL A	;IF STOP OR ZERO KEY
FAE1 32		PUL A	;WAS PRESSED, RETURN
FAE2 32		PUL A	;TO KEYIN ONLY
FAE3 32		PUL A	
FAE4 32		PUL A	
FAE5 32		PUL A	
FAE6 32		PUL A	
FAE7 8665		LIA A	#65H
FAE9 36		PSH A	
FAEA 86FB		LDA A	#CFBH
FAEC 36		PSH A	
FAED 8E00		LEA A	#C0
FAEF 36		PSH A	
FAF0 36		PSH A	
FAF1 36		PSH A	
FAF2 36		PSH A	
FAF3 36		PSH A	
FAF4 3B	BACK	RTI	
1340	PAD	EQU	1340H
FF70	HEXBCE	EQU	0FF70H
1342	PEC	EQU	1342H
FC68	> NMIRE5	ORG	0FC68H
FC68 86F8		LIA A	#24E
FC6A B71340		STA A	134EH ;LOAD TIMER FOR 2 MSEC
FC6C 9642		LIA A	42H
FC6F 8A08		ORA A	#8
FC71 B71340		STA A	PAD ;ENABLE RAM REFRESH
FC74 CE000F		LIX	#15
FC77 09	WAIT	LEX	
FC7E 26F1		BNE	WAIT ;WAIT FOR RAM REFRESH
FC7A 9642		LIA A	42H
FC7C 94F7		AND A	#CF7H
FC7E B71340		STA A	PAD ;STOP RAM REFRESH
FC81 9649		LIA A	49H ;WHAT MOLE WAS SELECTED
FC83 E007		SUB A	#7 ;BY KEYBOARD?
FC85 2E03		BGT	BACK
FC87 BCF100		JSR	ADC12 ;GO TO A/D IF REQUIRED
FC8A 3E	BACK	RTI	
1340	PAC	EQU	1340H
FD00	ADC12	EQU	0FD00H
		END	

FA40	8600	ORG	07A40H	
FA42	9701	LIA A	#00	;DISPLAY TEMP IN DEG C
FA44	9702	STA A	01H	
FA46	970A	STA A	0EH	;SET UP FOR DIVISION
FA48	865B	STA A	0AH	
FA4A	901F	LDA A	#5BH	
FA4C	970C	SUB A	1FH	
FA4E	8607	STA A	0CH	;CHANGE SIGN OF ADC
FA50	9223	LDA A	#07	
FA52	970B	SBC A	23H	
FA54	8617	STA A	0BH	
FA56	9707	LIA A	#31	
FA58	CE0002	STA A	0FH	;31 TO DIVISOR
FA5B	950C	LDX	#02	
FA5D	950C	QUAI LDA A	0CE	;QUAIRUPLE OUTPUT
FA5F	970C	ALL A	0CH	
FA61	960B	STA A	0CE	
FA63	990B	LDA A	0BH	
FA65	970B	ADC A	0BE	
FA67	960A	STA A	0BH	
FA69	990A	LDA A	0AH	
FA6B	970A	ADC A	0AH	
FA6D	09	STA A	0AE	
FA6E	20EB	LDX		
FA70	B1F184	BNE	QUAI	
FA73	960B	JSR	LVII	;CONVERT TEMP TO WHOLE DEGREES
FA75	9741	LDA A	0EH	;BY DIVIDING AND COUNT
FA77	960C	STA A	41H	;BY 7.75
FA79	9740	LDA A	0CH	
FA7B	B1FF70	STA A	40H	
FA7D	7EFB65	JSR	HEXB0D	;DISPLAY PEAK TEMPERATURE
FA80	9621	JMP	KEYIN	
FA82	9706	ORG	0FA00H	
FA84	962C	LIA A	21H	;PEAK THRUST IN 2.1N
FA86	9720	STA A	0EH	;FMAX (LC) TO MPCAND
FA88	8664	LDA A	2CH	
FA8A	9704	STA A	0EH	;FMAX (HI) TO MPCAND
FA8C	8600	LIA A	#120	
FA8E	9702	STA A	04H	;MULTIPLIER = 100
FA90	9703	LDA A	#00	
FA92	9701	STA A	02H	
FA94	B1F1CF	STA A	03H	
FA96	9602	STA A	0DE	
FA98	970A	JSR	MULT	;MULTIPLY FMAX BY 100
FA9A	9603	LDA A	0EH	
FA9C	9701	STA A	0AE	;PRODUCT TO DIVIDEND HI
FA9E	9601	LDA A	03E	
FAA0	970C	STA A	0FH	;DIVIDEND MEI
FAA2	9601	LIA A	04H	
FAA4	970C	STA A	0CH	;DIVIDEND (LO)
FAA6	901A	LIA A	4AH	
FAA8	9707	STA A	0FH	;CALIB TO DIVISOR (LO)
FAAA	964F	LDA A	4EH	
FAAC	970E	STA A	0EH	;CALIB TO DIVISOR (MED)
FAAE	B1F184	JSR	DVID	;DIVIDE PRODUCT BY 10 N.
FAB0	960C	LIA A	0CE	;DISPLAY PEAK THRUST
FAB2	9740	STA A	40H	;FMAX (LC) TO HEXBCD
FAB4	960F	LDA A	0EH	
FAB6	9741	STA A	41H	;FMAX (HI) TO HEXBCD
FAB8	B1FF70	JSR	HEXB01	
FABA	7EFB65	JMP	KEYIN	
FABC	FLB4	EQU	0FLB4H	
FAC0	FLCF	MULT	0FLCFH	

FD84	CE0019	>	EVID	ORG	0FL84H	
FL87	9E00			LIX	#25	
FL89	9707			LIA A	#00	
FD8B	9708			STA A	07	;CLEAR UPPER DIVIVILEND
FL8D	9709			STA A	08	
FD8F	9609		UNSDV1	STA A	09	
FL91	9712			LDA A	09	;SUBTRACT DIVISOR FROM
FL93	900F			STA A	12H	;UPPER DIVILEND
FL95	9709			SUB A	0FH	
FL97	9E08			STA A	0E	
FD99	9711			LEA A	08H	
FL9B	920E			STA A	11H	
FL9D	9708			SEC A	0EH	
FD9F	9607			STA A	08	
FLA1	9710			LDA A	07	
FLA3	920L			STA A	10H	
FLA5	9707			SBC A	0EH	
FLA7	240F			STA A	07H	
FLA9	9610			BCC	UNSI42	;BRANCH IF NO OVERFLOW
FLAB	9707			LDA A	10H	;CANCEL SUBTRACT IF OVERFLOW
FLA1	9611			STA A	07	
FLA7	9708			LEA A	11H	
FLB1	9612			STA A	08	
FLB3	9709			LEA A	12H	
FLB5	0C			STA A	09	
FLB6	2001			CLC		;PUT A 0 IN QUOTIENT
FLB8	0D		UNSDV2	BRA	UNSI43	
FLB9	79000C		UNSI43	SEC		;PUT A 1 IN QUOTIENT
FLBC	79000B			ROL	0CH	
FLBF	79000A			ROL	0EH	
FLC2	790009			ROL	0AH	
FLC5	790008			ROL	09	
FLC8	790007			ROL	0E	
FLCB	09			ROL	07	;ROTATE DIVILEND LEFT
FLCC	25C1			DEX		;TEST COUNTER
FLCE	39			BNE	UNSI41	
				RTS		
				END		
				ORG	0FDCFH	
FD0F	8600	>	MULT	LDA A	#00	
FL11	9700			STA A	00	
FL13	9701			STA A	01	
FDD5	CE0019		UNSM0	LIX	#25	;SET ITERATION COUNTER
FL18	0C			CLC		
FL19	760000		UNSM1	RCR	00	
FDDC	760001			ROR	01	
FL1F	760002			ROR	02	
FL22	760003			ROR	03	
FDE5	760004			ROR	04	
FL28	09			DEX		;DECREMENT COUNTER
FDE9	1F19			STX	19H	
FL2B	2712			BEQ	UNSM2	;TEST COUNTER
FL2D	24EA			BCC	UNSM1	;BRANCH IF MULTIPLIER BIT IS 0
FL2F	9601			LEA A	01	;ALL MULTIPLICAND IF
FLF1	0C			CLC		;MULTIPLIER BIT IS 1
FLF2	9906			ADC A	06	
FD44	9701			STA A	01	
FLF6	9600			LIA A	00	
FLF8	9905			ADC A	05	
FDFA	9700			STA A	00	
FLFC	7EFD19	>	UNSM2	JMP	UNSM1	
FLFF	39			RTS		
				END		

PCD5	3F	>	CALIB	ORG	0FC15H	
PC16	BLFC37			SWI		
PC19	9614			JSR	AVG8	;STORES ADC OUTPUT
PC1B	904C			LCA A	14H	;FOR 10N INPUT
PC1E	974A			SUB A	4CH	;SUBTRACT CALIB ZERO REF (LO)
PC1F	9615			STA A	4AH	;CALIBRATION (LO)
PC1	924D			LIA A	15H	
PC3	974B			SBC A	41H	;SUBTRACTS CALIB ZERO REF (HI)
PC5	9642			STA A	4FH	;CALIBRATION (HI)
PC7	81B3			LIA A	42H	
PC9	2704			CMP A	#013H	;LIGHT UP CALIB LED
PCB	8643	SET4		BEQ	SET6	
PC1	2002			LDA A	#43H	
PCF	8663	SET6		BRA	PUT0	
PCF1	B71340	PUT0		LCA A	#63H	
PCF4	9742			STA A	PAD	
PCF6	860F			STA A	42H	
PCF8	974E			LCA A	#0FH	
PCFA	B71345			STA A	4EH	;SET CALIBRATION INDICATOR
PCFD	7EFB65			STA A	1345H	;DISABLE NMI
	FB31	>		JMP	KEYIN	
FB31	3F		CALSET	ORG	0FB31H	
FB32	BLFC37			SWI		
FB35	9614			JSR	AVG8	;STORES ADC OUTPUT
FB37	974C			LEA A	14H	;FOR ZERO WEIGHT
FB39	9615			STA A	4CH	;CALIB ZERO REF (LO)
FB3B	974D			LEA A	15H	
FB3D	B71345			STA A	4DH	;CALIB ZERO REF (HI)
FB40	7EFB65			STA A	1345H	;DISABLE NMI
	FE65			JMP	KEYIN	
	FC37	KEYIN		EQU	0FB65H	
	1340	AVG8		EQU	0FC37H	
		PAD		EQU	1340H	
				END		
FC00		>		ORG	2FC00H	
FC00	7EFB65		VECTOR	JMP	0FB65H	;KEYIN
FC03	7EFE00			JMP	0FE00H	;OPER (NO TEMP)
FC06	73FE00			JMP	0FE00H	;OPER
FC09	7EFB31			JMP	0FE31H	;CALSET
FC0C	7EFC15			JMP	0FC15H	;CALIB
FC0F	7EFBA0			JMP	0FBA0H	;DDTHST
FC12	7EFBAC			JMP	0FBACH	;IDTEMP
FC15	7EFA00			JMP	0FA00H	;DTHST
FC18	7EFBC8			JMP	0FBC8H	;LIMP
FC1B	7EFB00			JMP	0FB00H	;DBTIM
FC1E	7EFB24			JMP	0FB24H	;ITV
FC21	7EFA40			JMP	0FA40H	;TEMP
FC24	7EF800			JMP	0F800H	; (APCOR)
FC27	7EFC00			JMP	0FC00H	;LCSC
FC2A	7EFB24			JMP	0FB24H	;DDTIM
FC2E	BLFC8B			JSR	0FC8BH	;ZERO
FC30	7EFB65			JMP	0FB65H	
	FFF8	>		ORG	0FFF8H	
FFF8	F4B0			WORD	0FAB0H	;IRC VECTOR
FFFA	FC68			WORD	0FC68H	;SWI VECTOR
FFFC	FC68			WORD	0FC68H	;NMI VECTOR
FFFE	FB43			WORD	0FB43H	;STARTUP VECTOR
				END		

FBC8	9632	>	LIMP	ORG	0FBC8H	
FBCA	9632			LDA A	32H	; COMPUTES & DISPLAYS IMPULSE
FBCC	97EC			ADD A	32H	; DOUBLE IMPULS
FBCE	9633			STA A	3CH	
FBD0	9633			LDA A	33H	
FBD2	970B			ADC A	33H	
FBI4	9634			STA A	03H	
FBI6	9634			LIA A	34H	
FBD8	972A			ADC A	34H	
FBI A	964A			STA A	0AH	; STORE IMPULS IN DIVIDEND
FBDC	970F			LIA A	4AH	
FBCE	964B			STA A	0FH	
FBE0	970E			LDA A	4FE	
FBE2	96E0			STA A	0EH	; STORE CALIBRATION IN DIVISOR
FBE4	9701			LDA A	#00	
FBE6	BLF164			STA A	3EH	
FBE8	960B			JSR	IVIL	; PERFORM DIVISION TO GET
FBE9	9741			LIA A	0BH	; UNITS OF .01 N-SEC
FBE1	960C			STA A	41H	
FBEF	9740			LIA A	0CH	
FBF1	BLFF70			STA A	4EH	; CONVERT QUOTIENT TO BCD
FBF4	9645			JSR	HEXBCD	; AND DISPLAY
FBF6	8002			LIA A	49H	
FBF8	2E03			SUB A	#2	
FBFA	7EFF17			EGT	ECK	
FBFD	7EFB65			JMP	TEST3	
FB00			BCK	JMP	KEYIN	
FB00	9637			ORG	2FE00H	
FB02	970C		>	LEPTIM	LIA A	37H
FB04	9638			STA A	0CH	; CONVERT BURN TIME
FB06	970B			LIA A	38H	; INTERVAL COUNT
FB08	8605			STA A	0EH	; MOVE BURN TIME TO DIVIDEND
FB0A	970F		SETUP	LDA A	#05	
FB0C	8600			STA A	0FH	; DIVISOR=5
FB0E	970F			LDA A	#00	; CLEAR OTHER DIVISION BYTES
FB10	970E			STA A	01H	
FB12	970A			STA A	0EH	
FB14	BLFD84			JSR	DVID	
FB17	960C			LIA A	0CH	
FB19	9740			STA A	40H	
FB1B	960B			LDA A	06H	
FB1D	9741			STA A	41H	
FB1F	BLFF70			JSR	HEXBCD	; DISPLAY TIME IN HUNDREDTHS
FB22	200A			ERA	GOEK	; OF A SECOND
FB24	963A			LIA A	3AH	; CONVERT DELAY TIME
FB26	970C		>	LETIM	STA A	0CH
FB28	963B			LIA A	3FH	; INTERVAL COUNT
FB2A	970B			STA A	0EH	; MOVE DELAY TIME TO DIVIDEND
FB2C	201A			BRA	SETUP	
FB2E	7EFL65			JMP	KEYIN	
FF17				GOEK	KEYIN	
FF70				TEST3	0FF17H	
FB65				HEXBCD	0FF70H	
FL84				KEYIN	0FF65H	
				IVIL	0FDE4H	
				END		

THIS DOCUMENT IS BEST QUALITY AVAILABLE.
 THE COPY FURNISHED TO YOU MAY CONTAIN A
 SIGNIFICANT NUMBER OF PAGES WHICH DO NOT
 REPRODUCE LEGIBLY.

F828		ORG	0F828H	
F829 9624	NETTST	LDA A	24H	
F82A 9614		SUB A	14H	
F82C 9717		STA A	17H	;NET THRUST LOW BYTE
F82E 84F0		ANL A	#0F0H	
F830 9721		STA A	21H	;UNPACKED NET THRUST MED BYTE
F832 9625		LDA A	25H	
F834 9215		SBC A	15H	
F83C 9718		STA A	18H	;NET THRUST HIGH BYTE
F839 8460		ANL A	#80H	
F83A 8160		CMP A	#60H	
F83C 2608		BNE	NET	
F83E 8C00		LDA A	#00	;MAKES NET THRUST ZERO
F840 9717		STA A	17H	;IF NEGATIVE
F842 9718		STA A	18H	
F844 9721		STA A	21H	
F846 C50004	NET	LIX	#04	
F849 9617		LIA A	17H	
F84B 1618		LDA B	18H	
F84L 48	UNPCK	ASL A		
F84E 58		ASL B		
F84F 09		DEX		
F850 1F19		STX	19H	
F852 2EF9		BGT	UNPCK	
F854 9720	BACK	STA A	20H	;UNPACKED NET THRUST LOW BYTE
F856 1722		STA B	22H	;UNPACKED NET THRUST HI BYTE
F858 39		RTS		
		END		
FBA0		ORG	0FBA0H	
FBA0 9613	DLTHST	LIA A	#013H	;ENABLE THRUST COMPARTOR
FBA2 971340		STA A	PAL	
FBA5 9642		LIA A	42H	
FBA7 571340		STA A	PAD	
FBA9 260A		BCA	ATOL	
FBA0 8603	DLTEMP	LIA A	#0C3H	;ENABLE THERMOCOUPLE COMPARTOR
FBAE 971340		STA A	PAL	
FBB1 9642		LDA A	42H	
FBB3 571340		STA A	PAD	
FBB6 10FD00	ATOL	JSR	ALC12	
FBB9 9624		LIA A	24H	
FBBB 9740		STA A	40H	;LOW ADC TO HEXBCD
FBBE 9625		LIA A	25H	
FBEF 9741		STA A	41H	;HIGH ADC TO HEXBCD
FEC1 B0FF70		JSR	HEXBC1	
FEC4 7EF665		JMP	KEYIN	
FF70	HEXBCD	EQU	0FF70H	
166	KEYIN	EQU	0FF65H	
1346	PAL	EQU	1340H	
FE06	ADC12	EQU	2FE02H	
		END		

THIS DOCUMENT IS UNCLASSIFIED PRACTICALLY.
 IT IS THE PROPERTY OF THE AIR FORCE AND IS LOANED TO YOU.
 IT IS TO BE RETURNED TO THE AIR FORCE AT THE END OF THE LOAN PERIOD.
 REPRODUCE LEGALLY.

LIST OF REFERENCES

1. Ott, H.W., Noise Reduction Techniques in Electronic Systems, John Wiley, 1976.
2. Cook, N.H., and Rabinowicz, E., Physical Measurement and Analysis, pp. 125-134, Addison-Wesley, 1963.
3. Baumeister, I., and Marks, L., Standard Handbook for Mechanical Engineers, 7th ed., pp. 5-30 - 5-104, McGraw-Hill, 1967.
4. Wait, J.V., Huelsman, L.P., and Korn, G.A., Introduction to Operational Amplifier Theory and Applications, McGraw-Hill, 1975.
5. National Semiconductor Corp., Linear Applications Handbook, vol. 1, 1977.
6. Analog Devices, Inc., Data Acquisition Products Catalog, 1978.
7. Millman, J., and Halkias, C.C., Integrated Electronics, pp. 549-553, McGraw-Hill, 1972.
8. Miller, J.T., The Instrument Manual, pp. 210-264, United Trade Press, 1971.
9. Weast, R.C., Handbook of Chemistry and Physics, 49th ed., pp. E103-E111, Chemical Rubber Co., 1968.
10. Analog Devices, Inc., Analog-Digital Conversion Handbook, 1972.
11. Intel Corporation, Component Data Catalog, 1978.
12. Peatman, J.B., Microcomputer-Based Design, McGraw-Hill, 1977.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3. Department Chairman, Code 62 Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
4. Assoc Professor M. L. Cotton, Code 62 Cc Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
5. LT Arthur H. Barber III, USN 1408 N. Cascade Ave. Colorado Springs, Colorado 80907	1