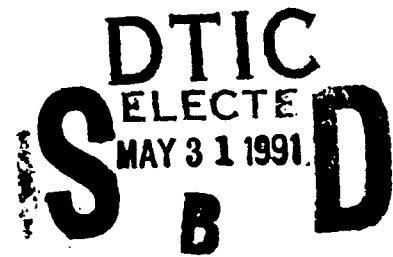# NAVAL POSTGRADUATE SCHOOL
Monterey, California

AD-A236 618

THESIS

AN AERONAUTICAL ENGINEERING USER'S MANUAL
TO THE
HP-28S HANDHELD CALCULATOR

by

Harry F. Molyneux

March, 1991

Thesis Advisor: Oscar Biblarz

Approved for public release; distribution is unlimited

91-00674

91 5 29 033

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY Multiple Sources	3. DISTRIBUTION / AVAILABILITY OF REPORT
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE (OADR)	

4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School	6b. OFFICE SYMBOL (If applicable) 31	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School

6c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000	7b. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000

8a. NAME OF FUNDING / SPONSORING ORGANIZATION	8b. OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS			
	PROGRAM ELEMENT NO	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)

AN AERONAUTICAL ENGINEERING USER'S MANUAL TO THE HP-28S HANDHELD CALCULATOR

12. PERSONAL AUTHOR(S)
MOLYNEUX, HARRY F.

13a. TYPE OF REPORT Master's Thesis	13b. TIME COVERED FROM _____ TO _____	14. DATE OF REPORT (Year, Month, Day) 1991 Mar 27	15 PAGE COUNT 69

16 SUPPLEMENTARY NOTATION

17	COSATI CODES		18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	HP-28S, handheld calculator, gas dynamics,
			programmable calculator

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

Handheld calculators are now powerful enough to have become indispensible tools for the engineer and scientist. With enhanced equation solving capability and extensive user memory, the HP-28S introduces exciting new possibilities. The entire set of tables for one-dimensional gas dynamics can be accessed with unequalled accuracy and speed.

But this enhanced power cannot be properly tapped without a pre-planned user directory organization which takes advantage of the HP-28S internal structure. Experience has shown that many students buy expensive programmable calculators but underuse them, finding their powerfulness baffling and frustrating. They employ the same computational techniques with sophisticated $200 continuous-memory programmable machines as could be accomplished with a simple $20 scientific calculator.

This manual contains a compendium of useful formulae, programming, and computational techniques for the popular HP-28S Pocket Calculator. In addition to helpful instructions on units conversion, directory organization, and problem-solving methodology which will benefit any HP-28S user, the Aeronautical Engineering student will find sections on Thermodynamics, Aerodynamics, and Controls which will prove useful in those fields of study.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION
22a NAME OF RESPONSIBLE INDIVIDUAL Oscar Biblarz	22b TELEPHONE (Include Area Code) (408) 646-2491	22c OFFICE SYMBOL AA/Bi

An Aeronautical Engineering User's Manual
to the
HP-28S Handheld Calculator

by

Harry F. Molyneux
Lieutenant Commander, United States Navy
B.S., Purdue University, 1980

Submitted in partial fulfillment
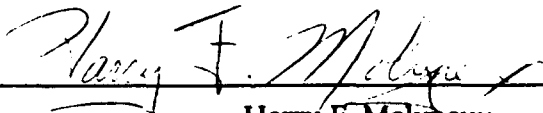of the requirements for the degree of

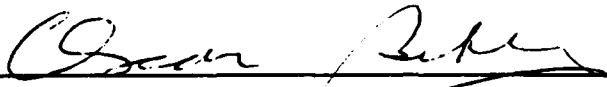MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING
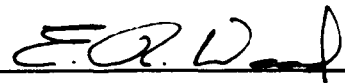
from the

NAVAL POSTGRADUATE SCHOOL
March 1991

Author: _____
Harry F. Molyneux

Approved by: _____
Oscar Biblarz, Thesis Advisor

_____
Louis V. Schmidt, Second Reader

_____
E. R. Wood, Chairman
Department of Aeronautics and Astronautics

ABSTRACT

Handheld calculators are now powerful enough to have become indispensible tools for the engineer and scientist. With enhanced equation solving capability and extensive user memory, the HP-28S introduces exciting new possibilities. The entire set of tables for one-dimensional gas dynamics can be accessed with unequalled accuracy and speed.

But this enhanced power cannot be properly tapped without a pre-planned user directory organization which takes advantage of the HP-28S internal structure. Experience has shown that many students buy expensive programmable calculators but underuse them, finding their powerfulness baffling and frustrating. They employ the same computational techniques with sophisticated $200 continuous-memory programmable machines as could be accomplished with a simple $20 scientific calculator.

This manual contains a compendium of useful formulae, programming, and computational techniques for the popular HP-28S Pocket Calculator. In addition to helpful instructions on units conversion, directory organization, and problem-solving methodology which will benefit any HP-28S user, the Aeronautical Engineering student will find sections on Thermodynamics, Aerodynamics, and Controls which will prove useful in those fields of study.

iii

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

The author wishes to extend his heartfelt thanks to the following people:

LT Dennis Sorensen, U.S. Navy, a fellow Test Pilot School instructor who provided continual support throughout this endeavor;

MAJ Mike Skaggs, U.S. Army, who spent many hours testing the author's programs and theories, and was inspirational in coaxing the author to correct inefficiencies in the IEEE student's HP-28S programming guide;

CDR Duym, curricular officer of the Aeronautical Engineering Department, who went into battle for the author when he most needed an ally;

And professors Biblarz and Schmidt, who were invaluable in turning the author's work into a Master's thesis.

The author further wishes to thank all of the students at NPS who tested the programs contained herein and offered advice and support for creating a user's manual.

Finally, the author recognizes Jan Kleinschmidt and Annette Molyneux as the two ladies who kept him going when the going got tough.

Thanks, everyone.

I.   INTRODUCTION

A.   WHY THIS MANUAL WAS WRITTEN

Recently, the Hewlett Packard HP-28S Advanced Scientific Calculator has been one of the most popular and powerful tools available for the serious science and engineering student. Unfortunately, the owner's manual [Ref. 1] and reference guide [Ref. 2] provided by Hewlett Packard for these handheld calculators do not furnish a sufficiently clear tutorial on many of the most useful features embedded in its design, nor is an efficient organizational methodology for user-created software proposed. Not unlike other such calculators, users have been forced to devise their own schemes to extract the most performance from this machine. This work proposes to document one such effort.

Many operations and engineering problems become quite cumbersome if the user-programmable software is set up in an inefficient manner, since the most obvious or readily obtainable organizational schemes are not necessarily the most efficient. Solving problems quickly with the HP-28S requires skill and some degree of prior planning. A systematic approach and a consistent problem-solving methodology yield not only fast results, but also a reduced probability of error.

This manual was written to assist most serious users of the HP-28S and, in particular, Aeronautical Engineering students. If the guidelines delineated herein are followed, the user will be rewarded with:

1. a logically-organized internal directory structure which will allow fast access to stored equations,

2. the ability to make unit conversions instantly, correctly, and accurately,

3. a universal problem-solving methodology which will allow difficult problems to be handled with confidence and minimal confusion.

B. THE IMPORTANCE OF ORGANIZATION

The key to extracting the most from the HP-28S lies with getting the directory structure organized, yet no guidance on this crucial task is provided by the manufacturer. Though the functions used to create directories are adequately described in the owner's manual [Ref. 1], the user is left on his own to decide which directories to create or how to arrange them.

Organizing the directories properly must be done for several important reasons.

1. A certain logical organization enables the user to locate stored programs, equations, and constants quickly.

2. A working area must be set aside ahead of time where temporary variables can be created, stored, and manipulated easily without losing them among the programs, equations, and constants previously stored by the user.

2

3. An isolated working area also prevents accidental deletion of stored programs and equations when temporary variables are deleted after solving a problem.

A one-time investment of a few hours spent programming the calculator and organizing the directory tree structure will save a great deal of time later on. Without such an organization, the rapid, systematic approach to solving problems with the HP-28S suggested here is not possible.

C. A WARNING ON BATTERIES

Before proceeding with an extensive programming effort, the user should consider the following information on the importance of the condition of the batteries. Battery power is necessary to retain the user memory of the HP-28S.

Hewlett Packard literature [Ref. 2] claims approximately 10 hours of calculator usage remains at the first indication of low battery power. Several owners, however, have reported a low battery indication followed within minutes by complete loss of calculator memory.

Since the typical engineering student may well spend many laborious hours entering equations and formulae into the 32K memory of this machine, a sudden loss can be disastrous. It is therefore strongly recommended that the calculator be shut off IMMEDIATELY upon the first indication of low battery power to prevent possible loss of all user-entered data.

Further, it is recommended that fresh batteries be

purchased from an electronics store such as Radio Shack, rather than from a department store where battery storage conditions and shelf life are not as carefully monitored. Comparisons among users indicate a much longer battery life from those batteries purchased at specialty electronics stores.

Before investing a great deal of time and effort programming the HP-28S, the user should be certain the batteries installed in his machine are fresh. Losing the contents of the calculator's memory is a painful but entirely unnecessary experience.

II. DIRECTORY ORGANIZATION

A. GENERAL STRUCTURE

The HP-28S has a directory structure which can be likened to the roots of a tree. The highest directory, located at the "base" of the tree, is the HOME directory. One or more subdirectories may be created below the HOME directory, and each subdirectory may have its own subdirectories branching down below it.

Programs, equations, or data in various formats may be stored in the HOME directory or in any subdirectory, and may be duplicated in several directories. The same variable name can be used to store different constants in different directories without internal conflict in the calculator, however duplicated information may easily lead to user confusion!

The computer operates as follows: When a lower level subdirectory is the current directory, the calculator can "see" all the programs, equations, variables, and other stored information contained in the directories "above". When data are recalled from memory, the calculator first attempts to access the current directory to obtain the information. If these data are not located in the current directory, the HP-28S will systematically search the directories in ascending

order up the tree root structure until the HOME directory is reached and searched.

B. THE HOME DIRECTORY

The bottom-to-top search pattern in the calculator's user memory means that no matter which subdirectory is current, the information stored in the HOME directory can always be accessed from anywhere in the tree.

HOME is thus the logical place to keep "global" constants and utility programs. As an example, the user may have a need to switch the calculator back and forth from numeric to symbolic modes. Utility programs to accomplish this can be left in the HOME directory and called with a few keystrokes while working a problem several subdirectories down -- without the need to change directories.

Everything cannot be left in the HOME directory, however, without serious trouble developing. The HOME directory quickly becomes too cluttered, and soon the user has difficulty finding stored programs and equations, or worse, he finds himself deleting items he should have kept.

C. SUBDIRECTORY ORGANIZATION

An optimal tree structure for the HP-28S allows fast access to all user-entered information, an uncluttered working area for storage of temporary variables, and access to global constants and programs without risk to the HOME directory.

If the organizational guidelines provided herein are followed, the user will almost never get "lost in his calculator" again.

1. The PLAY Directory

HOME is where the user should keep those "global" programs and constants as mentioned earlier. There should be only <u>one</u> subdirectory under HOME: the primary working area, which in this manual is called PLAY. Users should rarely need to go into the HOME directory, and should <u>never</u> do any computational work in there.

PLAY is where <u>all</u> computations should be done. All of the temporary variables created while solving a problem should be stored in PLAY. Once the user has finished working a problem, everything stored in this directory can be quickly purged, and the calculator is ready for the next problem. The PLAY area should be kept clear unless the user is actively involved in a problem.

2. The DOWN Directory

Directly below the PLAY directory is the one this manual calls DOWN. This directory contains all of the subdirectory storage areas, and can be considered as the gateway to all of the user's personal programs and equations.

Each subdirectory contained within the DOWN directory may be likened to a filing cabinet drawer. Separate "drawers" for individual applications allow rapid access to stored

equations. The typical Aeronautical Engineering student will wish to create separate subdirectories for each of various fields of study: Gas Dynamics, Structures, Aerodynamics, Materials Science, Controls, Propulsion, and so on.

When the user steps DOWN from PLAY, he is then able to access any of his subdirectories with a single step.

3. The UNITS Subdirectory

Though the HP-28S carries hundreds of useful scientific and engineering conversion factors in its memory, using this built-in feature is tedious and frustrating. The creation of a subdirectory with stored conversion codes classified by type makes this task quick, simple, and risk-free.

4. Problem Solving Methodology

The basic operational strategy suggested here is as follows: The user starts in the PLAY directory. As he works, he will occasionally pop DOWN to a subdirectory to grab an equation, then pop back up to PLAY and work the problem. If, as he works, a units conversion is required, he will pop DOWN to the UNITS subdirectory, convert, and pop back up. Once the problem is solved, any temporary variables or constants stored in the PLAY directory are deleted.

Think of the arrangement like a nursery school. The PLAY area is kept tidy. When the user steps DOWN from PLAY, he is then able to open the drawers or cabinets where the

"toys" are kept in an orderly arrangement. The user selects the "toys" he wishes, then pops back to the PLAY area to enjoy them. Once the "fun" is over, the PLAY area is quickly cleaned, ready for the process to begin anew.

Once again, it should be emphasized: At no time should the user ever need to go into the HOME directory while solving problems. This may sound a bit confusing at first, but after some practice the user will quickly become adept at moving through the directory tree structure.

A diagram of the directory structure used in this manual is presented in Figure 1, Appendix A.

III. HOW TO USE THIS MANUAL

A. CASE SENSITIVITY

The programs listed herein must be typed into the HP-28S exactly as shown. It is crucial to observe upper and lower case carefully. In the calculator, 'X' and 'x' might have been defined to represent two different values; in fact, one could be a complex number or a string, and the other could be a matrix!

Note for example that the statement 'sin=5', using all lowercase letters, is perfectly legal, and assigns the value 5 to the variable named 'sin'. The sine trig function is called SIN, with all uppercase letters, and neither this function nor the 'sin' variable is equivalent to another variable named 'Sin'!

To further confuse things, all upper and lower case characters are displayed only in upper case on the softkey display. Thus, two variables 'X' and 'x' could both appear simultaneously as 'X' on the display.

It is thereby apparent that things may get very confusing if capitals and small letters are indiscriminately mixed, thus care must be exercised to avoid this. Programs listed in this manual must be keyed in carefully with regard to upper and lower case, or errors will certainly result.

B. THE PROGRAM DELIMITER

When the symbols << or >> are shown in the program listings, this refers to the HP program delimiters found on the left side keyboard next to the SPACE key. It is one key, not a pair of "less than" or "greater than" signs. These delimiters indicate the boundaries of an executable program or subroutine.

C. CODES USED IN THIS MANUAL

1. Capital Letters Indicate Keyboard Features

Generally, words which appear in all caps, like DROP, ENTER, or PURGE can be found on the right side keyboard. If the word is not found there, it is probably a variable name which must be keyed in letter by letter.

2. Boldface indicates User Menu Features

Any word which appears in boldface type, like **DOWN**, refers to a function stored on the USER menu. These functions should be entered using the "softkeys" (the top row of keys on the right side keyboard).

3. Underlining Indicates HP Menu Functions

Underlined words appearing in this manual refer to functions found in the HP menus. As an example, it is not necessary to type S-I-N for the trigonometric sine function, even though it is perfectly legal to do so on the HP-28S. Instead, the user should call up the TRIG menu, locate the <u>SIN</u> function, and push the corresponding softkey.

In this manual, HP-resident functions are shown underlined, and the menu on which they appear is shown to the right after the # sign. **DO NOT TYPE IN THE MENU NAME WHEN ENTERING PROGRAMS.** It is included only to assist the user in entering the programs faster.

For example, enter the following program:

<< " This is the top line." 1 <u>DISP</u> >>   #CONTRL

This indicates that the function <u>DISP</u> can be found in the CONTRL menu. When entering this program, the user may go to that menu and push the <u>DISP</u> softkey instead of typing D-I-S-P on the alpha keys. Note in this case that the <u>DISP</u> function is on the second page of the CONTRL menu; the user must push the NEXT key to call <u>DISP</u> up to the display.

IV. PRELIMINARY ORGANIZATION

A. SOME HANDY GLOBAL UTILITIES

The user can begin organizing his calculator by storing the following six useful utility programs in the HOME directory. To make sure HOME is the current directory, type:

HOME USER   #MEMORY

1. Functions to Reposition the UP and DOWN Keys

Input the following programs. NOTE: the { key is the one above the = sign.

<< {DOWN} ORDER ENTER 'DD STO USER   #MEMORY

<< {UP} ORDER ENTER 'UU STO USER   #MEMORY

If this sequence was keyed in properly, the left-most softkeys should now be titled **UU** and **DD**. These utility programs will be fully explained later in the manual.

2. Speaker Control Utilities

Continue by entering the following two programs in the HOME directory:

<< 51 SF ENTER 'SHHH STO   #TEST

<< 51 CF ENTER 'NOISE STO   #TEST

Two more softkeys titled **SHHH** and **NOISE** should now appear on the USER menu. These utility programs will also be explained in the next section, "USING GLOBAL UTILITIES."

3. Numeric/Symbolic Mode Control

Finish by installing these last two global utilities, whose functions will also be explained in the next section.

```
<< 35 SF 36 SF ENTER 'SY STO          #TEST
<< 35 CF 36 CF ENTER 'NU STO          #TEST
```

4. Program Entry Check

If all of these programs have been installed correctly, pressing the USER key will reveal the newly-created functions **NU**, **SY**, **NOISE**, **SHHH**, **UU** and **DD** on the softkey display.

B. CONSTRUCTING THE DIRECTORY TREE

If the user has already stored information in other subdirectories, there is no need for concern. Data can be moved later, and nothing previously saved will be lost.

1. The PLAY Directory

Begin building the directory tree structure by creating the **PLAY** area, where the majority of all computational work in the future will be performed.

```
PLAY CRDIR          #MEMORY
```

Now drop into the **PLAY** directory by hitting USER, then the new **PLAY** softkey. The HP-28S is now in the PLAY directory, which is empty.

2. The DOWN Directory

Next create the **DOWN** directory below **PLAY**.

```
DOWN CRDIR          #MEMORY
```

14

3. The UP Program

The **UP** program will become the program used most often while working with the HP-28S. This program will appear in EVERY directory with the exceptions of **PLAY** and HOME. It is very simple, but vital, since it allows the user to "pop" back to the **PLAY** directory from any other subdirectory in the tree.

The first place the **UP** program should be installed is in the newly-created **DOWN** directory. Begin by moving into **DOWN**. (Press USER, then **DOWN**.) Then enter the following:
<< PLAY ENTER 'UP STO

The user should now be able to move quickly between **PLAY** and **DOWN**. Try it a few times. Press **UP**, and the calculator moves to **PLAY**. Press **DOWN**. Press **UP**.

For now, stay in the **PLAY** directory. The user will always know when the HP-28S is in **PLAY** because the **DOWN** softkey will be on the USER menu.

C. USING GLOBAL UTILITIES

The utility programs placed in the HOME directory can now be demonstrated. Because they are global, the calculator can remain in the **PLAY** directory and still call these functions.

1. Speaker Control

With the stack cleared, press the + key. The HP-28S should make an annoying beep and display an error message. Now type SHHH on the left side keyboard and hit ENTER. Try pressing the + key again. Notice this time, the error message

still appears, but the annoying beep has been silenced. The SHHH program has turned off the HP-28S speaker. This is particularly useful when taking tests, and might save the user from embarrassment in a crowded room.

If restoration of the beep is desired, type NOISE, and hit ENTER. Now pressing the + key once again gives the beep with the error message. Most users prefer to leave their HP-28S calculators in the SHHH mode.

2. Numeric/Symbolic Mode Control

The SY and NU programs allow the user to switch back and forth between the calculator's symbolic and numeric modes. Type SY, then ENTER. Now find the π key, and ENTER. Notice that the 'π' symbol is placed on the stack. Type 6 *, and now the stack says '$\pi*6$'. This is annoying when dealing with engineering computations.

Switch to the numeric mode by typing NU and ENTER. Now type 6 /, and notice the numeric value of π is placed on the stack instead of the symbol. For most engineering work, the user will probably prefer to leave the HP-28S in the numeric mode. Remember, to switch back and forth, type either SY or NU. It is not necessary to return to the HOME directory to do so.

3. Repositioning the DOWN Softkey

The DD utility helps organize things in the PLAY directory while working a large problem. To see how it works,

make sure the calculator is in the **PLAY** directory, then save
six or more constants as temporary variables.

A way to do this quickly is to first fill the stack
with half a dozen numbers by pressing: 1 ENTER 2 ENTER 3 ENTER
4 ENTER 5 ENTER 6 ENTER. Next, store these digits by pressing
A STO B STO C STO D STO E STO F STO. Press USER and note the
new softkeys, **A**, **B**, **C**, etc., appearing on the display.

Notice that to see the **DOWN** softkey, the user must
search through the **PLAY** directory using NEXT. If many
temporary variables are stored in **PLAY**, it becomes frustrating
to hunt for the **DOWN** key. Hit USER to show the temporary
variables just created.

Now type DD on the left side keyboard and press ENTER.
Note that the DOWN key moves back to the first position in the
PLAY directory. This function allows the user to go down the
tree quickly when the PLAY directory starts to get filled up.

D. PURGING TEMPORARY VARIABLES

Clearing the **PLAY** directory of temporary variables is a
sequence the user will be required to perform often. To
accomplish this quickly, the PURGE function is invoked with a
list of the temporary variables. The keystrokes are simple
and should become second nature to the user after the first
few tries.

Begin by pressing the { key (the one above the = sign).
This is the "list delimiter". Next press the α key, which

puts the calculator in alphabetic entry mode. The Greek alpha character should appear at the top of the display.

As these two keys are pressed, it helps to say to oneself, "List Alpha". Next, go to the softkeys and press all of them except **DOWN**. The user may be required to hit the NEXT key to display and retrieve all of the temporary variables stored in **PLAY**. After hitting all of the names to be deleted, press ENTER, which puts the list on the stack, then PURGE (above the 4 key). The **PLAY** directory is now clear of all the previously-stored temporary variables!

The user should practice this a few times until he can do it quickly and without hesitation. Store several constants, then think the following words as the keys are pressed: "List Alpha bang, bang, bang...ENTER red PURGE." It should take no more than two or three seconds to clear the **PLAY** directory and prepare the calculator to start a new problem.

This process is one of the fundamental techniques necessary to develop an efficient problem solving methodology with the HP-28S.

V.   SETTING UP SUBDIRECTORIES

A. CREATING NEW STORAGE AREAS

As discussed previously, most of the programs and equations stored by the user will be kept in separate subdirectories reached from the **DOWN** directory. Engineering students typically find it useful to create directories for each of their classes. This manual provides some ideas for subdirectories which may prove most useful for the Aeronautical Engineering student, however the list is only meant to provide initial guidance. Ultimately, the user must decide what is needed to work the problems he encounters most often.

B. RELOCATING DATA

If any programs or equations have previously been stored in the HOME directory or in other subdirectories created before, now is the time to relocate them. From the **DOWN** directory, use CRDIR to make new subdirectories. Refer again to Figure 1 in Appendix A for guidance.

Whenever a new subdirectory is created, the user should immediately go into that directory and install the **UP** program. Remember that the **UP** program will allow the user to pop up to the **PLAY** directory from anywhere in the tree. Again, the **UP** program can be installed with the following sequence:

19

<< PLAY ENTER 'UP STO

To relocate programs and equations into the new subdirectories, they must first be copied to the stack and then stored in the new subdirectories. After using <u>CRDIR</u> in the **DOWN** directory to make the new subdirectory, go to the old directories (use the <u>HOME</u> function in #MEMORY). Next use the RCL function to get the desired information on the stack. Immediately after the contents of a file are on the stack, press the single quote key and the softkey again, then ENTER, which will bring the name of the file onto the stack.

With the data to be transferred and the storage name both on the stack, move down from HOME through **PLAY** and **DOWN** into the new directory, and press STO to store the data under the original name.

Finally, go back to the old directory again and do the "List Alpha bang, bang, ...ENTER PURGE" trick to clean out the old copy of the file. Any other EMPTY subdirectories under HOME can also be PURGEd. When the process is completed, there should be only one subdirectory under HOME -- the **PLAY** directory.

Note that the stack can hold many programs and equations at one time, so entire subdirectories can be copied to new locations with a few keystrokes. Merely RCL the contents of a softkey, then the name of the softkey for each one to be transferred. When in the new subdirectory, press STO until the stack is cleared.

Organization of the calculator's user memory in this fashion is crucial to developing a systematic procedure for mastering complicated problems and quickly locating stored programs and equations. When this step is completed, the user should have a directory tree structure with HOME, **PLAY**, and **DOWN** as the top three directories, and with no other subdirectories except **PLAY** accessible from HOME.

C. REPOSITIONING THE UP PROGRAM SOFTKEY

It is convenient to keep the UP softkey as the left-most key in every directory. The **UU** global utility installed earlier does this. When the calculator is in any directory containing the **UP** program, type UU on the left side keyboard, and press ENTER. The **UP** softkey will be repositioned to the first space on the user menu.

D. ORDERING THE SOFTKEYS IN A SUBDIRECTORY

To put files or subdirectories in a desired order, create a list and use the ORDER function in #MEMORY. The keystrokes are similar to those used in the PURGE routine: First, hit the list delimiter, {, then the α key, then hit each of the softkeys in the desired order, then ENTER. This puts the list on the stack. Now press ORDER, then USER again, and the softkeys will be in the desired sequence. Once a particular subdirectory is in the desired order, it should not be necessary to do it again.

VI.   THE UNIT CONVERSION DIRECTORY

A.   BACKGROUND

Converting units with the HP-28S is supposed to be easy, but it is not always so. For example, say a user decides to convert 2.0 BTUs of energy to foot-pounds. The procedure is as follows:

1. Enter a 2 on the stack.

2. Type 'Btu and ENTER. (NOTE CASE SENSITIVITY!)

3. Type "ft * lbf" and ENTER. (Note the <u>double</u> quotes!)

4. Finally, press CONVERT. (Located over the comma key.)

5. The result, 1556.33852453 ft-lbf, is shown on the stack, with the unit string still left on the first level.

The problems with this method are obvious. First, the user must remember and type in the proper unit exactly as the machine expects it. 'BTU' or 'btu' will not work! Although the proper unit string can be "fetched" off the HP Units menu, located over the R key, this method requires a great deal of button mashing. Also, the HP Units menu has no provision for something like foot-pounds, so a string with the mathematical expression inside must be constructed by hand. Lastly, after the conversion is done, the conversion unit string remains on the stack, and must be DROPped to get it out of the way.

After all this trouble, most engineering students decide that simply multiplying BTUs by 778 is close enough! The problems with this "fall-back" method are that it requires the user to carry (or memorize) a table of conversion factors, it does not perform the conversion to the limits of the calculator accuracy, and there remains the risk of operator error such as when conversion constants are multiplied instead of divided.

B. A BETTER METHOD

Fortunately, there is a way to use the calculator's stored conversion constants without having to enter or memorize complicated codes. By setting up the conversion subdirectories as instructed in this manual, the user will be able to convert any quantity to different units in seconds, with full 13 place accuracy and minimal possibility of making errors.

The secret is to create conversion areas which contain the proper strings or unit codes for the units most frequently used. Each conversion area has four different units in it, plus a **CONV** program, and the ubiquitous **UP** program.

To see how this works, the steps are provided here to set up the units conversion area for energy. To see how the **UNIT** subdirectory is structured, see Figure 2, Appendix A.

To begin, first go to **DOWN** and create the **UNIT** subdirectory. Next, go into **UNIT** and install a copy of **UP**.

Now from within the **UNIT** subdirectory, create another directory and call it **ENRG** (for converting units of ENeRGy.) After creating the **ENRG** directory, press UU and ENTER to relocate the **UP** program softkey in the **UNIT** subdirectory.

Now go into **ENRG** and enter the following functions. (Note: the → key is found on the left side keyboard over the letter U.)

```
<< CONVERT DROP ENTER 'CONV STO
'J ENTER   '→J STO
'Btu ENTER   '→BTU STO
"ft*lbf ENTER   '→FTLB STO
'cal ENTER   '→CAL STO
<< PLAY ENTER   'UP STO
```

Now test it. Put a 1 on the stack. Press →BTU, then →FTLB, then **CONV**. The number 778.1692... should appear. Continue by converting from foot-pounds to calories, then calories to Joules, and finally from Joules back to BTUs. After these four conversions, the number 1, indicating the original 1 BTU, should remain on the stack.

Now try it from the **PLAY** directory. Go **UP** and pretend to be working a problem. Suddenly, a number in BTUs which should be in foot-pounds appears on the stack. (Enter a number.) Pop **DOWN** to **UNIT** then **ENRG**. Press →BTU to →FTLB, **CONV**, **UP**. The conversion is fast, accurate, and simple, with no unit string residue left on the stack.

The user should practice this until it can be accomplished quickly and effortlessly. When comfortable with the method, conversion directories for other common groups of units may be installed as desired in the **UNIT** subdirectory. Some common conversion directories are presented in Appendix B. The user may create as many conversion directories as he wishes. Remember to create each of these directories from the **UNIT** directory, so that the resulting tree structure looks like Figure 2 in Appendix A.

VII. USING THE HP SOLVER

A. CAPABILITIES OF THE SOLVER

The HP-28S Solver is one of the most powerful and useful features of this calculator. The Solver can take virtually any algebraic equation and solve it for an unknown variable. A complicated equation with N variables can be solved for any one of the variables if N-1 of them are known. Without the Solver, a user would have to re-write equations in terms of the unknown variable, a time consuming and sometimes unwieldy task.

With the HP-28S, the engineering student may call useful equations from storage in his subdirectories, load them into the Solver, and obtain a solution in seconds.

B. LIMITATIONS OF THE SOLVER

To illustrate the limitations of the Solver, consider the equation $y = x^2 + 1$. Before proceeding, sketch this function on a piece of paper on the interval x = -3 to x = +3, noting that it is defined for all real x, but only for y greater than or equal to 1.

Enter the equation 'Y = X^2 + 1' onto the stack, then press SOLV **STEQ SOLVR**. Note the left-most softkeys are Y and X.

Press 1 and then the X softkey to enter 1 for X. Now ask, "What is Y?" by pressing the red button followed by the Y softkey. The result, 2, is returned, along with the word "Zero" on the second line of the display. This is important! It means the answer given exactly satisfied the equation.

The Solver works by making successive approximations to find an answer that satisfies the equation. When it returns the "Zero" message, it means the difference between the left and right side of the equation is zero, thus an exact solution has been found.

Incidentally, the HP-28S allows the user to observe this iterative process, if desired. Immediately after pressing the red button and the softkey for the variable to be solved, press ENTER. The display will show the successive approximation process in action.

Now do a problem where Y is known and the Solver must solve for X. Enter 9 for Y by pushing the 9 and then the Y softkey. Ask, "What is X?" by pressing the red button and then the X softkey. The result is the square root of 8, placed on the stack as before, but now note that the second line of the display says, "Sign Reversal." This means the calculator did NOT find an answer which exactly satisfied the equation, but is presenting an approximation which is as close as the machine accuracy allows. In other words the calculator is only able to provide a 13 digit approximation to the square root of eight. For all practical purposes, the appearance of

27

the Sign Reversal message indicates the correct answer has been found.

But what if there is no real answer? Look again at this function and note it is not defined for Y < 1. Enter zero for Y and ask the question, "What is X?" The HP-28S will take a few seconds to "think" about this, and will deliver some number close to zero, with the message, "Extremum". This message is very important! It says that the answer given <u>did not satisfy the equation</u>. Rather, the value found occurred near a relative maximum or minimum of the function and was the closest the calculator could get to satisfying the equation for the unknown variable.

It is vitally important to realize the Solver can give the wrong answer. When the word "Extremum" appears in the display, the Solver is informing the user that it could not find a value to satisfy the equation.

C. CORRECT PROCEDURE FOR USING THE SOLVER

When the value 9 was put in for Y in the previous section, the answer (an approximation for the square root of 8) was given. For this function, however, there are two values of X which satisfy the condition Y = 9. Only the positive value was returned.

To get the other value, the user must tell the Solver where to begin looking for an answer. This time, enter 9 for Y and -3 (a guess) for X. Then ask, "What is X?" by pressing

the red button and the X softkey again. This time, the negative square root of 8 is returned, with the "Sign Reversal" message.

The Solver should ALWAYS be provided with a "best guess" estimate of the answer before being asked to find a solution. Many engineering equations have multiple roots, and without an initial value to try, the Solver will frequently yield an inappropriate answer which satisfies the equation but does not apply to the problem at hand.

An example of this is the area-ratio/Mach number equation for converging/diverging supersonic nozzles. When the nozzle is choked, a specific nozzle-to-throat area ratio gives TWO Mach numbers - a subsonic one and a supersonic one. By entering an initial guess of M = 0.5, the Solver will return the subsonic answer. An initial Mach number estimate of 3 will yield the supersonic value.

The user does not need to spend a great deal of effort trying to determine an accurate initial guess for the Solver algorithm to find an answer. In the example given previously for X and Y, any positive X guess returns the positive root, and any negative guess returns the negative root. A reasonable initial guess, however, may reduce the time the Solver spends finding the desired solution.

29

D. ANOTHER HANDY GLOBAL UTILITY

The general problem-solving method proposed in this manual requires the user to retrieve his stored equations from subdirectories listed in the **DOWN** directory. As previously stated, the user will move from **PLAY** to **DOWN** to the appropriate "filing cabinet", grab an equation previously stored there, and then pop **UP** to **PLAY** with a copy of the equation on the stack.

Usually, this is immediately followed by the sequence SOLV **STEQ SOLVR**, which puts the HP-28S into the Solver with the desired equation. The user then enters the all the "knowns" in the equation and an estimate for the remaining unknown, and the Solver does the rest.

To save time, a short global utility can be created which will automatically move the calculator back to the **PLAY** directory and install the equation from the stack into the Solver, saving the repetitive "**UP** SOLV **STEQ SOLVR**" keystrokes.

As with the other global utilities, this one should be saved in the HOME directory. Follow this sequence to install the program:

HOME << PLAY STEQ 24 MENU ENTER   #MEMORY #SOLV #MEMORY
'XX STO

Press USER and note the new softkey labeled **XX**. The user may now wish to use the ORDER function described earlier to make **PLAY** the left-most softkey again; make a list of the softkeys in the desired order (**PLAY** first), ENTER it onto the

stack, and then use <u>ORDER</u> which is found in #MEMORY. Press USER, then **PLAY** to leave the HOME directory.

To test this new utility, the user must have already stored an equation into a subdirectory below **DOWN**. Move **DOWN** and into the subdirectory, then press the softkey to bring the desired equation to the stack. Type XX, then ENTER. The equation will be installed in the Solver, ready to go!

VIII. CUSTOM MENUS

A. DECREASING KEYSTROKES WITH CUSTOM MENUS

The capability to create custom menus is a function seldom used by most owners of the HP-28S, yet this handy feature can put any combination of the HP functions and user-installed functions on one menu, saving time in many calculations. It is particularly useful for accessing the global constants, functions, and programs stored in the HOME directory without having to key in the function name on the left side keyboard or go to the HOME directory to find and press the corresponding softkey.

For example, suppose a series of calculations involving natural logarithms, sine and cosine trigonometric functions, and complex numbers were to be solved. The user might spend considerable time continuously calling the LOGS, TRIG, and COMPLX menus to the display in order to access those functions while performing these calculations. By putting LN, EXP, SIN, COS, R→C, and C→R on a single menu, however, the whole series of calculations could be accomplished with all the needed functions readily available at a single keystroke.

This feature is also useful when keying in programs. For example, the program in Appendix F of this manual can be entered with much less effort if the functions COS, DUP, →STR,

and DISP are placed on a custom menu prior to keying in the program.

All of the global functions, constants, and programs stored by the user in the HOME directory can be loaded into custom menus as well. Several different custom menus can be created and installed when needed. For example, a menu with common thermodynamic constants might be useful when working those problems; whereas a system control menu would be a handy place to find the programs **NU**, **SY**, **SHHH**, and **BEEP** saved earlier.

B. A PROGRAM TO INSTALL CUSTOM MENUS

Installing a custom menu is not complicated, and merely involves making a list of the functions the user wishes to include on the menu, then pressing the MENU key on the MEMORY menu. Until a different custom menu is installed to replace it, this newly-created menu will remain available whenever the CUSTOM menu is selected from the right side keyboard.

If more than one custom menu is needed, it becomes tedious to execute the custom menu installation process repeatedly. By installing simple programs in the HOME directory which, in turn, install custom menus on demand, an assortment of custom menus may be kept and installed or exchanged at will.

The following sequence of keystrokes will place a custom menu installation program in the HOME directory. This program will put some of the global system control functions from

Chapter IV into a custom menu.

HOME USER << { **NU SY SHHH NOISE** } MENU   #MEMORY #MEMORY
ENTER 'SMEN STO

The user may wish to re-order the softkeys in the HOME directory (using the ORDER function described earlier) before moving down to the **PLAY** directory. Once back in the **PLAY** directory, the program may be tested.

To install the System MENu, key in SMEN on the left side keyboard and press ENTER. The **SMEN** program will install the custom menu, which then appears on the display. The user now has instant access to those global utilities via the softkeys on the CUSTOM menu. Press USER to return to the normal display, and CUSTOM to access the global utilities when desired.

Other menu installation programs can be created using the same keystrokes listed above, replacing the function names on the list with the desired functions and storing the program under a different name. Most aeronautical engineering students will require several custom menus in the course of their studies, including, perhaps, one called **GMEN** with a list of global gas dynamics constants which have been stored in the HOME directory, or one called **STMEN** with global constants or functions used to solve structures problems.

Some users wish to know how to "get rid of" a custom menu. There really is no reason to ever remove them, other than to replace them with new ones, however the custom menu display

34

can be cleared by installing an empty custom menu to replace
the existing one. The process is:

{ ENTER <u>MENU</u>   #MEMORY

IX. GAS DYNAMICS

The study of gas dynamics is computation intensive, and the problems encountered lend themselves well to a stored equation approach. The HP-28S, when programmed with a complete set of thermodynamic equations, can make problems in gas dynamics much simpler to solve. Use of the calculator to compute thermodynamic variables eliminates the possibility of errors caused by incorrect reading of graphs and tables or incorrect interpolation of tabular values.

Typically, the engineer uses tables generated by computers to find ratios of thermodynamic variables under given conditions. For example, assuming isentropic flow conditions and a fixed ratio of specific heats (denoted by the lower case Greek letter gamma), the ratio of the static pressure to the total (stagnation) pressure is a function of the Mach number. Textbooks [Ref 3] carry isentropic tables which tabulate this pressure ratio versus Mach number. The usual approach, given two of the three parameters, is to find the closest tabulated entry to the known value (either pressure ratio or Mach number, whichever is known), and use the corresponding tabulated value to compute the missing variable. This may involve interpolating between tabulated values to obtain reasonable accuracy, and generally involves an intermediate step of recording a tabulated ratio and using it in a computation.

36

With the large storage capacity of the HP-28S and the power of the Solver, looking up numbers in tables, performing interpolations, and carrying intermediate ratios are all completely unnecessary. The user need only move down to the proper "table" in the calculator (a stored equation), grab the equation, and pop back up to use it in the Solver.

An example will illustrate the superiority of this method. Suppose the following question is posed: Air with a stagnation pressure of 140 psia passes through a frictionless adiabatic nozzle. Assume gamma = 1.4. At some point in the flow, the Mach number is 0.205. What is the static pressure at this point?

Using a thermodynamic table of isentropic flow parameters, the student finds the ratio p/p_t for M = 0.20 to be 0.97250, and for M = 0.21, the ratio is 0.96973. Averaging these gives an estimate (assuming the function is reasonably linear) for M = 0.205 of 0.971115. Since p_t is known, p can be computed by multiplying this ratio by 140 to obtain p = 135.956 psia. A reasonably fast engineer with the tables handy and an ordinary calculator can work this problem in about one minute.

This is how the same problem can be worked without tables using the HP-28S: First, the user moves **DOWN** into a **GAS** directory where he finds the names of the tables listed, including **ISEN**tropic. He selects the **ISEN** directory, which contains the formulae to generate the columns of the Isentropic Flow Parameter tables. He then presses the 'p/pt'

equation softkey, calling this equation to the stack. Without

hesitation he next keys XX and ENTER, which moves the

calculator to the **PLAY** directory and installs the equation in

the Solver (see Chapter VII, Section D.) He next enters 0.205

for **M**, then 140 for **pt**, and asks the Solver, "What is **p**?"

This entire process takes approximately 20 seconds, and the

answer, 135.958 psia, is not only presented on the stack, but

is also stored in the **PLAY** directory as a temporary variable

where it can be repeatedly recalled until purged by the user.

A further advantage to using this method is that tables

may be generated for various other specific heat ratios

(values of gamma). The tables in most thermodynamics texts

are given only for a specific gamma value, usually 1.2, 1.3,

or 1.4, which means complicated interpolations for a gamma

value of 1.33, for example. With a default gamma stored in

the HOME directory of the HP-28S as a global constant, the

values generated will correspond to the tables in the text,

but the capability of instantly generating new "tables" by

changing the gamma constant in the local directory provides

the formula user a distinct advantage over the engineer with

only textbook tables.

All of the formulae necessary to generate the tables of

Ref. 3 for Isentropic, Shock, Fanno, and Rayleigh flow

parameters are presented in Appendix C. The lines and block

spacing shown in Appendix C are just to make it easier to read

the equations while keying them in. When entering each

equation, key the entries as if the entire sequence was written on one continuous line.

To install the tables in the HP-28S, the user should create a **GAS** subdirectory, with **ISEN**, **SHOCK**, **FANO**, and **RAY** subdirectories beneath it. Of course, a copy of the **UP** program should be installed in each directory as soon as it is created.

Care should be exercised in entering the equations for each subdirectory to ensure they are installed correctly. After each equation is stored, it should be called up to the Solver, then given a sample Mach number entry for input and checked against an actual table of thermodynamic flow parameters. (By pressing the **RT=** softkey, the Solver returns the ratio shown in the tables with only Mach number input.) Remember to store the default value of **GAMA** in the HOME directory before using the Solver, so that entering a value for **GAMA** will not be required during computation.

X.   EIGENVALUES OF A MATRIX

Finding the Eigenvalues of a matrix can be done on the HP-28S in a few minutes with three small programs. The first program, which is called M→CE in this manual, converts the matrix to a list containing the coefficients of the characteristic equation, and is provided in Appendix D. It is a modified version of two separate programs found in a math applications supplemental publication from Hewlett Packard [Ref 4].

The other two programs, called PF and QD, are included in the EE department's HP-28S book of engineering application programs [Ref 5], which may be purchased at the bookstore for $10. The PF program takes a list containing the coefficients of an nth order polynomial (such as the one returned by the M→CE program) and breaks it into quadratic and first order equations. (Note: To use the PF program, another program called P1 must also be installed. P1 is provided in the EE application booklet on the same page as PF.) The program QD finds the roots of these equations (real or complex), which are the eigenvalues originally sought.

Before entering these programs into the calculator, create a subdirectory for them. Following the suggestion in the EE booklet, the user may wish to name this directory POLY. The lines and block spacing shown in Appendix D are just to make

it easier to read while keying it in. When entering the program, key the commands in as if the entire program was on one long line. Put the keyboard in the Alpha mode before beginning.

NOTE: The user will find that this manual does not agree with the strategy for organizing the directory tree structure presented in the EE applications booklet, nor its approach to unit conversion. After fairly extensive comparative studies among aeronautical engineering students, the methods in this manual have been strongly favored. The **PF** (with **P1**) and **QD** programs, however, are useful and efficient.

When using these programs, the user is advised to store the matrix and intermediate result (the coefficients of the C.E.). This takes only a second and may save considerable time later, particularly when working with large matrices.

After installing these three programs in the **POLY** directory, the procedure for finding the eigenvalues of any matrix is relatively simple:

As always, start from the **PLAY** directory. Input the matrix on the stack. (Most users find it faster to use commas to separate entries in the matrix rather than using the space key.) Next hit ENTER to duplicate the matrix, then A STO (if the matrix is called 'A'). This leaves a copy of the matrix in the **PLAY** directory in case it is needed again.

Next, pop **DOWN** to **POLY** with the matrix on the stack, and press **M→CE**. The program takes the calculator about 20 seconds

for a 4x4 matrix, moves the current directory back up to **PLAY** and leaves the result on the stack. When the program is done, press ENTER to duplicate the characteristic equation coefficients on the stack, then CE STO, which saves the characteristic equation coefficients in the **PLAY** directory.

Finally, pop back **DOWN** to **POLY**, and run the **PF** program to factor the polynomial, then the **QD** program as many times as required to get the roots, or eigenvalues.

The program **M→CE** is presented in Appendix D.

XI.   CONTROLS PROGRAMS

Frequently the aeronautical engineer must work with Bode plots to determine aircraft stability. He may be given a 2X2 matrix, a quadratic characteristic equation, or a complex conjugate set of eigenvalues, any of which describe a single complex conjugate pair of poles.

Three programs are presented in Appendix E to assist the user in working these problems. The programs should be stored in a **CNTRL** subdirectory.

All three programs work the same way. If the user is given a 2X2 matrix, he enters the matrix on the stack and runs the **MTRX** program. Similarly, if he has been given a second order characteristic equation, he can place a list of the coefficients on the stack and invoke the **CEQN** program. If the actual poles are given, the complex coordinates of one of the poles (an eigenvector) is placed on the stack, and the **EIGN** program is called.

Each of the programs will compute the damped and undamped natural frequencies, the damping ratio, and period of the function. If a matrix is given as input, both the coefficients of the characteristic equation and one of the eigenvalues (coordinate of the pole position) is returned. For C.E. or eigenvalue (pole coordinate) inputs, either the eigenvalue or C.E. coefficients, respectively, are returned.

43

XII. A SOLUTION TO AN AERONAUTICAL ENGINEERING PROBLEM

A problem solved by Bertin-Smith [Ref. 6] involves computing the lift and moment coefficients for an NACA Four Series airfoil given the equations of its mean camber line. The calculations are somewhat difficult, as several definite integrals must be solved to obtain the A0, A1, and A2 Fourier coefficients, which are then used to find the CL and CM values.

The entire process for computing the Fourier coefficients can be programmed into the HP-28S, which can then find the "A" constants in less than a minute, depending on how fast the user can enter six numbers! With the Solver, the lift and moment coefficients can be computed afterward.

When it is run, the program will ask for the NACA series designation (i.e., 4412), which provides the position of maximum camber and thus, the transition point between the camber line equations. In addition, it will ask for the five coefficients (C1 → C5) describing the camber curve given by the formulae:

$$Z/C = C1*[C2*(X/C) - (X/C)^2] \qquad \text{(forward)}$$
$$Z/C = C3*[C4 + C5*(X/C) - (X/C)^2] \qquad \text{(aft)}$$

Entering the program in the calculator takes considerable effort, but running it once it has been installed is simple. Create an **AERO** subdirectory, and build the program there.

44

Store the program under the name **FOUR**. Be especially careful of upper and lower case labels!

To use the program **FOUR**, merely go into the **AERO** directory and press the **FOUR** softkey. An input menu will appear (it looks like a Solver menu) and the user may then enter the required airfoil information (C1 → C5 and the NACA series under the variable **FOIL**). Once the numbers have been entered, press CONT (for CONTinue) and wait. The display will show when each of the A(n) is computed.

Once the program terminates, use either of the equations below with the Solver to compute lift and moment coefficients.

CAUTION! α MUST BE ENTERED IN RADIANS.   CONVERT FROM DEGREES (use **D→R** in the TRIG menu) BEFORE ENTERING α VALUES IN THE SOLVER!

NOTE: The value stored as A0 in the calculator is not really A_0.   Actually, $A_0 = \alpha -$ (the A0 calculated).   The equations below account for this difference.

Here are the CL and CM4 equations:

'CofL = π * (2 * (alfa - A0) + A1)'   Store as 'CL

'Cmc4 = π / 4 * (A2 - A1)'   Store as 'CM4

The program FOUR appears in Appendix F. The lines and block spacing shown are just to make it easier to read while keying it in. When entering the program, key the commands in as if the whole thing was on one long line. Put the keyboard in the Alpha mode before beginning.

XIII. EPILOGUE

A. THE NEW HP-48SX

The HP-28S, with its impressive user RAM storage (32K) and softkey menu-driven function access, is already being eclipsed by the HP-48SX Scientific Handheld Calculator. The HP-48SX features a display area twice as large, more than twice the user memory, an improved operating environment for handling graphical objects, and an RS-232 compatible serial interface which allows programs to be transferred electronically between calculators or between the HP-48SX and a PC.

A large user's group for the HP-48SX exists and communicates on various computer bulletin board services. The Corvallis BBS, operated by the Hewlett-Packard factory in Corvallis, Oregon, provides hundreds of programs, utilities, and games which may be downloaded to interested HP-48SX users.

Programs included in this manual may not be compatible or even required for the HP-48SX, however any equations given herein may be saved and used with the Solver, just as in the HP-28S. HP-48SX owners will find the Gas Dynamics equations section of this manual of particular use, since the programs provided by Hewlett-Packard do not conform to the tables used in the Zucker text, which is commonly used in the thermodynamic courses taught at the Naval Postgraduate School.

Though the capability to upload programs into a calculator without having to key them in manually is desirable, the closable case design of the HP-28S and dual keyboard layout are advantages over the cluttered and cramped array of multiple-function keys on the HP-48SX. Owners of the HP-28S need not scrap this fine machine in order to upgrade to the newer HP-48SX, and unless the specific graphics-handling capabilities of the HP-48SX are required, the HP-28S should provide adequate handheld computing power for the discerning engineer for many years to come.

B. THE POSSIBILITY OF ERRORS IN THIS MANUAL

Extensive efforts have been made to validate and correct the equations, programs, and entry sequences presented in this manual. More than fifty students at the Naval Postgraduate School have been or are currently using the methods described in this guide. Each chapter was read and followed by a high-school graduate with no engineering background as a test for ease of understanding. Despite these precautions, the author recognizes minor errors may exist, and apologizes for any inconvenience caused thereby.

If the programs or equations fail to perform as suggested, the user should carefully verify his entries. A duplicated character, upper/lower case error, or missing operand are frequently to blame.

Global Constants (R, GAMA, etc.)
Global Programs (NU, SHHH, etc.)

Computation area. Kept clear except
for storage of temporary working
equations and variables.

Gateway to all user subdirectories.
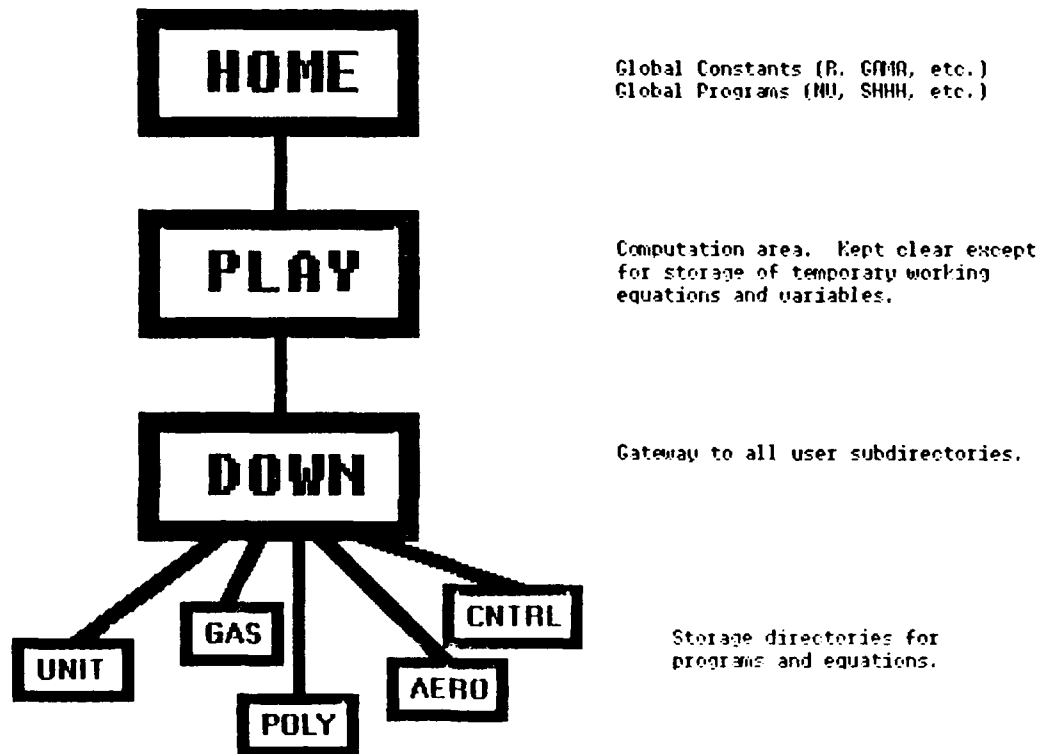
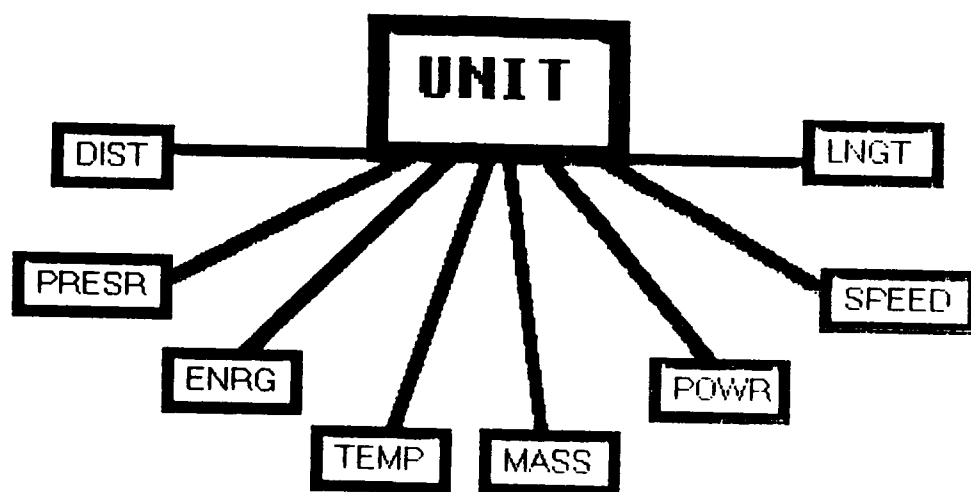Storage directories for
programs and equations.

Figure 1

Figure 2

APPENDIX B

Place the following in the **ENRG** subdirectory:

```
<< CONVERT DROP ENTER 'CONV STO
'J ENTER   '→J STO
'Btu ENTER   '→BTU STO
"ft*lbf ENTER   '→FTLB STO
'cal ENTER   '→CAL STO
<< PLAY ENTER   'UP STO
```

Place the following in the **PRESR** subdirectory:

```
<< CONVERT DROP ENTER 'CONV STO
"lbf/ft^2   ENTER   '→PSF STO
'Pa   ENTER   '→PASC STO
'psi   ENTER   '→PSI STO
'atm   ENTER   '→ATM STO
<< PLAY ENTER   'UP STO
```

Place the following in the **TEMP** subdirectory:

```
<< CONVERT DROP ENTER 'CONV STO
' R ENTER   '→ R STO
' K ENTER   '→ K STO
' F ENTER   '→ F STO
' C ENTER   '→ C STO
<< PLAY ENTER   'UP STO
```

Place the following in the **POWR** subdirectory:

```
<< CONVERT DROP ENTER 'CONV STO
"ft*lbf/s   ENTER   '→FLBS STO
'W ENTER   '→WAT STO
"Btu/h ENTER   '→BTH STO
'hp ENTER   '→HP' STO
<< PLAY ENTER   'UP STO
```

Place the following in the **MASS** subdirectory:

```
<< CONVERT DROP ENTER 'CONV STO
'slug ENTER   '→SLG STO
'lb ENTER    '→LBM STO
'g ENTER    '→GRM STO
'kg ENTER  '→KG STO
<< PLAY ENTER   'UP STO
```

Place the following in the **SPEED** subdirectory:

```
<< CONVERT DROP ENTER 'CONV STO
"ft/s" ENTER   '→FPS STO
'mph' ENTER   '→MPH STO
'kph' ENTER   '→KPH STO
'knot' ENTER   '→KTS STO
<< PLAY ENTER   'UP STO
```

Place the following in the **DIST** subdirectory:

```
<< CONVERT DROP ENTER 'CONV STO
'km ENTER   '→KM STO
'm ENTER   '→M STO
'mi ENTER   '→MI STO
'ft ENTER   '→FT STO
<< PLAY ENTER   'UP STO
```

Place the following in the **LNGT** subdirectory:

```
<< CONVERT DROP ENTER 'CONV STO
'm ENTER   '→M STO
'cm ENTER   '→CM STO
'ft ENTER   '→FT STO
'in ENTER   '→IN STO
<< PLAY ENTER   'UP STO
```

APPENDIX C

These equations belong in the subdirectory **ISEN**:

'p / pt = (1 / (1 + (GAMA-1) / 2 * M^2)) ^ (GAMA / (GAMA-1))'

(Save as 'PPT')

'T / Tt = 1 / (1 + (GAMA-1) / 2 * M^2)'

(Save as 'TTT')

'A / ASTAR = 1 / M * ((1 + (GAMA-1) / 2 * M^2) / ((GAMA+1)
/ 2)) ^ ((GAMA+1) / (2 * (GAMA-1)))'

(Save as 'ASTAR')

'p * A / (pt * ASTAR) = (1 / (1 + (GAMA-1) / 2 * M^2))
^ (GAMA / (GAMA-1)) * (1 / M * ((1 + (GAMA-1) / 2 * M^2)
/ ((GAMA+1) /2)) ^ ((GAMA+1) / (2 * (GAMA-1))))'

(Save as 'PAPTA')

'Nu = ((GAMA+1) / (GAMA-1)) ^ (1/2) * ATAN (((GAMA-1) /
(GAMA+1) * (M^2 - 1)) ^ (1/2)) - ATAN((M^2 - 1) ^ (1/2))'

(Save as 'nu')

'SIN(Mu) = 1 / M'

(Save as 'MU')

'A2 / A1 = M1 / M2 * ((1 + (GAMA-1) / 2 * M2^2) / (1 +
(GAMA-1) / 2 * M1^2)) ^ ((GAMA+1) / (2 * (GAMA-1)))'

(Save as 'A2A1')

'Rho / Rhot = (1 / (1 + (GAMA-1) / 2 * M^2))
^ (1 / (GAMA-1))'

(Save as 'RHOT')

52

These equations belong in the subdirectory **SHOCK**:

'M2^2 = (M1^2 + 2 / (GAMA-1)) / (2 * GAMA / (GAMA-1)
* M1^2 -1)'

(Save as '**M2**')

'p2 / p1 = 2 * GAMA * M1^2 / (GAMA+1) - (GAMA-1) / (GAMA+1)'

(Save as '**p2p1**')

'T2 / T1 = (1 + (GAMA-1) / 2 * M1^2) * (2 * GAMA / (GAMA-1)
* M1^2 - 1) / ((GAMA+1) ^ 2 / (2 * (GAMA-1)) * M1^2)'

(Save as '**T2T1**')

'(V1 - V2) / a1 = 2 / (GAMA+1) * ((M1^2 - 1) / M1)'

(Save as '**DVa**')

'pt2 / pt1 = ((GAMA+1) / 2 * M1^2 / (1 + (GAMA-1) / 2 * M1^2))
^ (GAMA / (GAMA-1)) * (2 * GAMA / (GAMA+1) * M1^2 - (GAMA-1)
/ (GAMA+1)) ^ (1 / (1-GAMA))'

(Save as '**pt21**')

'pt2 / p1 = ((GAMA+1) / 2 * M1^2 / (1 + (GAMA-1) / 2
* M1^2)) ^ (GAMA / (GAMA-1)) * (2 * GAMA / (GAMA+1) *

M1^2 - (GAMA-1) / (GAMA+1)) ^ (1 / (1-GAMA)) *
(1 + (GAMA -1) / 2 * M1^2) ^ (GAMA / (GAMA-1))'

(Save as '**pt2p1**')

'TAN(DELTA) = 2 / TAN(THETA) * ((M1^2 * SIN(THETA)^2 - 1)
/ (M1^2 * (GAMA + COS(2 * THETA)) + 2))'

(Save as '**Delta**'

'rho2 / rho1 = (GAMA+1) * M1^2 / ((GAMA-1) * M1^2 + 2)'

(Save as '**RHO21**')

'SIN(Mu) = 1 / M'   (Save as '**MU**')

53

These equations belong in the subdirectory **FANO**:

'T / TSTAR = (GAMA+1) / 2 / (1 + (GAMA-1) / 2 * M^2)'
                                              (Save as '**TSTAR**')

' p / pstar = INV(M) * ((GAMA+1) / 2 / (1 + (GAMA-1) /
 2 * M^2)) ^ (1/2)'
                                              (Save as '**PSTAR**')

'pt / ptstr = INV(M) * ((1 + (GAMA-1) / 2 * M^2) / ((GAMA+1)
 / 2)) ^ ((GAMA+1) / (2 * (GAMA-1)))'
                                              (Save as '**PTSTR**')

'V / Vstar = M * ((1 + (GAMA-1) / 2 * M^2) / ((GAMA+1) / 2))
 ^ (-1/2)'
                                              (Save as '**VSTAR**')

'fLmx / De = (GAMA+1) / (2 * GAMA) * LN((GAMA+1) / 2 * M^2
 / (1 + (GAMA-1) / 2 * M^2)) + INV(GAMA) * (INV(M^2) - 1)'
                                              (Save as '**FLDE**')

'dels / R = LN(1 / M * ((1 + (GAMA-1) / 2 * M^2) / (1 +
 (GAMA-1) / 2)) ^ ((GAMA+1) / (2 * (GAMA-1)))))'
                                              (Save as '**SMXR**')

54

These equations belong in the subdirectory **RAY**:

'Tt / TtSTR = 2 * (1+GAMA) * M^2 / (1 + GAMA * M^2) ^ 2
* (1 + (GAMA-1) / 2 * M^2'

(Save as **'TTSTR'**)

'T / TSTAR = M^2 * (1+GAMA) ^ 2 / (1 + GAMA * M^2) ^ 2'

(Save as **'TSTAR'**)

'p / pstar = (1+GAMA) / (1 + GAMA * M^2)'

(Save as **'PSTAR'**)

'pt / ptstr = (1+GAMA) / (1 + GAMA * M^2) * ((1 + (GAMA-1) /
2 * M^2) / ((GAMA+1) / 2)) ^ (GAMA / (GAMA-1))'

(Save as **'PTSTR'**)

'V / Vstar = (1+GAMA) * M^2 / (1 + GAMA * M^2)'

(Save as **'VSTAR'**)

'dels / R = LN((1 / M) ^ (2 * GAMA / (GAMA-1)) * ((1 + GAMA *
M^2) / (1+GAMA)) ^ ((GAMA+1) / (GAMA-1)))'

(Save as **'SMXR'**)

APPENDIX D

This is the program 'M→CE':

```
<< PLAY DUP SIZE 1 GET                      #STACK #ARRAY
→ g n << g 'tmp' STO { } 1 n

START 0 1 n FOR i tmp i DUP 2               #BRANCH #STACK
→LIST GET +                                 #LIST #ARRAY

NEXT 1 →LIST + 'tmp' g                       #BRANCH #LIST
STO* NEXT 'tmp' PURGE >>                     #STORE #BRANCH

DUP SIZE  → b n << {1} 1 n                    #STACK #ARRAY
FOR i → s << 0 1 i FOR j b j                 #BRANCH

GET s i j - 1 + GET * - NEXT                 #ARRAY #BRANCH
i / 1 →LIST s SWAP + >> NEXT  >>  >>         #LIST #BRANCH
```

56

APPENDIX E

This is the program **'MTRX'**:

```
<< → a << PLAY a 1 GET a 4 GET DUP2 +        #ARRAY #STACK
NEG ROT ROT * a 2                            #COMPLX #STACK
GET NEG a 3 GET NEG                           #ARRAY #COMPLX

* - → b c << 1 b c 3 →LIST 'CE' STO          #LIST
C √ DUP DUP 'Wn' STO b 2 / SWAP / DUP        #STACK
'Zeta' STO 1 SWAP SQ - √ * DUP 'Wd' STO      #STACK

2 π * SWAP / 'Td' STO Zeta Wn * NEG          #COMPLX
Wd R→C 'Eig' STO >> >> >>                     #COMPLX
```

This is the program **'CEQN'**:

```
<< LIST→ DROP → b c << PLAY c √              #LIST
DUP DUP 'Wn' STO b 2 / SWAP /                 #STACK
DUP 'Zeta' STO 1 SWAP SQ - √ *               #STACK

DUP 'Wd' STO 2 π * SWAP /                     #STACK
'Td' STO Zeta Wn * NEG Wd R→C                 #COMPLX
'Eig' STO DROP >> >>
```

This is the program **'EIGN'**:

```
<< → a << PLAY a DUP R→P C→R DROP            #STACK #COMPLX
DUP 'Wn' STO SWAP C→R DUP 'Wd' STO           #STACK #COMPLX

2 π * SWAP / 'Td' STO NEG SWAP /             #COMPLX
'Zeta' STO 1 2 Zeta * Wn * Wn
SQ 3 →LIST 'CE' STO >> >>                     #LIST
```

NOTE: If the user did not install the **NU** program in the HOME directory, change the first line of this program to read: 35 CF 36 SF STD RAD.

This is the program 'FOUR':

Program	Annotation
<< PLAY NU STD RAD	#MODE #MODE
{STO FOIL C1 C2 C3 C4 C5} MENU HALT	#MEMORY #CONTRL
6 FIX 1 FOIL 100 / IP 10 / FP 2 * -	#MODE #REAL #REAL
ACOS 'LIM' STO	#TRIG
C1 'I1' STO C2 1 - C1 * 'I2' STO	
C3 'I3' STO C5 1 - C3 * 'I4' STO	
CLLCD	#CONTRL
<< COS I1 * I2 + >> {0 LIM} 1E-7 ∫ DROP	#TRIG
<< COS I3 * I4 + >> {LIM π} 1E-7 ∫ DROP	#TRIG
+ π / 'A0' STO "A0 = alfa - " A0	
→STR + 1 DISP	#STRING #STRING
<< DUP COS 2 ^ I1 * SWAP COS I2 * + >>	#STACK #TRIG #TRIG
{0 LIM} 1E-7 ∫ DROP	
<< DUP COS 2 ^ I3 * SWAP COS I4 * + >>	#STACK #TRIG #TRIG
{LIM π} 1E-7 ∫ DROP	
+ 2 * π / 'A1' STO "A1 = " A1	
→STR + 2 DISP	#STRING #STRING
<< DUP DUP 2 * COS SWAP COS *	
I1 * SWAP 2 * COS I2 * + >>	#STACK #TRIG
{0 LIM} 1E-7 ∫ DROP	
<< DUP DUP 2 * COS SWAP COS *	
I3 * SWAP 2 * COS I4 * + >>	#STACK #TRIG
{LIM π} 1E-7 ∫ DROP	
+ 2 * π / 'A2' STO "A2 = " A2	
→STR + 3 DISP	#STRING #STRING
{I4 I3 I2 I1 LIM C5 C4 C3 C2 C1 FOIL}	
PURGE { } MENU 23 MENU CLMF>>	#MEMORY #CONTRL

LIST OF REFERENCES

1. Hewlett-Packard, *Getting Started with the HP-28S*, Hewlett-Packard Company, 1989.

2. Hewlett-Packard, *Technical Reference Manual for the HP-28S*, Hewlett-Packard Company, 1989.

3. Zucker, R. D., *Fundamentals of Gas Dynamics*, Matrix Publishers, Inc., 1977.

4. Hewlett-Packard, *Vectors and Matrices Applications Manual*, pp. 54-56, Hewlett-Packard Company, 1988.

5. IEEE Student Branch, *Engineering Application Programs for the HP-28S*, pp.27,31, US Naval Postgraduate School, 1989.

6. Bertin, J. J. and Smith, M. L., *Aerodynamics for Engineers*, pp. 217-220, Prentice Hall, 1989.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center   2
   Cameron Station
   Alexandria, Virginia, 22304-6145

2. Library, Code 52   2
   Naval Postgraduate School
   Monterey, California 93943-5002

3. CDR Duym, Code 31   10
   Naval Postgraduate School
   Monterey, California 93943-5000

4. Professor Biblarz, Code 31   1
   Naval Postgraduate School
   Monterey, California 93943-5000

5. Professor Schmidt, Code 31   1
   Naval Postgraduate School
   Monterey, California 93943-5000

6. LCDR Harry F. Molyneux   2
   P.O. Box 10059
   Alexandria, Virginia 22310