**General Disclaimer**

**One or more of the Following Statements may affect this Document**

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)

DOE/NASA TECHNICAL
MEMORANDUM

DOE/NASA TM-78291

SOLAR SITE TEST MODULE

By Ralph R. Kissel and Donald R. Scott
Electronics and Control Laboratory

# U.S. Department of Energy

Solar Energy

# TABLE OF CONTENTS

# LIST OF ILLUSTRATIONS

# LIST OF TABLES

# SOLAR SITE TEST MODULE

## INTRODUCTION

The Department of Energy (DOE), in conjunction with the National Aeronautics and Space Administration/Marshall Space Flight Center (NASA/MSFC), supports numerous commercial solar heating and cooling system installations throughout the country. These systems are to demonstrate the use of solar energy and are referred to as demonstration solar sites or simply as solar sites.

Solar sites may be divided into two categories, instrumented and uninstrumented. An instrumented site has an abundance of sensors mounted in strategic places for the sole purpose of gathering data from that site. The analog output of these sensors is routed to a central location called a J-box, then into a box called a site data acquisition system (SDAS) where the data are converted to digital form. These digital data are then stored on a tape and once a day transmitted via telephone to a central location for compilation and analysis. These sites permit convenient access to the sensor data through a connector on the side of the SDAS. This report describes a troubleshooting tool, an AIM 65 microcomputer, that interfaces with that connector to read all the sensor information and perform calculations as required. Uninstrumented sites have neither an SDAS nor the sensors to use with it.

A solar site troubleshooting team needs immediate information about the site they are checking. Problems can be found and corrected without delay. The test module described here serves to collect that information and present it immediately in calibrated engineering units. Remote data are not available until at least the next day.

Older IBM units, the DCM (digital count monitor) and OSM (on-site monitor), were deficient in that they were not easily portable (especially OSM), usually had no printer, produced only decimal counts (DCM), or else did not produce a periodic output greater than 2 sec. These units operate only with instrumented sites. The test module described here presently has that limitation also.

The test module is basically the Rockwell AIM 65 microcomputer adapted to this application. It will manually read one sensor, periodically read up to 10 sensors, or periodically read all sensors. Performance data are assigned a channel number and treated like any sensor channel. Results can be printed whenever a hard copy is needed.

The test module package (Figure 1) consists of the computer and carrying case, the cassette recorder/player, the cable between the computer, SDAS and recorder, and a cassette tape containing the program and site data.
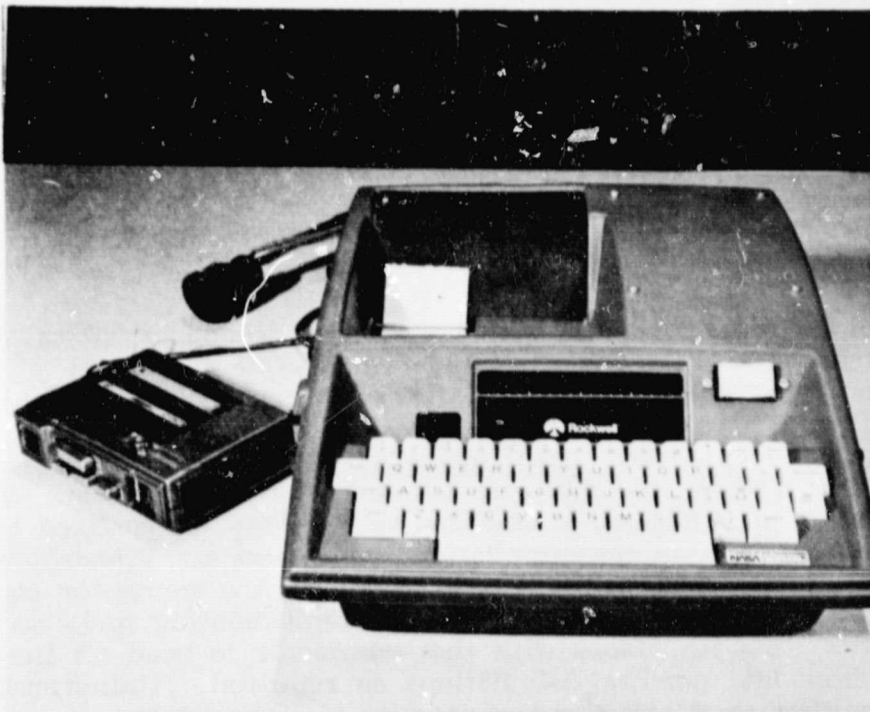


Figure 1. Test Module Package.

## SDAS INTERFACE

All sites where this test module is used must have an SDAS installed there. A fundamental design requirement was to know what all signals entering and leaving the test plug were and exactly how each was related to the other. Some of this information was not available from IBM and had to be measured. A 16-channel logic analyzer was used to do this. Appendix A contains details concerning interface signals and cable. Using this information and knowing that the SDAS hardware interface is TTL compatible indicated that the AIM 65 hardware would be compatible and allowed an assembler software routine to be written to interrogate the SDAS. Appendix C contains details of this 6502 software routine.

One problem encountered was that the strobe from the SDAS which signaled that the data was stable on the data lines was too short to activate reliably the interface device on the AIM 65. A pulse stretcher circuit was added which increased the strobe width from 200 to 700 ns and proper operation was achieved. Appendix B contains details on the pulse stretcher circuit.

2

# TEST MODULE

The AIM 65 has many features that make it well suited to this task. It is small enough to be hand-carried onboard a plane. The unit already built into a small carrying case is now available. Future units purchased for this task should be ordered this way.

It has a 20-column printer for permanent data records and has a 20-column display which is needed for user-prompt messages. It can store programs and data on most any cassette recorder. It uses the BASIC language which was designed for ease of man-machine communication. It also can use assembler language which allows the high speed necessary for communication with the SDAS. It has a full size keyboard which, with BASIC, allows site data to be stored or modified at the site if necessary. It includes 4K memory (RAM) which is adequate for this task. It costs approximately $600.

The particular recorder currently being used is one that was already available to the solar program. It cost approximately $40. When used for recording, the tape gap between records had to be increased somewhat from the default value in the AIM 65 monitor, but this is not unusual. Other recorders may or may not need this change. Recorder connections depend heavily upon the particular recorder being used and are described in the AIM 65 literature.

The cable must match the SDAS on one end and have at least a 44-pin printed circuit board connector on the other. The cable now being used is similar to that already being used with the SDAS. Future cables to the test module could be smaller in diameter and/or longer with an enclosed connector on the test module end. The recorder connections come off the same 44-pin connector as the SDAS lines. There are only three plugs to the recorder: audio in, audio out, and power (which is remotely controlled). Good quality tape is preferred and a backup copy is maintained in any case.

## PROGRAM FEATURES

The software package is written entirely in BASIC as one program. It includes the machine language program (to read the SDAS) as an integral part. It automatically checks for the proper number of parameters in the site description data. It is loaded from tape in one operation by BASIC.

It gives the user three main options: (1) Manually look at any channel, (2) periodically look at one to ten channels, or (3) periodically (or one time) look at all channels. The output can be printed or not as desired. Naturally, printout would be used for automatic, unattended operation. See Appendix D for the BASIC program description and Appendix F for program operating instructions.

3

The SDAS read portion always accepts an entire scan of data as fast as it comes from the SDAS. Then any processing and printout are done using this one simultaneous data set no matter how long it takes to actually print the results. In case the SDAS is busy when a data set is requested, the SDAS is rechecked every 2 sec until a data set is received or until the computer is interrupted. If a wire came loose, for example, the computer would print TIMEOUT and never do anything more until the problem was solved.

A portion of the program is used only for site data as taken from the IP (instrumentation program) for that site. This is normally set up on tape ahead of time as it will generally require about 4 hr to do. Once on site, channels (sensor or performance) may be updated, added or deleted as needed and new results obtained immediately.

It is possible to store data on tape for later use, although it is not now an automatic feature. Full, automatic printout should reduce the need for any such data storage virtually to zero, especially in the present troubleshooting application.

Results are given to 0.1°F for temperature and to approximately 0.1 percent for all other outputs. Outputs will be given to 0.1 units as a standard format.

## SITE DATA PREPARATION

The primary item needed to prepare data for the computer is the site IP; from it, all sensor channels with their calibration coefficients and units are available. Instantaneous performance parameters can also be defined using the IP.

Sensor channels are defined to the computer by assigning a program line number from 1000 to 1999 for each sensor, then giving the channel number, the calibration coefficients and the units code on that same line. In the case of a temperature sensor, a fixed scaling is used to save memory (no decimal point and no exponents needed).

Instantaneous performance parameters are also assigned a channel number and the data to the computer then consist of the temperature and flow channel numbers with the proper units code. For liquid flow, a fourth channel is specified which contains the specific heat and density coefficients. For insolation, the collector area is needed and for air systems, the duct area is needed also.

Once all the channels are specified, only a few minutes are needed to enter them into the computer. Running the program will automatically do considerable error checking, particularly as to whether the proper number of parameters was used for each channel. Once all is in order,

the complete program with site data can be recorded on tape for use at the site. Changes at the site can be entered just as in the initial preparation, and they can be either temporary or permanent changes depending on whether or not they are stored on tape. See Appendix E for further details on site data format.

A tape log is always needed to show what programs (or data) are on a particular tape and where on the tape they are located (from tape counter, etc.).

## OTHER CONSIDERATIONS

The present 4K user memory will handle the complete primary program and approximately 60 channels of data. Splitting the program into two parts (i.e., two tape loads) will allow about 25 more channels to be added. Each 1K memory allows about 40 channels to be added. Memory can be added from 8K more to 32K more by simply buying it. The processor can address up to 64K bytes easily.

AIM 65 speed should always be more than enough for any solar energy application. It is capable of lengthy performance calculations if the need arises. In the present application, there is some delay (a few seconds) in searching out some required channels. This delay could be reduced just to that required by the printer if another 2K of memory were available.

Portability, even for the older size unit, has already been demonstrated. The unit has been put inside a large carry-on briefcase and fits under the airplane seat. A portable unit with built-on handle is now available from at least one manufacturer and is even smaller than what has been used here. The complete AIM 65 with enclosure and power supply weighs approximately 10 lb and approximately half of that is power supply weight.

Many accessory and expansion boards are available for the AIM 65, and others are continually being introduced. There are larger memories, video boards, PROM programmers, DISK system, more I/O, etc. 32K memory can be purchased for less than $400.

## SUMMARY

The AIM 65 solar site checkout test module is a readily available, low-cost, portable unit well suited for solar site measurements. The 4K memory version, with some compromises, is adequate to use at all instrumented sites. It is expandable in capability, should the need arise, to

do other tasks such as full-scale performance analysis and automatic real-time site control. A data acquisition module could be added, probably inside the present module, to check out uninstrumented sites once a way to get data to the module is worked out. Either way, the module will monitor one or more channels at fixed time intervals and print out the results, if desired. It can be reprogrammed on-site to accommodate system parameters that are found to be different than expected.

# APPENDIX A

## SDAS - AIM 65 CABLE CONNECTIONS AND
## INTERFACE SIGNAL DESCRIPTION

The following table shows the signal line descriptions and their pin numbers on both ends of the cable.

### TABLE A-1.  CABLE CONNECTIONS

| SDAS SIGNAL | OSM PIN | AIM 65 J1 PIN | 6522 |
|---|---|---|---|
| STROBE | 13 | 20 | CA1 |
| STROBE | – | 18 | CB1 |
| INTR. REQUEST | 11 | 21 | CA2 |
| D7 | 29 | 14 | A∅ |
| D6 | 28 | 4 | A1 |
| D5 | 27 | 3 | A2 |
| D4 | 26 | 2 | A3 |
| D3 | 25 | 5 | A4 |
| D2 | 24 | 6 | A5 |
| D1 | 23 | 7 | A6 |
| D∅ | 22 | 8 | A7 |
| INTR. POLL | 10 | 13 | B4 |
| FCT 0 | 14 | 16 | B5 |
| CARD SEL. | 9 | 17 | B6 |
| FCT 1 | 15 | 15 | B7 |
| CONNECTED | 32 | K | – |
| GROUND | 33 | 1 | – |
| SEL. ACK. | 12 | 17 | B6 |

Select Acknowledge was connected directly to Card Select on the AIM 65 end since a very fast response was required (<.75 μs). Pins 18 and 20 on AIM 65 J1 are connected so that both interrupt inputs see the same strobe signal. The OSM connected (5 mA) and signal return lines from the SDAS are connected to ground on the AIM 65 J1 connector. So anytime the cable is plugged into the SDAS, all that is required to interrogate the SDAS is an active low on the Interrupt Request line. With this wiring choice, the only output from the AIM 65 itself is on the Interrupt Request line. The Power-On and Select-Acknowledge signals to the SDAS are hardwired. All other signal lines are inputs to the AIM 65. Pin K has no AIM 65 connection and was used as a ground tiepoint.

Recorder connections depend on the recorder used and must be worked out according to the AIM 65 instructions. The recorder used here was negative ground with plug shells being ground. The connections to AIM 65 J1 for remote control of this one recorder (T1) are as follows:

TABLE A-2. RECORDER CONNECTIONS

| FUNCTION | J1 PIN |
|---|---|
| POWER (Remote) | E |
| EARPHONE | L |
| MICROPHONE | M |
| +5V to TRANSISTOR | A to F (JUMPER) |
| SHELLS (GND) | 1 or K |

Figure A-1 shows the SDAS detailed signal and timing chart as measured on a Tektronix logic analyzer. IR initiates the data scan, IP and CS are SDAS replies and SA is the required reply back to the SDAS. IR is raised in response to the strobe from the SDAS.

The four bytes for the RTC are then sent and strobed in as shown. CS is used to generate SA. Note that the second data byte of each pair is on the line about 3 μs compared to the first byte time of around 570 μs. The strobe, however, comes 1.2 μs after the data is on the line in all cases.

There is a long wait before the first channel is sent while the offset is being determined. After the first channel (offset) is sent, all the other channels are sent in order. F1 is active while the last existing channel is being sent. F1 is therefore used to signal that there will be no more data.
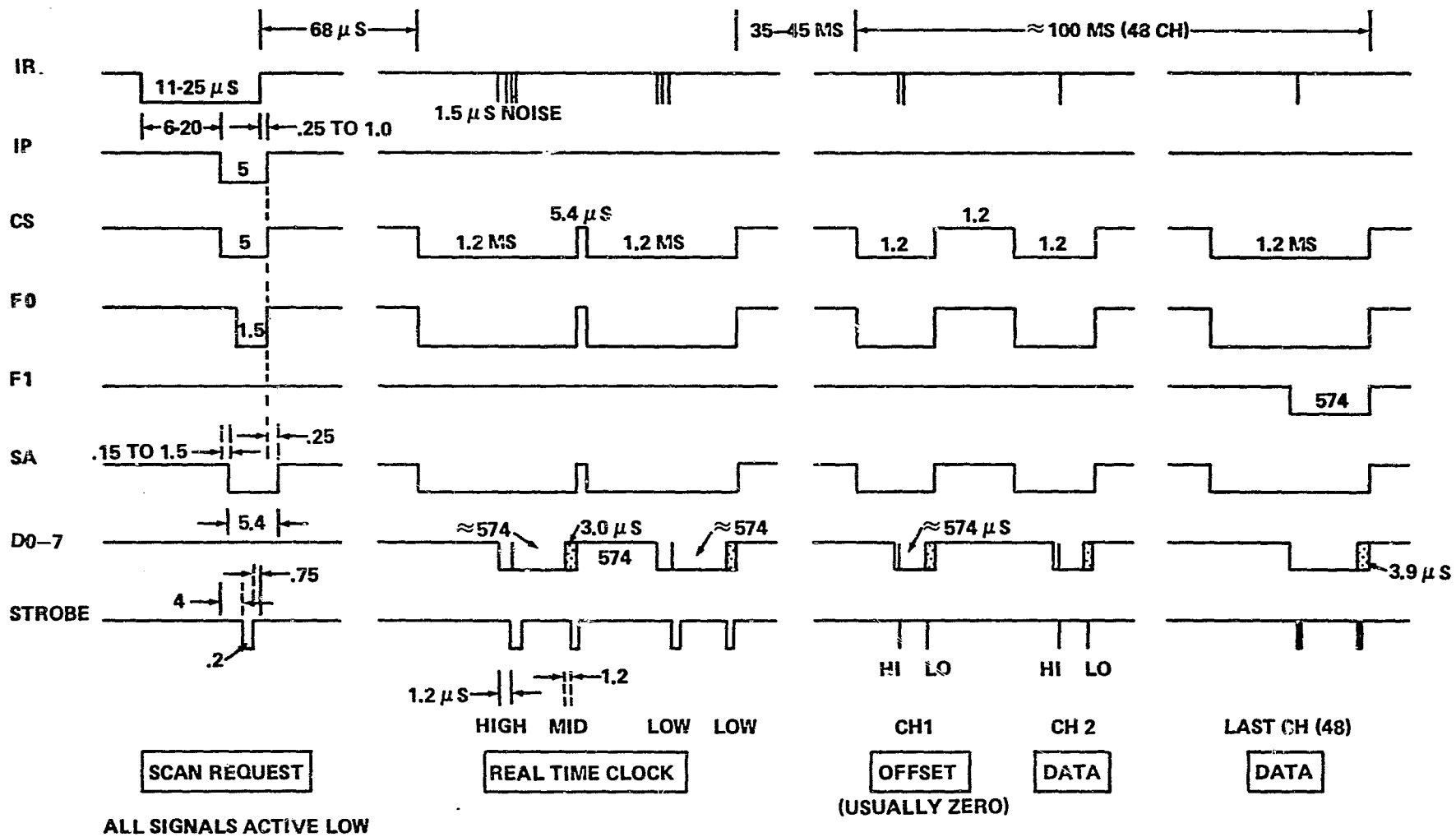
8

Figure A-1. SDAS signals and timing.

# APPENDIX B

## PULSE STRETCHER CIRCUIT

It was intended that the AIM 65 would be directly interfaced to the SDAS without modification. It turned out that the 200 ns strobe from the SDAS is too short to permit reliable operation of the 6522 interface IC in the AIM 65. It was therefore necessary to add a monostable flip-flop in series with the strobe line to stretch the strobe (as seen by the 6522) to 700 ns. The circuit shown in Figure B-1 does this, but it could be implemented in several other ways. The 54123 was used because it was on hand (dual monostable).

Figure B-1. Pulse stretcher.
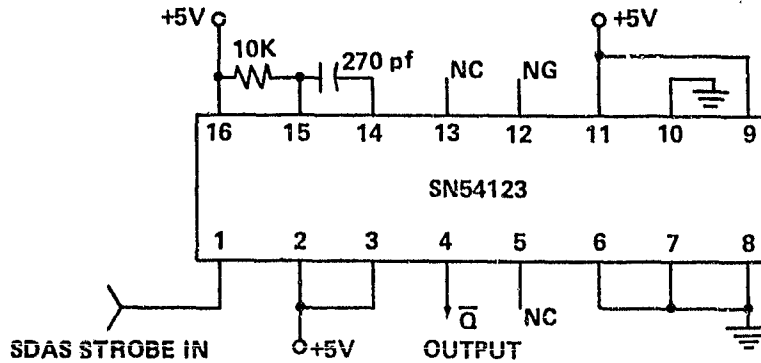
The $\bar{Q}$ pulse stretcher output was wired to pins 18 and 40 on the 6522 socket for convenience rather than to connector J1. The +5V was taken off J1 pin A and ground was J1 pin 1. The strobe is taken off either pin 18 or 20 on J1. Pins 18 and 40 on the 6522 were lifted out of their sockets and soldering was then done directly to them. This avoided cutting lines on the PC board.

# APPENDIX C

## 6502 ASSEMBLER PROGRAM TO INTERROGATE SDAS

The AIM 65 was programmed in machine language to read the SDAS because BASIC would be far too slow. The machine language program is called by BASIC when a scan is needed. The program initiates the scan request, uses the first strobe to recognize that the SDAS replied and uses all later strobes to latch in the data bytes.

The 6522 interface chip is set up to latch data whenever a strobe is seen and to latch it on the strobe trailing edge. Noise on the data lines was sometimes seen on the strobe leading edge.

As soon as the IR line is lowered, the processor begins looking for the strobe reply. As soon as it is seen, it raises IR, starts a 64 ms timer and begins looking for CS and F0 to become active. If no first strobe is seen after looking 255 times ($\approx 3.6$ ms) the program raises IR and goes back to BASIC with zero as the number of strobes seen. BASIC uses this to inform the user that the timeout was caused by no reply from the SDAS.

If CS and F0 are not seen and a timeout occurs, the program goes back to BASIC with 200 as the number of strobes seen. This informs the user that the timeout was not caused by no reply from the SDAS but by noise, a hardware glitch, malfunctioning SDAS, etc.

Once CS and F0 are seen, a search is started for the first of two strobes. If no strobe is seen before 64 ms are up, 200 is again sent back to BASIC. When the strobe is seen, the data are read, inverted to active high, and stored in the next sequential location in memory. After the second of the strobe pair is processed, a check is made to see if F1 was active along with CS and F0. If it was, the program goes back to BASIC since the scan is done. If it was not, the program gets the next two bytes.

If a strobe is missed somewhere along the line, a timeout will occur with 200 returned indicating a problem. An extra strobe will also cause this. An even number missed will not timeout, but will only give a channel count that is wrong and the data will probably make little sense, since at least some of it will be in the wrong place in memory. Strobes will not be missed when the equipment is working properly.

CS and F0 are checked, rather than just relying on strobes alone because false strobes (or noise) were seen at times when CS and F0 were not active during the scan. F0 is used with CS for better reliability, although CS should be sufficient by itself.

The SDAS transmits an error word (FF, FF) if for some reason it does not see SA. Since SA is hardwired, no specific provision is made for that case although the data would reflect this in the event that a timeout did not occur in the assembly program.

It should be noted that, as is visible in Appendix A, the IBM MSB is $\emptyset$ while for the AIM 65, the MSB is 7. This is taken care of in the cable wiring.

Further details can be obtained from the Assembler program listing which follows in Figure C 1.

```
        *=$0FSB  ;PGM START ADR
    LOC1 =$0EC7  ;DATA START ADR
    UDRB =$A000
    UDRAH=$A001
    UDDRB=$A002
    UDDRA=$A003
    UACR =$A00B
    UPCR =$A00C
    UIFR =$A00D
    UT2H =$A009
      LDA UIFR
      STA UIFR      ;CLR INTR'S
      LDA #0
      STA UDDRA     ;A ALL IN
      STA UDDRB     ;B ALL IN
      LDA #03
      STA UACR      ;ENABLE A,B LATCHES
      LDA #$1D
      STA UPCR      ;CA2(IR) LOW,INTR ON TRAIL EDGE
      LDX #0        ;DELAY CNTR
      LDY #0        ;STROBE CNTR
LOOP1 INX
      CPX #0        ;DELAYED 255?
      BEQ TIME      ;YES-SDAS BUSY
      LDA #$12      ;STROBE MASK
      CMP UIFR      ;STROBE YET?
      BNE LOOP1     ;NO,WAIT
      LDA #$1F
      STA UPCR      ;IR HIGH
LOOP3 LDA #$FF      ;.064 SEC
      STA UT2H      ;START TIMER
LOOP2 LDA UDRB
      AND #$60      ;F0,CS MASK
      BEQ SPOT1     ;OFF,WAIT
      LDA UIFR
      AND #$20      ;CK TIMEOUT
      BEQ LOOP2     ;NONE,CONT
      BNE HDW       ;PROBLEM
SPOT1 LDX #02       ;GET PAIR STROBES
LOOP5 LDA #$12
      STA UIFR      ;CLR IRA,B-NOT T2
LOOP4 LDA UIFR
      CMP #$12
      BEQ SPOT2     ;STR SEEN
      CMP #$20      ;CK T2
      BNE LOOP4     ;NONE,WAIT
  HDW LDY #200      ;200 TO BASIC
      BNE TIME      ;JMP
SPOT2 LDA UDRAH     ;LOAD DATA,CLR IRA
      EOR #$FF      ;TO ACTIVE HIGH
      STA LOC1,Y    ;STORE NEXT
      INY
      DEX           ;B LATCH ON 2ND STROBE
      BNE LOOP5     ;GET 2ND STR
      LDA UDRB      ;CLR IRB
      AND #$E0      ;CS,F0,F1 ALL ACTIVE?
      BNE LOOP3     ;NO,MORE DATA
 TIME LDA #$1F
      STA UP        ;IR  HIGH
      LDA #0        ;FOR BASIC
      JMP $C0D1     ;BACK TO BASIC
      .END
```

Figure C-1. Assembler listing.

```
OF8B  AD  LDA  A00D
OF8E  8D  STA  A00D
OF91  A9  LDA  #00
OF93  8D  STA  A003
OF96  8D  STA  A002
OF99  A9  LDA  #03
OF9B  8D  STA  A00F
OF9E  A9  LDA  #1D
OFA0  8D  STA  A00C
OFA3  A2  LDX  #00
OFA5  A0  LDY  #00
OFA7  E8  INX
OFA8  E0  CPX  #00
OFAA  F0  BEQ  OFF6
OFAC  A9  LDA  #12
OFAE  CD  CMP  A00D
OFB1  D0  BNE  OFA7
OFB3  A9  LDA  #1F
OFB5  8D  STA  A00C
OFB8  A9  LDA  #FF
OFBA  8D  STA  A002
OFBD  AD  LDA  A00D
OFC0  29  AND  #40
OFC2  F0  BEQ  OFCD
OFC4  AD  LDA  A00D
OFC7  29  AND  #20
OFC9  F0  BEQ  OFBD
OFCB  D0  BNE  OFDF
OFCD  A2  LDX  #02
OFCF  A9  LDA  #12
OFD1  8D  STA  A00D
OFD4  AD  LDA  A00D
OFD7  C9  CMP  #12
OFD9  F0  BEQ  OFE3
OFDB  C9  CMP  #20
OFDD  D0  BNE  OFD4
OFDF  A0  LDY  #C8
OFE1  D0  BNE  OFF6
OFE3  AD  LDA  A001
OFE6  49  EOR  #FF
OFE8  99  STA  OF07,Y
OFEB  C8  INY
OFEC  CA  DEX
OFED  D0  BNE  OFCF
OFEF  AD  LDA  A000
OFF2  29  AND  #E0
OFF4  D0  BNE  OFF8
OFF6  A9  LDA  #1F
OFF8  8D  STA  A00C
OFFB  A9  LDA  #00
OFFD  4C  JMP  C0D1
```

END

Figure C-1.   (Concluded).

# APPENDIX D

## BASIC PROGRAM DESCRIPTION

This program contains the software necessary to interrogate the SDAS, store each byte of data from the SDAS, and process and print the data in engineering units. It can do this for various combinations of channels or for all channels at time intervals of 2 sec or more and do it unattended if desired.

The software necessary to interrogate the SDAS was first written in assembler language then converted into decimal so BASIC could load it into the proper memory locations. This assembly language program is then called by BASIC when needed. It is located at statement numbers 2003 through 2009. The data are automatically stored from $0EC7_{16}$ ($3783_{10}$) through $0F8A_{16}$ ($3978_{10}$) and the program goes from $0F8B_{16}$ ($3979_{10}$) through $0FFF_{16}$ ($4095_{10}$).

On the way to loading the assembly program, all the channels are counted and this sum used later. If some numbers are left out or added (wrong number of parameters), an error will be displayed. If one number is left out before any channel 400-499 occurs, this will be missed so if the 400 channel group is put first this condition will be checked also. It is also possible to have one left out and one added somewhere else and not be detected. Careful data entry is always recommended.

The software has been designed to interrogate the SDAS no more than once every 2 sec. The situation arises as to what to do if the SDAS is busy or if some other problem occurs. One thought is to keep looking (in a loop) until a response is obtained. This, however, would occasionally cause a nonsense set of outputs because the interrupt request timed out just after the SDAS saw it and on the next busy check the AIM 65 saw a strobe in the middle of the data stream and assumed it was the first strobe. A minimum delay of about 300 ms is needed to avoid this and so 2 sec was always used. When the SDAS is interrogated and no response is received during one of the interval modes, the TIMEOUT message is printed and then the SDAS is checked every 2 sec until it once again responds. The same interval mode then continues as before. This would be needed in an unattended interval mode operation. The only change is that one interval is longer by the amount of time the SDAS did not respond and later intervals are then slipped in time by this amount. There is presently no absolute time interrupt scheme being used by the AIM 65. None is needed and the small amount of memory to do it is not available.

Answering the initial BASIC memory number with 3782 gives the maximum possible amount of memory for the BASIC program. The data and machine language routines go from 3783 to 4095 which takes up the rest of the 4K RAM. BASIC uses from 3782 down.

Channels are found by starting from the beginning of the DATA and searching to the end. If a channel is requested but not found, a message is given to that effect. In channels 100 and above, a channel requested and not found gives a different message showing which designated channel had the error in it. These channels, 100 and above, take longer to compute because a full search must be made for each of the specified sensor channels. This search time could be eliminated but another 2K memory would be needed and the time is not a problem anyway. It takes approximately 10 sec at most to do one 100-group channel.

Results are printed out to one-tenth unit accuracy. This was based primarily on temperature needs but is more than adequate for other parameters.

Present memory limits allow only about 60 channels (total) to be input while keeping the program all on one tape file. Separating the statements and data used only for machine language into their own file would allow another 25 or so channels to be added. This includes maximum data compression in the DATA statements.

Channel 0 is always the Real Time Clock. The SDAS output for the RTC consists of four bytes: High, Mid, Low, Low. These first three bytes are converted as one 24-bit word to decimal and printed. Each count represents 2 sec real time.

Channel 1 is always the lower 48-channel SDAS offset. It is sent as two bytes: high and low. This is assumed to be one 16-bit (rather than a 10-bit or two 8-bit words) and is converted to decimal as such. This is normally only a few counts or zero.

Channel 999 is always the last one and signals the end of channel data. It, like channels 0 and 1, is part of the primary program and never changes.

The number of channels seen from the SDAS is printed at the beginning of a full printout. This should be some multiple of 16 unless an error has occurred. The proper number is known for each site and can be checked from time to time.

All sensor channels, 2-96 (or to 99 if desired), are converted as one 8-bit word and the most significant two bits of the second 8-bit word to make up one 10-bit result. The two bits are the least significant bits of the 10-bit word. The resulting decimal count is then converted to the proper units for that sensor.

Further details can be obtained from the BASIC program listing, Figure D-1, which follows. Lines 1000 through 1032 totally define this typical site. All other lines are part of the primary program and never change. Also shown is a typical full printout in Figure D-2.

```
10 DIMA(5),B(3),C(4),A$(16):D=256:Z=3785:POKE04,139:POKE05,15
11 FORI=1TO200:GOSUB200:IFP=999GOTO20
12 NEXT
20 S=I:X=1:FORI=0TO16:READA$(I):NEXT
30 FORI=3979TO4095:READP:POKEI,P:NEXT
40 INPUT"1MAN,2AUT OR 3ALL";B:ONBGOTO70,55,55
50 GOTO40
55 INPUT"INT-M";T:IFB=3GOTO140
60 INPUT"#CH'S<=10";N:C%(0)=0:IFN=0GOTO140
61 FORI=1TON:INPUTC%(I):NEXT:GOTO140
70 INPUT"CH";C%(0):T=0:N=0:X=1:GOTO150
80 FORQ=0TON:RESTORE:FORI=1TOS:GOSUB200
81 IFP=C%(Q)ANDP<400THENC=P:GOSUB400:NEXTQ:GOTO130
90 NEXT:PRINT!C%(Q);"?":GOTO40
130 FORI=1TO42*(1+30*ABS(T*X)):P=SQR(2):NEXT:IFB=1GOTO70
140 IFX=1THENPRINT" "
150 Y=USR(1):IF(Y=0ORY=200)ANDX=1THENPRINT!"TIMEOUT":Y:X=0:GOTO130
131 IFY=0ORY=200GOTO130
152 X=1:IFB<>3GOTO80
155 PRINT.5*Y-2;"CHANNELS"
160 RESTORE:C=0:FORQ=1TOS:IFC<100GOTO165
161 RESTORE:FORI=1TOS:GOSUB200:IFC=PGOTO165
162 NEXT
165 GOSUB200:C=P:IFP<400THENGOSUB400
170 NEXTQ:IFT=0GOTO40
171 GOTO130
180 PRINTC;.1*INT(10*V);A$(C(4)):RETURN
200 READP:FORM=0TO4-(P>399):READA(M):NEXT:RETURN
300 IFP=0THENY=PEEK(Z-2)*D*D+PEEK(Z-1)*D+PEEK(Z):GOTO320
305 IFP=1THENY=PEEK(Z+2)*D+PEEK(Z+3):GOTO320
310 V=Z+2*P:Y=4*PEEK(V)+PEEK(V+1)/64:IFA(4)=4THENY=SQR(Y)
315 IFA(4)=1THENFORL=0TO3:A(L)=A(L)/10^(3*L+1):NEXT
320 V=A(0)+Y*A(1)+Y^2*A(2)+Y^3*A(3):RETURN
400 IFC<100THENGOSUB300:C(4)=A(4):GOSUB180:RETURN
500 FORJ=0TO4:C(J)=A(J):NEXT
510 FORJ=0TO2-(C<200)+2*(C>299):RESTORE
520 FORK=1TOS:GOSUB200:IFP=C(J)THENGOSUB300:B(J)=V:GOTO540
530 NEXTK:GOTO600
540 NEXTJ:ONINT(C/100)GOSUB550,560,570:RETURN
550 Y=(B(0)+B(1))/2:C(3)=(A(0)+Y*(A(1)+Y*A(2)))*(A(3)+Y*(A(4)+Y*A(5)))
560 V=(B(0)-B(1))*B(2)*C(3)*(60+58.92*(C>199)):GOSUB180:RETURN
570 V=B(0)*C(1):GOSUB180:RETURN
600 PRINT"ERROR CH";C:END
999 DATA0<0,1,0,0,9,1,0,1,0,0,10
```

Figure D-1. BASIC program listing.

```
1000 DATA400,.7742,4.221E-4,3.984E-7,8.838,-2.382E-3,-5.03E-6
1001 DATA401,1.0115,-2.348E-4,10.37E-7,8.346,.4129E-3,-9.961E-6
1002 DATA6,319,1241,23,0,1
1003 DATA9,325,3991,143,0,1
1004 DATA12,317,1983,33,0,1
1005 DATA14,323,1977,64,0,1
1006 DATA16,327,1962,48,0,1
1007 DATA17,324,1966,44,0,1
1008 DATA18,323,1982,33,0,1
1009 DATA20,321,1982,33,0,1
1010 DATA22,321,1982,33,0,1
1011 DATA24,315,2001,18,0,1
1012 DATA26,335,1981,65,0,1
1013 DATA28,325,1964,81,0,1
1014 DATA30,330,1979,65,0,1
1015 DATA32,330,1959,63,0,1
1016 DATA10,-170,1363,16,0,1
1017 DATA3,0,.9334,0,0,5
1018 DATA5,0,1.919,0,0,4
1019 DATA7,0,.9224,0,0,5
1020 DATA11,0,15,0,0,4
1021 DATA13,0,9.744,0,0,4
1022 DATA15,0,.9225,0,0,11
1023 DATA19,0,.00391,0,0,2
1024 DATA21,0,.001955,0,0,2
1025 DATA23,0,.0009765,0,0,2
1026 DATA8,0,2.961,0,0,0
1027 DATA100,14,12,11,400,3
1028 DATA101,28,26,13,401,3
1029 DATA102,32,30,5,401,3
1030 DATA2,324,1255,14,0,1
1031 DATA4,336,2032,18,0,1
1032 DATA300,8,6000,0,0,3
2000 DATA999,0,0,0,0,0,0
2001 DATABTU/FT2-HR,DEGF,KW,BTU/HR,GPM,GAL,FPM,DEG
2002 DATAVOLTS,RTC,COUNTS,CF,PER,CFM,MPH,SW,MISC
2003 DATA173,13,160,141,13,160,169,0,141,3,160,141,2,160,169,3
2004 DATA141,11,160,169,29,141,12,160,162,0,160,0,232,224,0,240,74,169
2005 DATA18,205,13,160,208,244,169,31,141,12,160,169,255,141,9,160,173
2006 DATA0,160,41,96,240,9,173,13,160,41,32,240,242,208,18,162,2,169
2007 DATA18,141,13,160,173,13,160,201,18,240,8,201,32,208,245,160,200
2008 DATA208,19,173,1,160,73,255,153,199,14,200,202,208,224,173,0,160
2009 DATA41,224,208,194,169,31,141,12,160,169,0,76,209,192
```

Figure D-1. (Concluded).

```
32 CHANNELS
0   52552 RTC
1   0 COUNTS
6   120.7 DEGF
9   151.4 DEGF
12  127.8 DEGF
14  144.8 DEGF
16  122.1 DEGF
17  112.9 DEGF
18  120.3 DEGF
20  120.7 DEGF
22  113.9 DEGF
24  112.6 DEGF
26  107.4 DEGF
28  129.4 DEGF
30  133.3 DEGF
32  155 DEGF
10  64.3 DEGF
3   640.3 GAL
5   11.1 GPM
7   378.1 GAL
11  111.2 GPM
13  71.6 GPM
15  753.6 CF
19  2.8 KW
21  .9 KW
23  .1 KW
8   284.2 BTU/FT2-HR
100  800210.4 BTU/H
R
101  777847.9 BTU/H
R
102  119215.4 BTU/H
R
2   56.6 DEGF
4   124.5 DEGF
300  1705536 BTU/HR

END
```

Figure D-2. Typical full printout for 32-channel system.

# APPENDIX E

## INPUT FORMATS

1. DATA statements define the site and performance parameters.

2. The DATA statements become part of the total program and can be stored on tape as a whole so only one tape load is required per site.

3. Statement numbers (BASIC) from 1000 to 1999 are reserved for the DATA statements which define the site to the computer.

4. Order of inputting channel numbers is arbitrary (see no. 10 below). A total printout will have the order the same as they are input. Program execution will be slightly faster if channels 2-99 appear first.

5. The desired performance parameters are assigned a channel number and they are accessed by that channel number. A log must be kept of these channels versus parameter for each site since they are not in the IP but are user defined.

6. New channels can be defined using the already programmed equations in any way desired (to monitor counts, multiply by constant, etc.).

7. The total number of channels is limited only by computer memory which is adequate but not overly so. Data input can be altered to increase memory efficiency, in particular by putting data for up to three channels in one DATA statement. An OM error (out of memory) requires better memory efficiency.

8. All numerical values need only have four significant digits. This gives sensor accuracy of 0.1 units, approximately, and performance accuracies of about 0.1 percent.

9. A TM error (type mismatch) will occur if the wrong number of terms is used in a channel.

10. It is better to put channels 400-499 first in the computer when there are any. Data checking by the computer is more complete this way.

Channels 2-99 order:

CH #, A0, A1, A2, A3, units # (see special format for temperatures)

Channels 100-199 order (liquid):

CH #, $CH_{Tout}$, $CH_{Tin}$, $CH_{flow}$, $CH_{coef}$, units #

Channels 200-299 order (air):

$$CH \ \#, \ CH_{Tout}, \ CH_{Tin}, \ CH_{flow}, \ KA, \ units \ \#$$

Channels 300-399 order (insolation):

$$CH \ \#, \ CH_{pyro}, \ A, \ 0, \ 0, \ units \ \#$$

Channels 400-499 order (liquid coefficients, $CH_{coef}$):

$$CH \ \#, \ a_0, \ a_1, \ a_2, \ b_0, \ b_1, \ b_2$$

Temperature coefficients are specially scaled before they are input to the computer in order to save memory and make it easier to input them. The values which go into the computer are:

$$A0 = IP \times 10^1, \ A1 = IP \times 10^4, \ A2 = IP \times 10^7, \ A3 = IP \times 10^{10}$$

They are later rescaled to their true value by the computer. This format allows temperature coefficients to be input with no decimal points and no exponents and still give $0.1°F$ accuracy. This automatic scaling is triggered whenever a 1 occurs in units #.

Whenever a liquid flow rate is being determined (by finding a 4 in units #) the square root of counts is used instead of just counts.

Channel 0 is always the real-time clock output.

Channel 1 is always the lower 48-channel offset counts.

KA is the air duct area in $ft^2$.

A is the solar collector area in $ft^2$.

$CH_x$ is the channel where the desired information is located.

$a_0-b_2$ are the liquid coefficients taken from the tables.

The equation for channels 2-99 is:

$$Y = A0 + A1 \ (X) + A2 \ (X^2) + A3 \ (X^3) \quad (X - decimal \ counts)$$

The equation for channels 100-199 is:

$$Y = (CH_{Tout} - CH_{Tin}) \ {}^*CH_{flow}{}^*Cp{}^*\rho{}^*60$$

where Cp and $\rho$ are determined using coefficients found in $CH_{coef}$. The equation for channels 200-299 is:

$$Y = (CH_{Tout} - CH_{Tin}) * CH_{flow} * KA * 1.08.$$

T.. equation for channels 300-399 is:

$$Y = CH_{pyro} * A$$

The equations for channels 400-499 are:

$$Cp = a_0 + a_1 T + a_2 T^2$$

$$\rho = b_0 + b_1 T + b_2 T^2$$

where $T = (T_{out} + T_{in})/2$ (average temp)

Units for channels 2-99 are as in the IP. Units for channels $> = 100$ are normally in BTU/HR. Only unit #'s 1 and 4 do anything except print.

The particular units which can be printed and their access numbers (as required in channel data) are:

0 - BTU/FT$^2$-HR insolation

1 - DEGF °F, also causes automatic coefficient scaling

2 - KW

3 - BTU/HR

4 - GPM Also causes $\sqrt{\phantom{x}}$ counts to be used

5 - GAL

6 - FPM

7 - DFG

8 - VOLTS

9 - RTC       Real-Time Clock

10 - COUNTS

11 - CF

12 - PER      percent

13 - CFM

14 - MPH

15 - SW         switch

16 - MISC      use for others not listed.

Examples:

1000DATA2,326,1979,50,0,1    [CH2,32.6,.1979,50E-7,0E-10, temperature]

1001DATA21,0,.2955,0,0,4     [CH21,0,.2955,0,0, liquid flow rate]

1002DATA100,12,45,21,400,3   [CH100,CH12,CH45,CH21,CH400,BTU/HR]

1003DATA200,2,26,13,2,3     [CH200,CH2,CH26,CH13,2FT$^2$,BTU/HR]

1004DATA300,7,720,0,0,3     [CH300,CH7,720FT$^2$,0,0,BTU/HR]

1005DATA400,1.011, 2.348E-4,10.37E-10,8.346,.4129E-3,-9.961E-6

Combining DATA statements, say 1003 and 1004 would produce:

    1003DATA200,2,26,13,2,3,300,7,720,0,0,3 (keep < = 72 characters).

Always omit blanks as they use memory too.

# APPENDIX F

## OPERATING INSTRUCTIONS

1. Connect cable, cassette and power.

2. Turn on AIM 65, turn printer off (CTRL, PRINT together).

3. Rewind tape, zero counter, volume full, move tape to within 3 counts of pgm start.

4. Press 5, memory = 3782, RETURN; width = RETURN.

5. Type LOAD, RETURN; IN = T, F = filename, T = 1, RETURN.

6. Start tape playback and note statements moving thru display. If not, reload.

7. When cursor ( ·            ) returns shut off tape.

8. Program is ready to run. Turn on printer if desired (CTRL PRINT).

9. Type RUN, RETURN.

10. Three options are available: MANual = 1, AUTomatic = 2, printALL = 3.

11. Enter choice and press RETURN. RETURN alone gets cursor back.

12. Manual asks only for a channel number, 0-399. When finished with one, it asks for another. When done, press RETURN alone. Go to 9.

13. (a) Automatic asks first for time interval in minutes between print-outs. Zero gives minimum time of 2 sec. Actual time between printouts is 2 sec + time interval requested + print time (if any).
    (b) Enter number of channels less than or equal to 10 to be monitored at once. Channel zero (Real Time Clock) is always given first automatically as a time interval reference on the printout. One RTC count = 2 sec.
    (c) Enter specific channels desired one at a time with a RETURN for each.
    (d) Press F1 key to interrupt computer when done. Go to 9.

14.    printALL asks only for the time interval in minutes. Actual time between prints is as explained in 13(a). However, entering a zero in this case just gives one printout. Anything greater than zero gives a print interval as before. When interval is being used, press F1 as before to stop. Full printout always includes RTC (CH∅) and SDAS offset (CH1) as well as number of channels read by SDAS. The minimum time between SDAS reads is 2 seconds under any conditions. Each print interval output is based on just one SDAS read no matter how long is required to compute and print the results.

Once the program is loaded, new channels can be added, within memory limits, by getting the cursor in the display, choosing an unused line number (1000-1999) and typing the DATA statement according to the proper format (see INPUT FORMAT). Pressing RETURN when through typing the line enters it into the main program. It can then be treated like any other channel. Entering line number, RETURN deletes that line.

If the ESC (escape) key is ever pressed (to return to the monitor, for example) or if the reset (panic button on left side of case) is pressed, BASIC can be re-entered just by pressing 6. This gets the cursor back in the display so changes can be made or RUN typed in.

If a read is attempted when the SDAS is busy, or if bad data is detected, this fact will be printed out by the word TIMEOUT followed by the number zero or 200. Zero means the SDAS was busy and 200 means a noise or hardware problem occurred. In either of the interval modes, the AIM 65 will keep trying the SDAS (each 2 seconds) until it responds. If a cable is broken, it will just wait until the operator intervenes with F1.

Data can be stored on tape using the monitor dump (D) command and read back in later using the monitor load (L) command. The data starting (FROM) address is $∅EC7 and the ending (TO) address is $∅F8A. Before giving RETURN after specifying tape unit number, turn on recorder. Answer MORE? with N. Both record and playback can be done using remote control of the player by turning tape unit 1 off before using D or L commands. Before recording, change $A409 from $08 to $20 or the data (or BASIC program) will not reload. The monitor M command is used for this change. Type M, A409, RETURN. Then /, 20, RETURN.

Any tape storage with the AIM 65 using the present cassette recorder requires that the tape gap be increased from the default value stored in the monitor (not necessary for playback). The $08 must be increased to at least $10 and preferably $20 (see above). This applies to BASIC as well as monitor recording. A different recorder may not require this change.

A log must be kept of programs, counter values for program or data locations, dates, program type, file name, etc., for later records.

The only known way the program can hang up is if there is a problem in the hardware or SDAS. Pressing the reset will then be necessary in order to regain control.

Once all site data has been entered into the computer, the entire program can then be saved (by BASIC) using the SAVE command. Type SAVE, RETURN; then T, five character file name, 1. Before pressing RETURN, start recording at the desired tape counter location. A more accurate way (so far as file start location) is to first go back to the monitor (ESC) and turn off tape unit 1 (1 key) then press the record keys on the recorder so it will start remotely by itself. Then re-enter BASIC (6 key) and go through the SAVE procedure. Be sure that memory location %A409 was changed to $70 as mentioned before for recording.

The program can be exercised without an SDAS by changing program line 150. Do this by typing:

150Y 100, RETURN

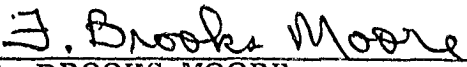This assumes a 48 channel system. The value for Y (number of data strobes) is 2* (number channels) + 4 in case some other number of channels is being used. Y can be anything except zero or 200.
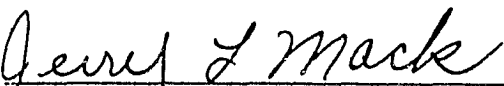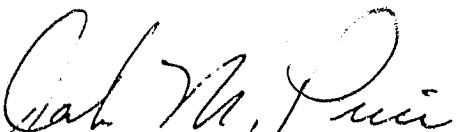
# APPROVAL

## SOLAR SITE TEST MODULE

By Ralph R. Kissel and Donald R. Scott

The information in this report has been reviewed for technical content. Review of any information concerning Department of Defense or nuclear energy activities or programs has been made by the MSFC Security Classification Officer. This report, in its entirety, has been determined to be unclassified.

F. BROOKS MOORE
Director, Electronics and Control Laboratory

JERRY L. MACK
Chief, Guidance, Control and Instrumentation Division

CLYDE S. JONES JR,
Chief, Electronics and Servo-Analysis Branch

WILLIAM A. BROOKSBANK, JR.
Manager, Solar Energy Applications Projects