FINAL REPORT

NAS8-36955
DELIVERY ORDER 134

VACUUM CHAMBER TRANSLATION / POSITIONING MECHANISM AND WELDING POWER SUPPLY CONTROLLER

Submitted to

Richard M. Poorman
George C. Marshall Space Flight Center
National Aeronautics and Space Administration
Huntsville, Alabama 35812

Prepared by

James E. Smith, Jr., Ph.D.
Associate Professor and Chairman

and

John L. Cashon
Graduate Research Assistant

Department of Chemical and Materials Engineering
College of Engineering
The University of Alabama in Huntsville
Huntsville, Alabama 35899

August 1992

FINAL REPORT

NAS8-36955
DELIVERY ORDER 134

VACUUM CHAMBER TRANSLATION / POSITIONING MECHANISM
AND WELDING POWER SUPPLY CONTROLLER

Submitted to

Richard M. Poorman
George C. Marshall Space Flight Center
National Aeronautics and Space Administration
Huntsville, Alabama 35812

Prepared by

James E. Smith, Jr., Ph.D.
Associate Professor and Chairman

and

John L. Cashon
Graduate Research Assistant

Department of Chemical and Materials Engineering
College of Engineering
The University of Alabama in Huntsville
Huntsville, Alabama 35899

August 1992

TABLE OF CONTENTS

Vacuum Chamber Translation/Positioning Mechanism

and Welder Power Supply Controller

1.0 Introduction

Welding in the vacuum of space represents an important and fundamental problem for space exploration. Repairs or connection of metal components on orbit or during travel to the moon or distant planets may be required. Cracks or holes in space craft skin or supporting structures external to the pressurized section will require some type of repair that must be permanently made to the skin or support by welding. This research addressed the development of a translation/positioning system that will permit research into welding of metal samples in a small vacuum chamber located at Marshall Space Flight Center (MSFC). The unit, described below, is fully programmable and currently configured to perform straight line translation. The software and hardware have also been provided for the eventual control of start and power control functions of a welder power supply located at MSFC.

A photograph of the completed device is shown in Figure 1. For reference and following discussions, a coordinate system will be assigned as Z, the vertical axis, X, the lateral axis, and Y the horizontal axis or direction in which the sample table is translated.
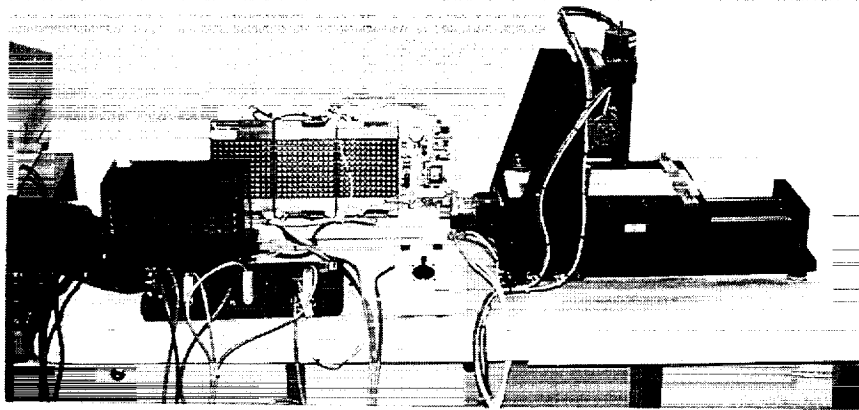
1

FIGURE 1
Vacuum Chamber Translation / Positioning Mechanism
and Welding Power Supply Controller
Left to Right: Welding Power Supply Controller Relay,
24VDC Power Supply, PC23 Interface Board, PC23 DB Drives
& 5VDC Power Supply mounted, Daedal Joystick, 3-axis
Translation / Positioning Mechanism

2.0 Equipment

The vacuum welding positioning system is composed of a custom designed three axis positioning system with vacuum sealed industrial grade microstepping motors and three subsystems. The subsystems are the IBM compatible PC23 three axis position control system with motor interface and limit and home switch controls, joystick for manual positioning of two axes at a time and a Keithly/Metrabyte DAC-02 D/A interface card that may be used to control welder start and power level functions.

The major components that were used to develop the vacuum welder positioning system and computer cards that can be used to control the welder power supply are listed in Table 1.

2.1 Translation/Positioning System

The translation/positioning mechanism for the vacuum welding system under development at MSFC, was manufactured from three individual translation devices. The specimen to be welded, namely 2 - 2-4" wide by 12-14" long by 1/4" thick metal plates, are mounted on a specially designed translation table and moved below two Daedal model 105021S-20E-LH series linear tables that are 5" on a side and provide 2" of linear travel. These tables are mounted in tandem on supports above the sample to form a two axis positioning system. This permits both vertical (Z) and lateral (X) positioning of a welder electrode or other device over the sample. Straight line positioning accuracy for these units is ±0.0004in with a total load capacity of 30 pounds.

TABLE 1.

Parts Listing

Item	Manufacturer	Model #	Serial #
1. X, Z Translation Motors	Daedal	EMA57-51-V-LH	2002 2003
2. Y Translation Motor	Daedal	EMA57-51-V-LH	2004
3. X, Z Linear Tables	Daedal	105021S-20E	9202111501 9202110501
4. X Linear Table	Daedal	5086141S-LH	92011550601
5. Motor Drives	Compumotor	DB-Drives	902021000064 902021000065 902021000066
6. Joystick	Daedal	JS5300	9112032CN02
7. 5 VDC Power Supply	TDK	MRW150KV	9Z414565
8. 24 VDC Power Supply	Sola Electric	28-24-280-0	0890210

Horizontal positioning of the sample is provided by a Deadal rail table model 506141S-20E-LH which provides 14" of travel with a straight line accuracy of ±0.002 in/in or ±0.0028 in over the total distance traveled. The units positioning accuracy is ±0.0035 in over the travel with a repeatability of ±0.0002 in. The load capacity of this rail is 200 pounds when mounted in the horizontal direction as used in the position systems.

2.2 Positioning Motors

All axes on the vacuum welding translation/positioning system are powered by identical vacuum sealed industrial grade microstepping motors. As delivered, these motors are programmed for 25,000 pulses per revolution. One revolution moves the translation mechanism 0.20 inches for an approximate resolution of 8×10^{-6} in/pulse. The stepping motors are sealed units designed to run in vacuum. They have been vacuum prepared to industrial standards by Daedal and run off of individual drive controllers located outside the vacuum environment, powered by an independent 24 VDC power supply.

2.3 Equipment Mounts

The translation equipment is mounted to a unit designed and fabricated at UAH. The mount is assembled from five heavy aluminum plates. The base plate was prepared from a 3/4" thick aluminum plate to which the Deadal model 506141S-20E-LH rail table mounts using 1/4x28 S.S. cap screws. The base plate has

adjustable legs with plastic ends to both level and cushion the unit. At one end of this base plate, two vertical supports are attached by 1/4x28 S.S. cap screws. These supports were manufactured from 3/4 inch thick aluminum plate. The vertical support members are connected by a horizontal strut that is fastened by 1/4x28 S.S. cap screws. The horizontal strut was prepared from 1/2" thick aluminum plate and has threaded holes in the center that match the Deadal model 105021S-20E-LH series linear table.

To the rail table translation platform is mounted the sample support table. This table is 14" long and can accommodate samples up to this length, though a 12" sample length is recommended to provide shielding for the translation components below the unit. The table has a channel that is lined with a thin sheet of brass to protect the table. The table contains a hole pattern that accepts 1/4x28 mounting screws (S. S. cap screws have been provided with the unit). Aluminum mounting dogs are used to secure the samples in place and are provided with 1/4 inch spacers to facilitate sample capture. This table is connected to the translation platform by 1/4x20 <u>black iron</u> cap screws so caution must be exercised when connecting or disconnecting this component, since all other screws throughout the unit are 1/4x28 <u>S.S.</u> cap screws.

All mounting plates have been black anodized to minimize reflection during the welding process. Electrical cables that power and control the motor and translation tables are attached to the mount by pee-clamps making the unit easy to move and service.

2.4 PC23 Positioning Control System

The IBM compatible PC23 positioning control system is composed of three components. A computer interface card that must be installed in the back-plane of a 286 or greater IBM compatible computer. To this card, and external to the computer, is an interface bus that is connected by cables to the steppermotor drivers and home/limit switch assemblies. This external interface is powered by a separate 5 VDC interface power supply which is the third component. Through the appropriate software, written in BASIC, this system provides the pulses to position and translate the microstepper motors and protects the unit from damage at the limits of translation.

2.5 Joystick

The vacuum translation/positioning system has been provided with a joystick that can be used to manually position any two axes at a time after simple software commands. The proportional joystick, Daedal model JS5300 interfaces with the PC23 through the external interface and provides direction and velocity control of any two of three axes.

7

3.0 Control Software

Operations software for the vacuum translation system is supplied on a 5 1/4" floppy disk. Copies of the source code are provided in the files, VW.BAS and VW.BAK. A listing of the program VW.BAS is provided in Table 2 and can be used as a guide to further customize the software for particular needs. The operations software was developed around a demonstration program SIMPLEX.BAS, provided by Compumotor for the PC23 hardware. The PC23 control portion of the SIMPLEX code has not been altered, and a description of its operation can be found in the documentation provided by Compumotor. The VW operations software issues commands defined in the Compumotor X Command Language in order to control the movement of the motor that moves a given axis. These commands are also described in Compumotor literature delivered with the unit.

Welder power may be controlled using a Keithly/Metrabyte DAC-02 two channel digital to an analog converter board and a 12 VDC micro-mechanical relay that handles up to 5 amperes. Channel one from the DAC-02 turns on and off a relay that can be wired to the foot control pedal interface of the welder power to replace the on-off switch currently used by the foot pedal controller. The power level from the foot pedal controller provides 0-10 VDC on the wiper of the potentiometer relative to ground as the operator depresses the foot pedal control. This function can also be controlled by the computer using channel two of the DAC-02. Channel two outputs a proportional 0-10 VDC signal which may

be wired between the negative wire of the foot pedal and the wiper of the potentiometer. In order to do this the foot pedal must be removed from the circuit. The output voltage of the DAC-02 is determined by the operations software, based on inputs requested of the operator. These inputs are the maximum current which the welder operates and the currents at which to weld. A linear interpolation is then accomplished and a subsequent output voltage between 0-10 VDC determined. Both channels are controlled with OUT statements in the VW software.

The operations software is supplied in compiled form for ease of operation and runs from the floppy disk after typing the VW command to run the program VW.EXE. The VW software was written and compiled using Microsoft's QuickBasic version 4.0.

4.0 Operating Procedures

To begin operations, apply power to the 24 VDC power supply and computer. Insert the operations software floppy disk into the disk drive of any 286 or greater IBM compatible computer (not supplied with the system) and access the disk by typing A: or B: depending how the system is configured. Once the proper drive is enabled, simply issue the command "VW" to begin systems operations. At this time all three translation stages will move to their "home" positions. The user is then prompted to respond to a series of questions regarding the desired weld to be performed, and welder system parameters. A list of these questions are given in Table 2.

9

TABLE 2.

Operating Commands and Limitations

Prompt   Limitation or Comments

Enter the length of the weld (min .5 inch, max 14 inches)?

Enter the weld translation speed (min .05 in/sec, max .4 in/sec)?

Enter electrode weld height (max .5 inch)? No negative numbers.

Enter time to raise electrode (sec)? Checks max and min velocity.

Enter the MAXIMUM operating current for the system?

Enter the initial current to begin with?

Do you wish to change any of the welding parameters (Y,N)?

After these questions are answered, the user is then prompted to use the joystick to manually align the sample to be welded with the welding electrode. Finally, the user must manually position the electrode to the initial position and set the desired height to begin the weld.

To align the sample, place the samples on the table and loosely tighten the dogs so that the sample will just move. Enable the joystick by typing either "A" to move the X (lateral) and Z (vertical) axes or "B" to move the Y (horizontal) or Z (vertical) axes, per the screen instructions. Using the A or B software modes, lower the electrode and position it at one end of the table approximately at the center line of the table. Adjust the position of the samples until the seam is just below the electrode. Select B, and horizontally translate the table to the other end and again align the samples with the electrode. Check the alignment by horizontally translating along the seam to be welded. When the sample is aligned to the users satisfaction, tighten the dogs to secure the sample.

The electrode must be manually moved to the starting position. The starting position is defined as the furthest position along the sample seam and away from the motor end of the horizontal translation axis. Select B and manually move the electrode to this end of the sample and using a feeler gauge, carefully lower the electrode to the initial starting gap. Once the user is

11

satisfied with the electrodes position, and vacuum has been achieved, the user should press "C" to exit to the next screen and follow the prompts to begin welding.

The next screen menu is then displayed showing system parameters, system power ON/OFF, maximum system power, and key functions for desired operations. The user can press "S" to begin system operations (welder start and power levels are output and horizontal translation of the samples are begun simultaneously).

In operations, the system first enables the start switch of the welder power supply and outputs the appropriate voltage to the power supply consistent with the operators initial power requirements to strike the arc. It then raises the electrode at the user defined rate to the steady welding height which is relative to the initial starting height set during the alignment procedures. It then begins horizontally translating the sample at the user defined rate. While the system is translating, the user can increase or decrease the welder current in 1% increments of the maximum scale by pressing the "U" or "D" key of the key board to increase or decrease to welder power, respectively. Pressing the "T" key will terminate the welding operations at any time during the sequence.

Upon completion of the defined operation, or interruption from the operator, the software automatically turns off power to the welder, and returns the translation stages to their home posi-

tion. The operator is then allowed to exit the applications software or begin another weld following the same procedures as described above.

5.0 Final Actions to Complete and Install the System at MSFC.
The principle actions to install the system at MSFC is to insert the two cards into the back-plane of the user supplied 286 or greater IBM compatible computer and check for address conflicts. The addresses for these two boards are &H310 for the DAC-02 (the smaller board) and &H304 for the stepper motor drive board (the larger board supplied). Further information on these boards is available in the manufacturers manuals supplied with the system.

Mount the panel containing the DB-Drivers and 5VDC power supply and interface, noting wire and cable locations if they must be disturbed, to a user supplied standard electronics cabinet or similar supporting surface. Provide support and power connections for the 24VDC power supply. Provide feed through connectors and cut and connect wires to the motors and limit/home switches matching wire colors and gauge. Additional wiring and operating information for the DB-Drivers, external interface, Home\Limit switches and stepper motors are provided in the manufacturer manuals supplied with the system.

13

The final steps that must be accomplished at MSFC are to develop a machine mounting hardware to attach the welding torch and gas supply lines to the upper translation mechanism and grounding wire to the translation table. It has been assumed that the mount for the torch, will displace the point of the electrode 2" out from the Z translation surface and be mounted such that it will touch the translating samples within the 2" translation distance of the Z translation stage.

6.0 Conclusions

The system and associated software has been tested to the extent possible without the availability of the welder power supply or control computer that must be supplied by MSFC. Software has been developed for straight line welding. More extensive and varied translations are possible with simple alterations to the operating software to use the full capabilities of this three axes system. The source code "VW.BAS" has been provided to serve as an example for further development of the vacuum welder translation system.

With the delivery and acceptance of the translation mechanism and the equipment for eventual control of the welder power supply, we have met the requirements of this contract.

APPENDIX 1

Listing of VW.BAS Source Code
Prepared Using Microsoft's QuickBasic

```
10 '     ********************************************************************
11 '     **                                                              **
12 '     **            NASA-MSFC / UAH   Vacuum Welding System           **
13 '     **                                                              **
14 '     **                        Developed by                         **
15 '     **                                                              **
16 '     **                  Dr. James E. Smith, Jr.                     **
17 '     **                          and                                **
18 '     **                     John L. Cashon                          **
19 '     **                                                              **
20 '     **      Department of Chemical and Materials Engineering        **
21 '     **                                                              **
22 '     **           University of Alabama in Huntsville               **
23 '     **                Huntsville, Alabama 35899                    **
24 '     **                   (205) 895 - 6810                          **
25 '     **                                                              **
26 '     ********************************************************************
27 '
28 '
29 '
180 GOSUB 200: REM    Initialize the PC23 for translation
190 GOTO 5000: REM    Run the interactive Vacuum Welding System code
191 '
192 '
193 '
194 '
196 '
197 '       The following list of variables is used to keep track of
198 '       addressing and the Control Byte
199 '
200 ADDRESS% = 772: REM  This is the factory default base address setting
210 '
220 INTRCLR = &H20 'This variable is for clearing Control Bit 5
230 '              (to signal "Restart Watchdog Timer")
240 '
250 RESTART = &H40 'This variable is for clearing Control Bit 6
260 '              (to signal "Restart Watchdog Timer")
270 '
280 CONTROL = &H60 'This is the normal state of the Control Byte
290 '              (only Bits 5 and 6 are high)
300 '
310 READY = &H17    'This is the normal state of the Status Byte
320 '               (bits 0, 1, 2, and 4 set)
330 '
340 HALT = &H64     'This variable is for setting Control Bit 2
350 '               (to signal the "Watchdog Timer" to time out)
360 '
370 CMDRDY = &H70   'This variable is a mask for setting Control Bit 4
385 '               (to signal "Command Byte Ready in the IDB or not)
390 '
400 RECEIVED = &HE0'This variable is a mask for setting Control Bit 7
410 '               (to signal "Message Received from the ODB")
420 '
430 STOPPED2 = 1     'This variable is a mask for testing Status Bit 1
440 '                (is the motor moving?)
450 '
460 STOPPED1 = 2     'This variable is a mask for testing Status Bit 1
470 '                (is the motor moving?)
480 '
490 STOPPED3 = 4     'This variable is a mask for testing Status Bit 1
```

16

```
 500 '                    (is the motor moving?)
 510 '
 520 ODBREADY = 8    'This variable is a mask for testing Status Bit 3
 530 '                    (is a response waiting in the Output Data Buffer?)
 540 '
 550 IDBREADY = &H10'This variable is a mask for testing Status Bit 4
 560 '                    (is the Input Data Buffer ready for a command byte)
 570 '
 580 FAIL = &H20     'this variable is a mask for testing Status Bit 5
 590 '                    (has the STD22 suffered a processing  failure?)
 600 '
 610 MASK = &H7F     'this variable is a mask for the Status byte MSB
 620 '
 690 '          ***********************************************************
 691 '          *                                                         *
 692 '          *              PC23 "RESET" SUBROUTINE                    *
 693 '          *                                                         *
 694 '          ***********************************************************
 695 '
 696 ' The following subroutine allows the "Watchdog Timer" to timeout,
 697 ' and Reset the PC23.  Then the timer is restarted.  A "GOSUB 700"
 698 ' instruction will reset the PC23.
 699 '
 700 BYTE = 0: TIMEOUT = 10000                     'Set timeout duration
 710 OUT ADDRESS% + 1, (HALT)             'Control Bit 2 high
 720 WHILE (BYTE AND FAIL) = 0 AND TIMEOUT > 0'Test for fail or timeout
 730    BYTE = INP(ADDRESS% + 1)                  'Read Status Byte
 740    TIMEOUT = TIMEOUT - 1: WEND                'repeat until timeout or fail
 750 IF TIMEOUT <= 0 THEN PRINT "Invalid response from address"; ADDRESS%: END
 800 BYTE = 0: TIMEOUT = 10000                     'Set timeout duration
 810 OUT ADDRESS% + 1, (RESTART)             'Control Bit 2 high
 820 WHILE (BYTE AND MASK) <> READY AND TIMEOUT > 0
 830    BYTE = INP(ADDRESS% + 1)                  'Read Status Byte
 840    TIMEOUT = TIMEOUT - 1: WEND                'repeat until recovery
 850 OUT ADDRESS% + 1, (CONTROL)               'Restore Control byte
 860 OUT ADDRESS% + 1, (INTRCLR)               'Restore Control byte
 870 IF TIMEOUT <= 0 THEN PRINT "Timeout recovering from reset!": BEEP: END
 880 FOR I = 1 TO 1200: NEXT
 890 RETURN
 975 '
 976 '          ***********************************************************
 977 '          *                                                         *
 978 '          *              PC23 "BASIC" OUTPUT DRIVER                 *
 983 '          *                                                         *
 994 '          ***********************************************************
 995 '
 996 '      The following is a handshake subroutine allowing data to be
 997 '      transferred from the PC BUS to the PC23.  Command string data
 998 '      is sent to the PC23 one character at a time.
 999 '
1000 FOR I = 1 TO LEN(CMD$)
1010     CHAR$ = MID$(CMD$, I, 1)                 'fetch command characters and
1020     GOSUB 1100                               'send them one at a time
1030 NEXT I
1040 CHAR$ = CHR$(13): GOSUB 1100                 ' follow with a carriage return
1045 FOR I = 1 TO 1200: NEXT I
1050 RETURN
1070 '
1100 BYTE = 0: TIMEOUT = 10000                         'Set timeout duration
1110 WHILE (BYTE AND IDBREADY) = 0 AND TIMEOUT > 0'Test for ready or timeout
```

```
 1120    BYTE = INP(ADDRESS% + 1)                    'Read Status Byte
 1130    TIMEOUT = TIMEOUT - 1: WEND                 'repeat
 1140 IF TIMEOUT <= 0 THEN PRINT "Timeout during write!": BEEP: END
 1150 OUT ADDRESS%, ASC(CHAR$)                       'write command character
 1160 OUT ADDRESS% + 1, (CMDRDY)                     'signal character waiting
 1200 BYTE = 255: TIMEOUT = 10000                    'Set timeout duration
 1210 WHILE (BYTE AND IDBREADY) > 0 AND TIMEOUT > 0'Test for busy or timeout
 1220    BYTE = INP(ADDRESS% + 1)                    'Read Status Byte
 1230    TIMEOUT = TIMEOUT - 1: WEND                 'repeat
 1240 OUT ADDRESS% + 1, (CONTROL)
 1250 IF TIMEOUT <= 0 THEN PRINT "Timeout after write!": BEEP: END
 1260 RETURN
 1270 '
 2990 '         *******************************************************
 2991 '         *                                                     *
 2992 '         *      COMPUMOTOR PC23 "BASIC" INPUT DRIVER           *
 2993 '         *                                                     *
 2994 '         *******************************************************
 2995 '
 2996 '     The following is a handshake subroutine allowing data to be
 2997 '     transferred from the PC23 to the PC BUS. This data is sent
 2998 '     one character at a time.
 2999 '
 3000 ANSWER$ = ""                                   'Initialize response string
 3010 BYTE = 0: TIMEOUT = 5                          'Initialize variables
 3020 WHILE (BYTE AND ODBREADY) = 0 AND TIMEOUT > 0 'Test for ready or timeout
 3030    BYTE = INP(ADDRESS% + 1)                    'Read Status Byte
 3040    TIMEOUT = TIMEOUT - 1: WEND                 'repeat
 3050 IF TIMEOUT <= 0 THEN RETURN                    'Give up if no message
 3060 ANSWER = INP(ADDRESS%)                         'Read 1 response byte
 3070 OUT ADDRESS% + 1, (RECEIVED)                   'Signal character received
 3100 BYTE = 255: TIMEOUT = 1000                     'Initialize variables
 3110 WHILE (BYTE AND ODBREADY) > 0 AND TIMEOUT > 0 'Test for busy or timeout
 3120    BYTE = INP(ADDRESS% + 1)                    'Read Status Byte
 3130    TIMEOUT = TIMEOUT - 1: WEND                 'repeat
 3140 IF TIMEOUT <= 0 THEN PRINT "Timeout after read!": BEEP: END
 3150 OUT ADDRESS% + 1, (CONTROL)                    'Restore control byte
 3160 CHAR$ = CHR$(ANSWER)                           'convert code to char.
 3170 ANSWER$ = ANSWER$ + CHAR$                      'Add char to answer
 3180 IF CHAR$ = CHR$(13) THEN RETURN ELSE 3010
 3190 '
 4000 '
 4001 '
 4002 '
 4003 '     The following code was developed for controlling the
 4004 '          NASA-MSFC / UAH   Vacuum Welding System
 4005 '
 4006 '
 5000 ' *** SUBROUTINE :  RUN VACUUM WELDER  ***
 5001 '
 5010 GOSUB 6000: REM SYSTEM INITIALIZATION
 5012 GOSUB 19000: REM POSITION ELECTRODE WITH JOYSTICK
 5020 GOSUB 7000: REM RUN SYSTEM
 5999 '
 6000 ' *** SUBROUTINE :  SYSTEM INITIALIZATION ***
 6001 '
 6010 CLS
 6015 GOSUB 12000: REM  MAKE SURE RELAY POWER IS OFF
 6016 GOSUB 17000: REM SET UP PC23
 6017 GOSUB 18000: REM SEND ALL THREE AXIS HOME
```

```
6020 CLS : LOCATE 1, 1: PRINT "NASA-MSFC / UAH   Vacuum Welding System": PRINT "
6024 PRINT "": PRINT "Enter the length of the weld (min .5 inch, max 14 inches)"
6025 IF WL > 14 OR WL < .5 THEN PRINT "": PRINT "Value out of range [.5, 14]!":
6026 PRINT "": PRINT "Enter the weld translation speed (min .05 in/sec, max .4 i
6027 IF TS < .05 OR TS > .4 THEN PRINT "": PRINT "Value out of range [.05, .4]!"
6028 PRINT "": PRINT "Enter electrode weld height (max .5 inch)"; : INPUT EH: IF
6029 PRINT "": PRINT "Enter time to raise electrode (sec)"; : INPUT ET: IF ET <
6030 IF ET = 0 THEN ES = 0: GOTO 6036
6031 ES = EH / ET: IF ES > .4 OR ES < .05 THEN PRINT "": PRINT "Electrode speed
6036 PRINT "": PRINT "Enter the MAXIMUM operating current for the system";
6040 INPUT CURM
6045 IF CURM <= 0 THEN PRINT "": PRINT "Value must be greater than zero!": GOTO
6050 PRINT "": PRINT "Enter the INITIAL current to begin with"; : INPUT CURI: IF
6060 CURC = CURI
6063 IF CURI < 0 THEN PRINT "Value cannot be less than zero!": GOTO 6050
6070 CURD = CURM / 100
6071 PRINT "": PRINT "Do you wish to change any of the welding parameters (Y,N)?
6072 KK$ = INKEY$: IF KK$ = "" THEN GOTO 6072
6073 IF KK$ = "y" OR KK$ = "Y" THEN GOTO 6020
6075 GOSUB 21000:    REM  CALCULATE TRANSLATION VALUES
6090 GOSUB 16000:    REM  CALCULATE INITIAL CURRENT OUTPUT VALUES CHANNEL 2
6100 RETURN
6999 '
7000 '  *** SUBROUTINE :  RUN SYSTEM  ***
7001 '
7005 WFLAG = 0
7010 CLS
7020 GOSUB 8000: REM SCREEN DISPLAY
7025 GOSUB 10000: REM OUTPUT CURRENT CHANNEL 2
7030 K$ = INKEY$
7034 IF WFLAG = 0 THEN GOTO 7045
7035 CSEC = TIMER
7036 IF CSEC >= SSEC + WSEC THEN WFLAG = 0: GOSUB 12000: GOSUB 8000
7045 IF K$ = "" THEN GOTO 7030
7050 IF K$ = "S" OR K$ = "s" THEN SSEC = TIMER: WFLAG = 1: GOSUB 11000: GOSUB 22
7060 IF K$ = "T" OR K$ = "t" THEN GOSUB 12000: GOSUB 23000: GOSUB 20000: REM sto
7080 IF K$ = "U" OR K$ = "u" THEN GOSUB 13000: REM INCREASE CURRENT OUTPUT
7090 IF K$ = "D" OR K$ = "d" THEN GOSUB 14000: REM DECREASE CURRENT OUTPUT
7100 GOTO 7010
7110 RETURN
7999 '
8000 '  *** SUBROUTINE : SCREEN DISPLAY OUTPUT ***
8001 '
8010 CLS
8020 LOCATE 1, 1
8030 PRINT "NASA-MSFC / UAH   Vacuum Welding System": PRINT "": PRINT ""
8040 PRINT "Weld Power is "; SCRO1$: PRINT ""
8050 PRINT USING "Maximum Operating Current = ####.## amp"; CURM: PRINT ""
8060 PRINT USING "              Current = ####.## amp"; CURC: PRINT ""
8070 PRINT "Key Functions:": PRINT ""
8080 PRINT "    U :   increase current by 1%": PRINT ""
8090 PRINT "    D :   decrease current by 1%": PRINT ""
8100 PRINT "    S :   start operations": PRINT ""
8110 PRINT "    T :   terminate operations"
8120 PRINT "                    or"
8125 PRINT "             procede after run": PRINT ""
8130 RETURN
8999 '
9000 '  *** SUBROUTINE  :   OUTPUT CHANNEL 1  ***
9001 '
```

19

```
9010 OUT &H310, O1L%
9020 OUT &H311, O1H%
9030 RETURN
9999 '
10000 '   ***   SUBROUTINE : OUTPUT CHANNEL 2   ****
10001 '
10010 OUT &H312, O2L%
10020 OUT &H313, O2H%
10030 RETURN
10999 '
11000 '   ***   SUBROUTINE  :  TURN POWER ON   ***
11001 '
11010 O1L% = 240
11020 O1H% = 255
11030 SCRO1$ = "ON"
11040 GOSUB 9000: REM OUTPUT CHANNEL 1
11050 RETURN
11999 '
12000 '    ***    SUBROUTINE :  TURN POWER OFF ***
12001 '
12010 O1L% = 0
12020 O1H% = 0
12030 SCRO1$ = "OFF"
12040 GOSUB 9000: REM  OUTPUT CHANNEL 1
12050 RETURN
12999 '
13000 '   ***   SUBROUTINE : INCREASE OPERATING CURRENT   ***
13001 '
13010 CURC = CURC + CURD
13015 IF CURC > CURM THEN CURC = CURM
13020 GOSUB 16000: REM CALCULATE OUTPUT VALUES FOR CHANNEL 2
13030 RETURN
13999 '
14000 '   ***   SUBROUTINE : DECREASE OPERATING CURRENT ***
14001 '
14010 CURC = CURC - CURD
14015 IF CURC < 0 THEN CURC = 0
14020 GOSUB 16000: REM CALCULATE OUTPUT VALUES FOR CHANNEL 2
14030 RETURN
14999 '
15000 '   ***   SUBROUTINE : EXIT FROM APPLICATION ***
15001 '
15010 GOSUB 12000: REM TURN OFF POWER
15011 CLS
15012 LOCATE 1, 1
15020 END
15030 RETURN
15999 '
16000 ' *** SUBROUTINE : CALCULATE CHANNEL 2 OUTPUT VALUES ***
16001 '
16010 OX = 4095 * CURC / CURM
16020 O2H% = INT(OX / 16)
16030 O2L% = OX - 16 * O2H%
16040 O2L% = 16 * O2L%
16050 RETURN
16999 '
17000 '   *** SUBROUTINE  :  SET UP PC23 ***
17001 '
17010 CMD$ = "1LD0 2LD0 3LD0 1MN 2MN 3MN 1A10 2A10 3A10 1V1 2V1 3V1 "
17020 GOSUB 1000: REM SEND CMD$ TO PC23
```

```
17030 RETURN
17999 '
18000 '   ***   SUBROUTINE :  SEND ALL THREE AXIS HOME ***
18001 '
18010 CMD$ = "1GH+2 2GH+2 3GH-2 "
18020 GOSUB 1000: REM SEND CMD$ TO PC23
18030 RETURN
18999 '
19000 '   **** SUBROUTINE : POSITION ELECTRODE WITH JOYSTICK ***
19001 '
19020 J11$ = "1J1 ": J21$ = "2J1 ": J31$ = "3J1 "
19021 J10$ = "1J0 ": J20$ = "2J0 ": J30$ = "3J0 "
19030 B12$ = "X and Z": B13$ = "Y and Z"
19040 CMD$ = J10$: GOSUB 1000: CMD$ = J20$: GOSUB 1000: CMD$ = J30$: GOSUB 1000:
19060 CLS : LOCATE 1, 1: PRINT "NASA-MSFC / UAH   Vacuum Welding System": PRINT
19070 PRINT "Use the Joystick to manually align the plates and positon the elect
19080 PRINT "": PRINT "Active axes are "; SCR$
19090 PRINT "": PRINT "": PRINT "Key Functions:": PRINT ""
19100 PRINT "       A :   Move axes X and Z"
19110 PRINT "       B :   Move axes Y and Z"
19120 PRINT "       C :   Exit this screen"
19130 KK$ = INKEY$: IF KK$ = "" THEN GOTO 19130
19140 IF KK$ = "A" OR KK$ = "a" THEN CMD$ = J30$: GOSUB 1000: CMD$ = J10$: GOSUB
19150 IF KK$ = "B" OR KK$ = "b" THEN CMD$ = J20$: GOSUB 1000: CMD$ = J10$: GOSUB
19160 IF KK$ = "C" OR KK$ = "c" THEN CMD$ = J10$: GOSUB 1000: CMD$ = J20$: GOSUB
19170 GOTO 19060
19180 CLS : LOCATE 1, 1: PRINT "NASA-MSFC / UAH   Vacuum Welding System"
19190 PRINT "": PRINT "Do wish to change the position of the electrode?"
19200 KK$ = INKEY$: IF KK$ = "" THEN GOTO 19200
19210 IF KK$ = "Y" OR KK$ = "y" THEN GOTO 19000
19220 RETURN
19999 '
20000 ' *** SUBROUTINE : END OPERATIONS  ****
20001 '
20010 CLS : LOCATE 1, 1
20020 PRINT "NASA-MSFC / UAH    Vacuum Welding System": PRINT ""
20030 PRINT "Key Functions:": PRINT ""
20040 PRINT "    S  : start a new weld operation:"
20050 PRINT "    E  : end this application"
20060 KK$ = INKEY$: IF KK$ = "" THEN GOTO 20060
20070 IF KK$ = "S" OR KK$ = "s" THEN GOTO 5000
20080 GOSUB 15000: REM TURN OFF POWER
20090 RETURN
20999 '
21000 ' *** SUBROUTINE : CALCULATE TRANSLATION INFO ***
21001 '
21010 TV = TS / .2
21020 TV$ = STR$(TV): TVL = LEN(TV$)
21030 V3$ = "3V" + RIGHT$(TV$, TVL - 1)
21040 EV = ES / .2
21050 EV$ = STR$(EV): EVL = LEN(EV$)
21060 V1$ = "1V" + RIGHT$(EV$, EVL - 1)
21070 WD = WL * 125000
21080 WD$ = STR$(WD): WDL = LEN(WD$)
21090 D3$ = "3D" + RIGHT$(WD$, WDL - 1)
21100 ED = EH * 125000
21110 ED$ = STR$(ED): EDL = LEN(ED$)
21120 D1$ = "1D" + RIGHT$(ED$, EDL - 1)
21130 WSEC = WL / TS: REM  CALCULATE WELDING TIME
21140 RETURN
```

```
21999 '
22000 ' ***   SUBROUTINE :   START TRANSLATION ***
22001 '
22010 CMD$ = V1$ + " " + V3$ + " " + D1$ + " " + D3$ + " "
22020 GOSUB 1000: REM SEND CMD$ TO PC23
22030 CMD$ = "1G 3G "
22040 GOSUB 1000: REM  SEND CMD$ TO PC23
22050 RETURN
22999 '
23000 ' ***     SUBROUTINE :   STOP TRANSLATION ***
23001 '
23010 CMD$ = "1S 2S 3S "
23020 GOSUB 1000: REM SEND CMD$ TO PC23
23030 GOSUB 18000: REM  ALL THREE AXES GO HOME
23040 RETURN
```

APPENDIX 2

Listing of Documentation Provided with the System

1. Compumotor DB Drives User Guide

2. DAC-02 User Manual & Software

3. JS5300 Joystick Operation Manual

4. Daedal Limit & Home Switch Cable Pigtail Drawing

5. Compumotor PC23 Indexer User Guide

6. PC-23 Application Developer Disk 1

7. PC-23 Application Developer Disk 2

8. Daedal Inc. MC5300 Indexer Basic Software