AEROTECH, INC.

REFERENCE DOCUMENT

# AEROTECH

# SMART I

# Programmable, Microprocessor-Based

# Two, Three and Four Axis

# Six Digit, Contouring Motor Controls

# Instruction Manual

Rev. 1/17/79

101 Zeta Drive, Pittsburgh, Pennsylvania 15238 • Telephone 412-963-7470 • TLX 710-795-3125

TABLE OF CONTENTS

TABLE OF CONTENTS CONT. (PAGE 2)

TABLE OF CONTENTS CONT. (PAGE 3)

## 1.0 GENERAL INFORMATION

### 1.1 INTRODUCTION

SMART is a microcomputer controlled, parallel word absolute/incremental, stored program contouring system. This system utilizes any of Aerotech's Stepping Motor or Encoded DC Motor Drives. One* chassis contains all electronics, including servo controllers or translators, for a complete 2-axis system.

* DC servo controllers larger than 6020 require an additional chassis or plate for power amplifiers mounting.

## 1.2 FEATURES

### 1.2.1 STANDARD FEATURES

Manual Data Input-keyboard entry

Absolute/Incremental Programming-mixed

Programmable Speed, 4 digit

Manual Control-run, step

Six Digit Commands

Home (Reference Zero)

Home Offset (Full Floating Zero)

99 M Functions

Programmable Dwell-.001 to 99.999 seconds

1000 Byte Program Storage-250+ Commands

Editor

Search

Auto/Single Block Operation

Programmed Stop

2 axis Simultaneous Contouring

Display-input commands in MDI or EDIT, block number in AUTO or SINGLE

Test RAM and PROM memory

Subroutines

Multiple Program Storage

Error Message Display

Program Length Display

### 1.2.2 OPTIONS

Position Readout

8000 Byte Program Storage

16000 Byte Program Storage

Tape Reader Input

Magnetic Tape Read/Write

Latched M Function and Isolator

Mirror Image

Manual Feedrate Override

Corner Rounding

Serial I/O, TTY or RS 232C

## 2.0 INSTALLATION INSTRUCTIONS

### 2.1 UNPACKING

Unpack the system from its shipping container and, referring to the sales order, verify that all of the items are present. Save the packing material for storing and reshipping the system. If the shipping container is damaged upon receipt, request that the carrier's agent be present while the system is being unpacked and inspected.

### 2.2 INSPECTION

The system should be inspected upon receipt for broken, damaged or loosened parts. Retighten any loosened connectors.

### 2.3 INSTALLATION

Install all motor connectors and cables before applying power to the system. If the system includes an Aerotech stage, refer to the stage manual and remove shipping clips; all adjustments have been made at the factory and no further adjustments necessary. If the system does not include a stage, the drives will have to be adjusted for proper response to the load inertia and friction; refer to Section 5 and the drive manuals.

#### 2.3.1 AC POWER REQUIREMENT

The standard system requires 115 VAC ± 10%, 50-60 HZ. Chassis with internal servo controllers and translators have the 115 VAC line fused at 5 amp. This is the maximum the SMART chassis will require. Units with Servo Controllers larger than can be housed in the chassis, will require additional power for the Servo Controller; the power requirement for the individual controllers is given in the servo controller manual.

## 2.4 CUSTOMER INTERFACE

The customer interface consists of the M function and IN PSN outputs and a CONTINUE input.

### 2.4.1 M FUNCTIONS

The rear panel connector has logic outputs (5V logic capable of driving two low-power TTL loads, one low-power schottky TTL load or two HTL loads) with the capability of exercising 99 M functions. The outputs are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 00, 10, 20, 30, 40, 50, 60, 70, 80, and 90.

This permits 18 outputs with no decoding, or 99 outputs with minor decoding (connecting 9 and 20 to an AND gate to generate M29).

The standard outputs will go hi as soon as the M function is read by the Control, and will remain hi until a different M fcn. is read. If more than one M function appears in a block, the first one read will go hi only for a few microseconds (until the next one is read). If it is desired to have a longer time hi, insert a Dwell of the desired time after the first M fcn.

### 2.4.2 MO/CONTINUE

The rear panel connector containing the M function outputs also contains the IN PSN and CONTINUE. The IN PSN line goes hi (Open Collector driver with 2K pull-up capable of sinking 0.2 amp) when the Drive has completed the move (is In-Position).

When MO is read in the program, the system will stop until a CONTINUE command is received on the rear panel interface. If Ml is read in the program, the system will stop until ENTER is pressed on the keyboard.

The $\overline{\text{CONTINUE}}$ line should be pulled lo for at least 3 m.s since there is debouncing circuitry monitoring the line for a continuous low for about 2.6 milliseconds.

2.4.3 IN PSN

The IN PSN output is buffered and used to notify the user that the motors are in their commanded position. This open collector output has a 2K ohm pull-up resistor and is capable of sinking 0.2 amp.

2.5 LIMITS

If the system includes an Aerotech stage, the limits have already been wired. If not, they must be connected to limit motor travel when required. They must be connected if HOME is to be used. The CW limit must provide a signal COM to the CW limit input when the motor is at the maximum CW position (CW is defined looking into the mounting flange of the motor). The CCW limit must also provide a common to the CCW limit input when the motor is in the extreme CCW position; this CCW limit is also required for the HOME circuit to function.

For systems without an Aerotech stage, the limit appears on the 6 pin connector, with one foot wire length, coming out of the motor housing. Refer to stage wiring Drawing C630-1045 for connections.

## 3.0 OPERATION

The operation of the eight system modes are discussed in this section.

### 3.1 EDIT

The Edit Mode is used to examine memory, erase commands, and search for blocks or commands. Searching is also used to count the number of bytes used in programming.

#### 3.1.1 START

Pressing the start key places the memory pointer to the start of memory.

#### 3.1.2 STEP

This key steps to the next command for display. Pressing - causes the previous command to be displayed.

#### 3.1.3 ERASE

Causes the command, which is shown on the display, to be erased from memory.

#### 3.1.4 SRCH

The search key is upper case and requires that the SHIFT key be pressed (and released) prior to pressing search. The displayed command will be searched for, and when found, the number of bytes from the beginning of memory to that command will be displayed. The searched for command must be entered into the display in the MDI mode and cannot be an M, G or EOB.

### 3.2 MDI

The manual data input mode is used to enter data into the display, and/or memory. When searching for a command or block number, data is entered only into the display.

3.2.1 START

Pressing START places the memory pointer at the start of memory so that the following data will be entered at the beginning of the memory storage. Data which is already in memory will not be erased; it will simply be pushed toward the end of memory.

3.2.2 ENTER

Pressing ENTER enters the displayed data into memory and clears the display.

3.2.3 COMMANDS

All commands must be preceded by an address such as X, Y, D, N etc. These letters (Addresses) are all upper case. Upper case characters are obtained by pressing the SHIFT (↑) key first (and releasing it) and then pressing the desired character. The letter will then appear on the LED lamp. If a negative command is desired, it should be entered next (the negative light will come ON). The numerical data should then be entered next and will appear on the display. If zero numerical data is desired, it need not be pressed (such as MOO).

3.2.4 ERASE

After entering data into the display, but prior to entering it into memory, it can be erased from the display with the ERASE key. In the MDI mode, data cannot be erased from memory.

3.2.5 EOB

Terminates all the commands to be executed in that block of information.

3.3 MANUAL MODE

The manual mode allows motor positioning via the keyboard ↑ ↓ → and ← commands which drive the Y motor CW, CCW, X CW and XCCW, respectively. The absolute position of the motors is not kept track of, as it is in the AUTO or SINGLE mode. Pressing → in the Manual Run mode will cause the X motor to turn CW until the key is released (if any other key is pressed before → is released, the motor will continue CW). Pressing → in the Manual Step mode will cause the X motor to take a single step in the CW direction.

Pressing any of the above buttons in the MANUAL RUN mode causes a lo speed run. A medium speed run will occur when the arrow button is pressed and then also press ERASE. A high speed run will occur when an arrow button is pressed and STEP. A top speed run will occur when an arrow button is pressed and -.

3.4 AUTO

The Auto mode places SMART in the automatic mode whereby, once initiated, it continues executing the program until MO, M1 or M2 is encountered.

3.4.1 START

Pressing START places the memory pointer at the beginning of memory and starts executing the program.

3.4.2 ENTER

Pressing ENTER leaves the memory pointer at its present location and starts executing the program from there.

3.4.3 RESET

The Reset pushbutton interrupts the operation of any mode, resets the drives and readouts, and awaits further commands from the keyboard.

### 3.5 SINGLE

In the Single Mode, one block of data is executed each time a key is depressed. A block of data consists of all the commands between End-Of Blocks (EOB).

#### 3.5.1 START

The Start key places the memory pointer at the beginning of memory and executes the first block of data (all the data from the beginning of memory to the first EOB).

#### 3.5.2 ENTER

The Enter key enters memory at the location indicated by the memory pointer and executes all data until an EOB is encountered. To step thru the entire program one block at a time, press START, wait for the move to be completed, then press ENTER, wait for the move to be completed, then press ENTER, etc. The ENTER key can be considered a continue command.

### 3.6 EXTERNAL

The External mode is used for optional input and output of programs. Refer to the appropriate option addendums.

### 3.7 TEST

The Test mode is used to test SMART.

#### 3.7.1 CHECK RAM MEMORY

Pressing 0 causes all RAM memory storage to be tested. The memory is then left cleared (all zeros). If a bad memory location is discovered, its location is output on the display. If no bad memory locations are discovered, the end of memory is displayed. For example, with the standard 1K memory, if 1023 is displayed, all memory is good. If 735 is displayed, that memory location is defective.

3.7.2 CHECK ROM

Pressing 1 initiates a checksum of ROM memory to verify that the executive program has not changed. N-90000 will be displayed if the check is good. N-90009 will be displayed if the check shows an error.

3.7.3 CHECK AND CLEAR RAM SCRATCHPAD

Pressing 2 checks and clears scratchpad RAM. These memory locations are used for internal SMART calculations and storage such as the absolute position registers. 126 is displayed when the check is good. Any number lower than 126 indicates that memory location is bad.

3.6.4 CHECK RAM

Pressing 3 is similar to the 0 test except memory is not cleared to zero: it is left intact. Therefore this test can be done without destroying programs in memory.

## 4.0 PROGRAMMING

### 4.1 ENTERING PROGRAMS AT THE BEGINNING OF MEMORY

Programs are entered at the beginning of memory by switching to the MDI mode and pressing the START key. The first command entered will then be placed in the first memory location, the second command entered will be placed in the second memory location, etc. Any commands previously stored will be shifted toward the end memory.

The code M30 is usually used to notify SMART that this is the last command in memory and is required for external I/O of programs.

### 4.2 ENTERING MULTIPLE PROGRAMS

Additional programs may be entered in front of, or after, previously stored programs. To enter an additional program at the beginning of memory, refer to 4.1. To enter a program (PGM 2) after a previously entered program (PGM 1); search or step to the end of PGM 1, DO NOT PRESS START, switch to the MDI mode and enter PGM 2.

```
Example: N1 ... EOB
         .
         .          PGM 1
         .
         N99 M2 EOB
```

Search for N99, STEP until EOB is displayed on the LED, then switch to MDI and enter PGM 2.

### 4.3 ADDING, CHANGING OR DELETING COMMANDS

The program can be easily changed by using the EDIT and MDI mode. Commands are examined, deleted and searched for in the EDIT mode; and entered in the MDI mode.

### 4.3.1 ADDING A COMMAND

To add a command, display the previous command in the EDIT mode. Switch to MDI and enter.

Example: It is desired to enter a 1 sec. delay before the X1000 Y-200 move in block N50.

Present Program

| | | | |
|---|---|---|---|
| N49 | M20 | X300 | EOB |
| N50 | X1000 | Y-200 | EOB |
| N51 | | | |

First search for the block

MDI: SHIFT N 5 0

Note: The above means in the MDI mode you pressed the key shown: SHIFT, N, 5, 0. If the display had not been cleared, precede the above with ERASE.

EDIT: SHIFT SRCH

SMART will search for N50 and display the number bytes it took to find it.

Press STEP

N50 will be displayed

Any data entered now will follow N50 and precede X1000.

Enter the 1 sec. delay

MDI: SHIFT G 4 ENTER SHIFT F 1000 ENTER

The program is now:

N50 G4 F1000 X1000 Y-200 EOB

### 4.3.2 DELETING A COMMAND

To delete a command, display the command in the EDIT mode and press ERASE.

Example: It is desired to delete the command Y-200 in Block N50.

Present program

N50 Y-200 X1000 EOB

Search for the command.

MDI: ERASE SHIFT N 5 0

EDIT: SHIFT SRCH

After the command is found, press STEP twice and Y-200 will be displayed. Press ERASE.

The command will be erased from the display and memory.

The program is now:

N50 X1000 EOB

As can be verified in the Edit Mode.

### 4.3.3 CHANGING A COMMAND

To change a command, display the command and ERASE it, then enter the new command.

Example: It is desired to change the command X1000 to X1500 in block N50.

Present Program

N50 X1000 Y-200 EOB

Search for the command

MDI: ERASE SHIFT N50

EDIT: SHIFT SRCH

After the command is found press STEP twice. X1000 will now be displayed. Press ERASE. Then switch to MDI and enter the new command:

MDI: SHIFT X 1500 ENTER

The program is now:

N50 X1500 Y-200 EOB

as can be verified in the Edit Mode.

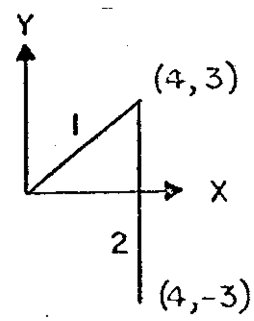
## 4.4 PREPARATORY CODES

### 4.4.1 LINEAR INTERPOLATION, G1

The two programming functions are linear and circular programming. In a linear motion, each axis maintains a constant speed throughout the move. For circular motion the speeds of each axis are varied sinusoidally to produce an arc or circle.

LINEAR INTERPOLATION, G1

In linear interpolation, a straight path is traversed from the present position to the commanded position. Consider a program to follow the path shown in Figure 1.




The first move is 4 inches for X and 3 inches for Y. The second move is 6 inches for Y. If the drive is 1 mil/step (1000 steps per inch), the first command is

X4000 Y3000

and the distance traveled is $\sqrt{4^2 + 3^2} = 5$ inches.

The feedrate number for a system with 1 mil resolution*

$$FRN = K \frac{IPM}{L}$$

where FRN = 4 digit feedrate number.

IPM = traverse speed in inches per minute

L = distance traversed in inches

K = 0.1 for 1 or 2 digits

| | |
|---|---|
| 1 | for 3 digits |
| 10 | for 4 digits |
| 100 | for 5 digits |
| 1000 | for 6 digits |

If the desired speed is 20 IPM:

$$FRN = 10 \frac{20}{5} = 40$$

For F40 as the speed command the program would be:

```
G1     X4000     Y3000     F40     EOB
       Y-6000              F33     EOB
M2
```

G1 prepares the system for linear interpolation. F40 sets the speed at 20 IPM for the move. X4000 commands 4000 steps in the +X direction Y3000 commands 3000 steps in the +Y direction. The first block of data consists of the entire first line. In SINGLE Mode, only one block of data is executed per key depression. In the second block of data, F33 sets the speed at 20 IPM for a six inch move. Y-6000 commands 6000 steps in the -Y direction. After the second block of data is executed, the code M2 is encountered and the memory pointer resets to the beginning of memory.

### 4.4.2 CIRCULAR INTERPOLATION, G2 OR G3

G2 is the preparatory command for a CW arc and G3 for a

* For systems with 0.1 mil resolution, multiply the calculated FRN by ten.

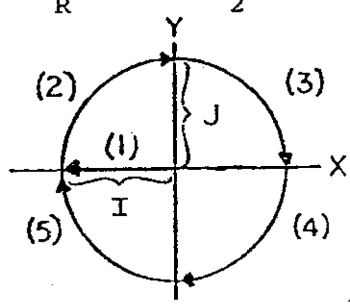
CCW arc. Consider a program to traverse a two inch radius circle shown in Figure 2 at 20 IPM for a drive with 1 mil steps, starting at zero.

When G2 or G3 are used, arc center offsets I and J must be specified: these are the distances from the starting point of the arc to the center of the circle. I is the offset along the X axis and J is the offset along the Y axis.

The feedrate number of circular interpolation is similar to that for linear interpolation except that the radius of the arc must be used and K is determined from the maximum digit I or J.

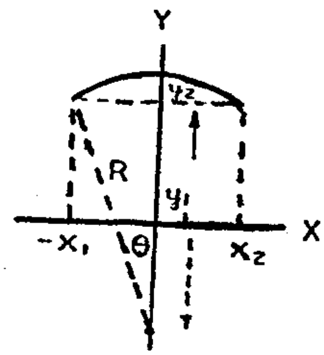
The feedrate number:

$$FRN = K \frac{IPM}{R} = 10 \frac{20}{2} = 100$$




| | | | | |
|---|---|---|---|---|
| G1 | G91 | X-2000 | F100 | EOB |
| G2 | X2000 | Y2000 | I2000 | EOB |
| | X2000 | Y-2000 | J2000 | EOB |
| | X-2000 | Y-2000 | I2000 | EOB |
| | X-2000 | Y2000 | J2000 | EOB |
| | M2 | | | |

Comments: In the first block G1 prepared for the linear move, F100 sets the 20 IPM speed for a 2 inch move. In the second block the G2 code prepared for CW arc and was not repeated for the remaining 3 blocks, F and G commands remain in effect until cancelled by later F or G commands. F is 100 for a 20 IPM feedrate with 2 inch radius. The distance traveled is 2000 steps in X and Y for each arc. The arc in the second and fourth blocks of data have an arc center offset, with respect to starting point, only in the X direction; therefore, I = radius and J = 0. Note that $R^2 = I^2 + J^2$. As another example it is desired to traverse the arc shown in Figure 3.




The radius R = 9 inches and $X_1 = X_2$ = 2 inches

$\theta = \sin^{-1} \frac{X1}{R} = \sin^{-1} \frac{2}{9} = 12.8^{\circ}$

$Y_1 = R \cos \theta = 9 \cos 12.8^{\circ}$ = Y axis arc center

= J = 8775

$Y_2 = R - Y_1 = R\ (1-\cos\theta) = 9\ (1-\cos) = 225$

= Y distance to be traveled.

The radius is 9000 steps.

If a feedrate of 20 IPM is desired

$$FRN = 10\ \frac{20}{9} = 22$$

And the program would be:

```
G2  X2000     Y225     I2000     J8775     F22    EOB
X2000     Y-225     J9000     EOB
M2
```

### 4.4.3 DWELL, G4

The dwell command is used to insert timed delays from .001 to 99.999 seconds in the program. The command following G4 contains the magnitude of the dwell (F1000 means 1000 milli-seconds or 1 second delay). When a dwell is encounterdd, the delay occurs before the rest of the block is executed.

Example:

```
G7     G4     F1000     X100     EOB
```

will cause the drive to go to the home position, then dwell for 1 second, then execute X100 steps in the CW direction.

G4 F1000 G7 X100 EOB

will cause a 1 sec. dwell, then the drive will go to the home position, then X will index 100 steps.

4.4.4 RESET ABSOLUTE POSITION REGISTERS, G5

G5 is used to reset the Absolute Position registers for indexing the Absolute mode. All absolute commands following G5 will use the location where G5 occurred as the zero reference.

4.4.5 HOME (REFERENCE ZERO), G7

The home command, G7, sends the drive to the home or reference position. The drive finds the home position by driving in the CCW direction until the CCW Limit Switch is closed. Then the motor reverses direction and begins rotating CW. It continues CW until the Marker Pulse is encountered. The first Marker Pulse after the limit resets the system and this establishes the home position.

Home Offset

Establishing a Home Offset or floating reference is accomplished by a procedure, rather than a special command. The procedure is: go to the Home Position, index the desired offset, reset the Absolute Position Registers (G5). It is optional whether the Position Readouts are reset or not, if it is desired that the Position Readouts agree with the Absolute Position Registers in SMART Memory, the G10 command should follow G5. Home Offsets must be placed immediately following the home command, in the same block. Typically, Home and Home Offset are placed in the first command block so that the Home Offset can easily be changed.

Example: Assume that the SMART is positioning a 4 inch stage with .001 inch travel per step and it is desired to place the Home Offset in the center of travel (approx. 2 inch from CCW end).

The first commands would be:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| N1 | G91 | G7 | G1 | X2000 | | F1000 | EOB |
| N2 | | G5 | | G10 | G90 | X1000 | EOB |

The first block (N1) would send the stage to the home position (G7). After the stage reached the home position, the X axis would move 2 inches (X2000) in the Incremental Mode (G91) at 200 IPM (F1000). The absolute mode (G90) should not be used until the Absolute Position Registers are reset (G5). After the drive reached the offset position of X2000, the second block (N2) would be executed. The Absolute Position registers in SMART Memory would be reset to zero by G5, then the Position Readouts would be reset to zero by G10 (if the Readout Option were included), then the system would go to the Absolute Position of X1000. The stage will then be 3 inches from the Home Position and the internal Absolute Position Registers are at +1000 steps.

The code G90 is not required in the second block unless absolute programming is desired. To change the Home Offset Position (the location of which the program starting at N2 begins) simply change X2000 in block N1 to the desired location. If the MIRROR OPTION is included, the Home Offset will ignore mirror commands as long as they are in the same block as the home command, G7, and follow it.

4.4.6 RESET POSITION READOUTS AND DRIVES, G10

G10 sends a reset pulse to the drives, which in turn sends a reset pulse to the readouts. The readouts will read all zeros when reset.

4.4.7 AXIS SELECT, G17-G20

| | |
|---|---|
| G17 | X, Y |
| G18 | X, Z |
| G19 | Z, Y |
| G20 | Z, Ø |

These commands select which two axis are to receive positioning commands. All axis receive the Home (G7), Reset Absolute Position (G5) and Reset Readout (G10) commands regardless of the axis select in effect. When no commands are given, G17 is selected by default.

When contouring, the first axis above is allocated the I offset and the second receives the J offset. For example if G18 were in effect for an arc, the offset from the center of the circle to the starting point along the Z axis would be programmed as J, and the offset from the center of the circle to the starting point along the X axis would be I.

4.4.8 CORNER ROUND/NORMAL POSITIONING, G23/G24

Corner Rounding or normal positioning is selected with preparatory codes G23 or G24. G24 selects normal positioning whereby the drives get into position before the next block of data is executed. G23 selects corner rounding whereby the drives get their next command before they get into position and while they are running the following error to zero; this keeps the drives running at near constant velocity.

Programs run with G23 will not be as accurate as those with G24, but will run faster since time will not be spent waiting to take up the following error (decelerate to zero speed) or accelerate when contouring with circular interpolation.

G24 remains in effect until cancelled by a G23 command and vice versa. When neither command is specified, G24 is selected by default.

### 4.4.9 THREADING/NORMAL SPEED CONTROL, G33/G36

The four digit Feed Rate Number controls the axis speed in the Normal (G36) Mode.

In G33 Mode (with the Threading Option) the axis speed will be dependent on the spindle speed (since the axis movement must be locked to the spindle movement) and the I or J command. The lead will be specified by the I or J command: for lathe work the X and Z axis will be used (G18) and the thread lead will be an I command.

For example assume it is desired to make a five pitch (.2 inch lead) thread, four inches long. The command would be (for 100 u.in. resolution).

G36 Z40000 I200000 EOB

The G36 command placed the system in the threading mode. The thread length of four inches required a Z40000 command (4.0000 inches). I200000 specified a .2 lead. The speed of the Z move in IPM will be the lead times the spindle RPM: if the spindle speed was set for 1000 RPM, the Z axis speed would be .2 X1000 = 200 IPM (inches per minute). All moves in G36 mode are synchronized to the Marker Pulse (a pulse which occurs once per revolution) on the spindle encoder; that is, the marker pulse will initiate the move. This permits successive cuts to be started at the same point on the thread, but also requires thread reliefs at the ends of the thread.

### 4.4.10 ABSOLUTE/INCREMENTAL PROGRAMMING, G90/G91

G90 places SMART in the Absolute Programming Mode so that all axis commands are with respect to the position stored in Absolute Position Registers. The Absolute Position Registers will keep track of the motor position in all but the MANUAL Mode. Absolute positioning commands must never exceed an incremental index of 999,999 steps. For example, if the Absolute Position is + 600,000, a command to go to the position -600,000 is not allowed because it would require an incremental index of -1200,000 steps which exceeds the 999,999 limitation.

INCREMENTAL PROGRAMMING, G91

G91 places SMART in the incremental mode so that all axis commands are with respect to the present position. The command X1000 in the incremental mode will cause the motor to take 1000 steps in the CW direction, regardless of where the present position is.

### 4.4.11 REPEAT FROM BEGINNING OF MEMORY, G99

G99 will cause the program to return to the beginning of memory and start executing from there.

For Example:

```
N1   X1000   F500   EOB
N2   X-1000         EOB
N3   G99
```

Will cause the drive to cycle endlessly thru N1, N2, N3, N1, N2, etc.

## 4.5 MISCELLANEOUS CODES

### 4.5.1 PROGRAMMED STOP, MO

MO is used in the AUTO Mode to stop the program. A lo applied to the rear panel CONTINUE will cause the program to continue from the location it has stopped, and cause the MO output to return lo. As soon as MO is read in a block of data, the system will wait until the CONTINUE input goes lo before continuing.

### 4.5.2 PROGRAM STOP, M1

M1 is used in the automatic mode to stop the program. Pressing ENTER will cause SMART to enter the program at the point it had stopped and continue executing data.

### 4.5.3 END OF PROGRAM, M2

M2 is used in the automatic mode to stop program execution and set the memory pointer to the beginning of memory.

### 4.5.4 END OF TAPE, M30

The M30 signals SMART that this is the end of the program when interfacing to a Paper Tape Reader, Mag Tape Read/Write, or Serial I/O. It is recommended the command always be placed at the end of all programs, even those entered in MDI mode.

```
MDI:  START
      SHIFT   M   3   0   ENTER
      START
```

Now enter your program. M30 will be pushed along in memory but stay at the end of the program.

### 4.5.5 MIRROR IMAGES, M57-M60

| | |
|---|---|
| M57 | MIRROR X |
| M58 | MIRROR Y |
| M59 | MIRROR Z |
| M60 | CANCEL MIRROR IMAGES |

These miscellaneous codes allow mirror images to be programmed. Entire programs can be written as subroutines with the mirror image commands specifying which axis are mirrored during the execution of the subroutines. Note that these commands remain in effect until cancelled: M60 will cancel M57. If M57 is deleted from the program after it has been run once, the mirror command is still stored, and will only be cancelled by a M60 or a MODE TEST 2 Command (Clear Scratchpad).

### 4.5.6 RETURN FROM SUBROUTINE, M99

This code tells SMART that this is the end of the subroutine and to return to the main program.

## 4.6 FEEDRATE AND ARC OFFSETS, F, I, J

### 4.6.1 FEEDRATE NUMBER, F

The Feedrate Number is used to specify feedrates or motor speed. F can range from 1 to 9999.

$F = K \frac{IPM}{L}$ for Linear Interpolation, G1

$= K \frac{IPM}{R}$ for Circular Interpolation, G2 or G3

K = a constant depending on the maximum number of digits in I or J for Circular Interpolation, or X or Y for Linear Interpolation.

IPM = Speed in inches per minute (or steps per minute),*

R = Radius of arc in inches (or steps),

L = Length of path $\sqrt{X^2 + Y^2}$ in inches (or steps),

| K | Digits |
|---|---|
| .1 | 1 |
| .1 | 2 |
| 1 | 3 |
| 10 | 4 |
| 100 | 5 |
| 1000 | 6 |

* Note: If steps are used, the units for R and L must also be steps; for .1 mil. resolution, multiply the calculated F by 10.

If an arc with a two-inch radius were made at 100 IPM:

$$F = 10 \frac{100}{2} = 500 \text{ for 1 mil resolution, or}$$

$$= 5000 \text{ for .1 mil resolution.}$$

For linear interpolation a 4 inch X move at 200 IPM would require:

$$F = 10 \frac{200}{4} = 500 \text{ for 1 mil resolution}$$

or 5000 for .1 mil resolution.

If X and Y are moving simultaneously and the vector velocity is specified, the path length is $\sqrt{X^2 + Y^2}$

For example a 7 inch Y move simultaneous with an 8 inch X move at a vector velocity of 200 IPM would require:

$$L = \sqrt{X^2 + Y^2}$$

$$F = 10 \frac{200}{10.63} = 188 \text{ for 1 mil resolution}$$

or 1880 for .1 mil resolution.

With high resolution systems at high speeds, the desired feedrate may not be obtainable with the 4 digit F number, i.e. the calculated F may be greater than 9999. If this occurs, the speed may be increased for linear interpolation contouring only (G1) by inserting I and J numbers with appropriate scaling (I = CX, J = CY where C equals a constant). For example, assume it is desired to travel at 400 IPM with a .1 mil resolution system with a .1 inch X and .2 inch Y move.

$$F = (10)\ 1 \frac{400}{.2236} = 17890 \text{ which is more than 4 digits.}$$

The true speed can be doubled by adding the I and J commands

J20000 I40000 ( twice the movements in X and Y respectively)

Now half the F can be used (17890 ÷ 2 = 8945).

The program would read:

G1 X10000 Y20000 J20000 I40000 F8945 EOB

which would yield the 400 IPM speed. Note that I and J numbers are not normally required for programming linear interpolation.

4.6.2 ARC OFFSET, I

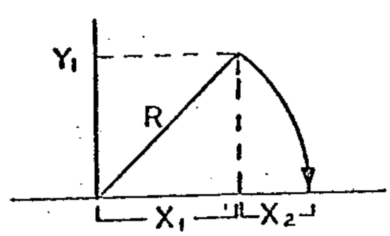
I commands can be up to six digits and are used when G2 or G3 are in effect. This command is the distance along the X axis for G17 or G18 mode (or Z axis in G19 and G20 mode) from the starting point of the arc to the center of the circle. A negative sign signals a digit overflow (See 4.6.4).

4.6.3 ARC OFFSET, J

J commands can be up to six digits and are used when G2 or G3 are in effect. This command is the distance, along the Y axis for G17 and G19 mode (or Z axis in G18 mode) from the starting point of the arc to the center of the circle. A negative sign signals a digit overflow (See 4.6.4).

4.6.4 DIGIT OVERFLOW ON I, J

A digit overflow condition can occur when programming arcs whose radius have a most significant digit of seven or greater, and which start around the 45, 135, 225 or 315 degree points and rotate 45 degrees or far enough to increase the number of digits. For example, if it is desired to traverse the arc shown:




$J = Y_1$ and $I = X_1$

If R = 1200, for the $45^{\circ}$ starting point shown,

$I = J = R \sin 45^{\circ} = R \cos 45^{\circ} = 849$

When the arc is completed at point 2, I will be 1200: it will have increased from a three to a four digit number. Therefore I should be programmed as a negative number (I - 849), and the F number increased by a factor of 10.

4.7 RESET

The RESET pushbutton on the front panel is used to interrupt the system from its present task and send it back to the Mode Switch and keyboard for further commands.

4.8 BLOCK NUMBERS, N

Block Numbers are not required but will save time in changing or checking programs. They are also displayed when that block of axis commands is being executed.

Example 1:

N100 X1000 Y200 EOB

While the motors are positioning (X1000 steps and Y200 steps, incremental mode) the display will show N100.

Example 2:

N50 G04 F1000 X1000 Y200 EOB

The previous block number will be displayed during the 1 second dwell, but while X and Y are executing the block commands, the display will show N50.

4.9 SUBROUTINES, N-

Subroutines are used to reduce programming time and memory for repetitive operations. The subroutine is called by the N- prefix and must be located after the main program M2. The end of the subroutine is identified by M99. The subroutine must be written after M2 and use the same number that followed the N- prefix, but it now must have simply an N prefix.

Example:

```
N1        ...          EOB
                           |------- MAIN PROGRAM
  .
  .
  .
N29       ...
N30       N-8000       EOB      <---------Subroutine Call
N31       ...          EOB
  .
  .
  .
N99       M2           EOB
N8000     ...          EOB
N100      ...          EOB  ------- SUBROUTINE
  .
  .
  .
N150              M99          <-- Return from Subroutine Command
```

The program will begin at N1 when START is pressed in the AUTO mode and execute the N29 block. Block N30 calls the subroutine N8000 and SMART will jump to the block N8000 following block N99 and execute the subroutine. Block N150 contains the Return code which causes the program to jump back and begin executing from the block N31. When block N99 is read, M2 will cause the program to go back to the beginning of memory and await further commands.

Multiple subroutines can be used and called any number of times, but they must appear following M2 of the main program (otherwise they will be executed with the normal flow of the program).

The EOB commands are not required at the end of Blocks N30, N99 or N150 since the command preceding these does not require an execute signal which the EOB causes. However, for the printout option, the EOB command will cause a carriage return and line feed.

## 4.10 REPEATS

The Repeat command is a powerful programming tool to reduce programming time, memory and tape requirements. This six digit special code allows any command or subroutine to be repeated up to 99 times. The Repeat command is N99 AA BB where:

N99 Specifies the repeat operation.

AA Is simply an operator inserted number similar to an ordinary N number.

BB Specifies the number of repeat cycles.

For example:

N990015 X100 EOB

Instruct the control to move X100 steps and repeat the move 15 times (total of 16 moves or 1600 steps).

N990015 G04 F1000 X100 EOB

Instructs the control to again index X 16 times, but also dwell one second before each index.

N990005 N-100 EOB

Instructs the control to repeat the subroutine, N100, five times.

During execution of the repeats, the Display will show N99 AA CC; where CC is the number of repeats left to perform.

The following program illustrates the use of Repeat command in a Threading operation.

```
N80   G1   G36   G23   G7   X-13930   Y7000 F3000 EOB
N81   N99 80 24   N-600   EOB
N82   N99 81 01   N-700   EOB
N83   N99 82 02   N-800   EOB
N84   N99 83 04   N-900   EOB
N85   N99 84 01   N-1000  EOB
N86   M2

N600   G1   G36   X-2100   F3000   EOB
N601   G33   Y20000   I16000   EOB
N602   Y1410   X1410   J16000   I16000   EOB
N603   Y 5000   I 16000   EOB
N604   X1250   Y1250   I16000   J16000   EOB
N605   Y-27660 X-660   J100000   I100000   EOB
N606   M99

N700  G1  G36   X-2010   F3000   EOB
N701  G33   Y20000   I20000   EOB
N702  Y1410   X1410   J   I2000   EOB
N703   Y5000   I2000   EOB
N704   X1250   Y1250   I4000   J4000   EOB
N705   Y-27660   X-660   J10000   I10000   EOB
N706   M99

N800   G1   G36   X-2050   F3000   EOB
N801   G33   Y20000   I 31250   EOB
N802   G36   X 2000               EOB
N803   Y-20000                    EOB
N804   M99
```

```
N900    G1    G36   X-2005    F3000    EOB
N901    G33   Y 20000    I 31250       EOB
N902    G36   X 2000                   EOB
N903    Y-20000                        EOB
N904    M99

N1000    G1    G36   X-2000    F3000   EOB
N1001    G33   Y20000    I31250        EOB
N1002    G36     X2000                 EOB
N1003    Y-20000                       EOB
N1004    M99
```

PROGRAM EXPLANATION

Block N80

G1-Linear Interpolation

G36-Normal Positioning (speed dependant on F code)

G23-Corner Rounding: do not wait till drives get into position before executing next block of data.

G7-Go Home (to reference position)

X-13930 Y7000-Home Offset

F3000-Feedrate Number

EOB-End Of Block

Block N81

N99 80 24-Repeat block 24 times, N99 specifies the repeat operation, 80 is a number distinguishing which repeat command is being referenced, 24 is the number of repeats.

N-600 -Subroutine Command, execute subroutine N600.

Block 81 commands the system to execute all the data in the N600 subroutine 25 times (10 mil cuts).

Block N82 thru N85 are similar to N81, but call other subroutines different numbers of times.

Block N86

M2-End of Program, stop executing blocks and return to the beginning of memory.

Block N601

G33-Threading, speed dependant on spindle speed and I command.

Y20000-2 inch Y clockwise command (20000 steps).

I16000-sets speed at 16 mil per rev. or 19.2 in/min at a spindle speed of 1200 RPM.

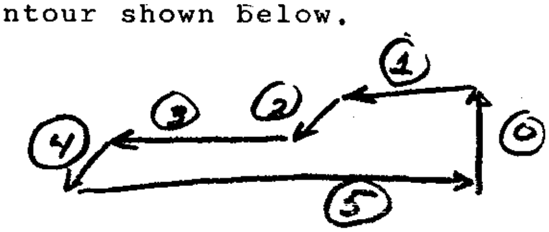
Block N605

I 100000-sets Y return speed at 120 IPM with 1200 RPM spindle speed.

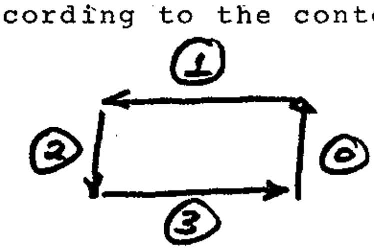
Block N606

M99-End of Subroutine code, return to main program.

The N700 subroutine is similar to the N600 subroutine except that N600 takes a 10 mil cut (at a speed of 16 mil per rev) and N700 takes a 1 mil cut (at 2 mil/rev) according to the contour shown below.




Subroutines N800 and N900 do the actual threading with N800 taking a 5 mil thread cut and N900 taking a 0.5 mil thread depth cut according to the contour shown below.




Block 801

G33-Threading command.

Y2000-2 inch thread length.

I31250-32 threads per inch $\left(\frac{1000000}{32}\right)$.

Block 802

G36-Normal Speed Control, F (feedrate number) controls speed. F3000 from Block N800 remains in effect for a commanded speed of 60 IPM $10 \times 10 \times 60 \div 2 = 3000$

## 4.11 TOOL OFFSETS

The number and type (single or pairs) of tool offsets can be selected by the user; up to the memory capability of the system. Offset magnitudes are up to the full indexing capability of the system. Offset directions can be either + (CW) or - (CCW motor rotation).

Once programmed, the tool offsets can be used by any or all programs in memory. The tool offsets can also be read and/or stored via any of the standard SMART interfaces: Magtape, papertape, TTY, or RS232. The tool offsets can be entered, examined, deleted and searched for.

### 4.11.1 Tool Offset Programming

Tool Offsets are accomplished with subroutines. This allows any number of offsets or offset pairs to be programmed. Assume the following offsets are desired:

| Tool Offset Sequence Number | Offset Axis/Direction/Distance |
|---|---|
| N511 | X - 200 |
| N512 | X - 500 |
| N513 | X - 100 Y - 300 |

The tool offset programming can be done in two steps whereby each tool offset has a corresponding offset cancel, or it can be done in an incremental fashion for minimum time. Both methods will be illustrated.

The program for Tool Offsets with cancels would be:

```
N511   X-200   EOB
 M99
N521   X200    EOB
 M99
N512   X-500   EOB
 M99
M522   X500    EOB
 M99
M513   X-100   Y-300   EOB
 M99
M523   X100    Y300    EOB
 M99
```

To call Tool Offset 1, the program need only be N-511, to cancel Tool Offset 1, program N-521. To call offset 2, program N-512; to cancel offset 2, program N-522. Etc. Note that for contouring systems not in the Linear Interpolation mode when the subroutine is called, G1 must also be added to the subroutines. Feedrate numbers should be included to program the speed of the offset. Multi (more than two) axis systems will also require the proper code (G17-G20) to select the desired axis pair. Only two axis may be offset per block, but the subroutine can contain many blocks so that all four axis can be offset by a single subroutine.

A program using the Tool Offsets with cancel would be:

```
N1    G1    G91    G7    F500    EOB
N-511    Inserts Tool Offset 1.
Program using Tool Offset 1.
N-521   Removes Tool Offset 1.
N-512   Inserts Tool Offset 2.
Program using Tool Offset 2.
N-522   Removes Tool Offset 2.
N-513   Inserts Tool Offset 3.
Program using Tool Offset 3.
N-523   Remove Tool Offset 3.
 M2
N511    X-200    EOB
 M99
N521    X200     EOB
 M99
N512    X-500    EOB
 M99
N522    X500    EOB
 M99
N513    X-100    Y-300    EOB
 M99
N523    X100    Y300    EOB
 M99
 M30
```

The first block placed the system in Incremental Mode, Linear Interpolation and set the feedrate number. The second block caused the N511 subroutine to be executed, which inserted tool offset 1. That part of the program using tool offset 1 was executed. Then N-521 caused subroutine N521 to be executed, removing tool offset 1. Then N512 was executed, inserting tool offset 2; etc. Note that the last block removed tool offset 3; in this case it was not necessary since the first block sent the drives to the home position (G7), however it is a necessary procedure to remove all tool offsets at the end of the program when the first block does not contain a G7 command.

The second method for using tool offsets does not require going back to the reference position (removing the previous tool offset), but it does require that the operator keep track of the tool position. The previous program could have been written as follows.

```
N1   G1   G91   G7   F500   EOB
N-511  Inserts Tool Offset 1.
  Program using Tool Offset 1.
N-515  Inserts Tool Offset 2.
  Program using Tool Offset 2.
N-516  Inserts Tool Offset 3.
  Program using Tool Offset 3.
N-523  Remove Tool Offset 3.
  M2
N511  X-200  EOB
  M99
N515  X-300  EOB
  M99
N516  X400  Y-300  EOB
  M99
N523  X100  Y300   EOB
  M99
  M30
```

This program is the same as the previous one, up to the end of that part of the program using Tool Offset 1. Then Tool Offset N515 combines N521 and N512. This saves the time and memory required to go back to the reference position and then to the new offset position. Similarly N516 combines the previous N522 and N513 tool offsets.

### 4.11.2 Inserting, Examining and Changing Tool Offsets

The Tool Offsets must be added following M2 of the program, the same as any other subroutines. To examine or change the offsets, enter the Tool Offset number in MDI into the display, then switch to EDIT and press SHIFT SRCH. The number of bytes of program memory preceding that offset will be displayed. Then press STEP and the tool offset sequence number will be displayed, followed by the offset commands when the subroutine is stepped through.

To change a Tool Offset command, search for and display the offset, then press ERASE. Then switch to MDI and enter the desired offset.

## 5.0 MAINTENANCE AND ADJUSTMENTS

There is no maintenance required on the electronics or stepping motors. DC motor/tachs should be checked for brush wear at 5000 hr. intervals; refer to the maintenance section of the Serial Load Encoded DC Motor Drive Instruction Manual.

## 5.1 ENCODED DC MOTOR DRIVE ADJUSTMENTS

Refer to the Serial Load Instruction Manual

## 6.0 TROUBLESHOOTING

Troubleshooting will be aimed at determining whether the Drive or computer Control is malfunctioning. The Control stores and processes the command signals to the Drive. The Drive accepts serial commands from the Control, positions the motor, then sends a signal to the Control indicating all commands have been completed; the Control then processes the remaining data. The Control will not continue in the normal positioning mode (G24) unless the Drive returns the CZ signal. The Control will not continue after a HOME command unless the Drive has arrived at the home position, and sends back an AT HOME indication.

## 6.1 GENERAL

The Drive consists of a Serial Load Card, the Servo Controller, and the Motor/Tach/Encoder Assembly.

Troubleshooting hints for the Drive can be found in the Serial Load Instruction Manual. The Control consists of the Microcomputer Card, the CLI Card and the I/O Card. The Control requires a ± 12V and +5V DC supply for operation. The Drive also uses a +5VDC supply, and ± 12 VDC supplies. A Check of all power supplies should precede other tests for a malfunctioning system.

The I/O Card communicates with all Parallel Interface options (Magtape, Papertape Reader), and the Front Panel Card with the keyboard and switches. The Microcomputer Card interfaces to the Serial Interface options (TTY, RS 232) and all the remaining cards.

## 6.2 SYSTEM STOPS AFTER HOME COMMAND

If the system did not contain an Aerotech stage, the problem may be caused by the encoder marker position being off with respect to the CCW $\overline{\text{Limit}}$. The system operates by driving into the CCW Limit, reversing rotation (motor turns CW), and resetting the drive at the first marker pulse encountered. The marker should be located 1/4 to 3/4 revolution away from the limit.

The Drive can be checked by observing the led HOME indication, this LED must light when the Home reference position is reached, for the Control to continue.

6.3 SYSTEM STOPS AFTER INDEX COMMAND (G24 MODE)

Check that the Drive is getting to position (PSN LED ON). If it is not, the Drive Balance or Analog Detent may be misadjusted. If it is ON, replace the CLI Card.

Check also that the MFO Switch is not set to zero and that an F command other than zero is in the program.

## 6.4 ERROR MESSAGES, N-9

The display will show error messages preceded by N-900 XX. The messages are:

00 Check AOK

01 LIMIT. One of the drives has entered a CW or CCW limit.

02 thru 04 Reserved

05 MEMORY OVERFLOW. The RAM Memory capacity has been exceeded.

06 Reserved.

07 INVALID DIRECTIVE. The command is not allowed.

08 RAM ERROR. The ram memory location, whose decimal address is shown after a 10 second display of N-90008, is stuck at a ONE or a ZERO.

09 PROM ERROR. The Prom Checksum has failed indicating a failure in the executive program.

10 PARITY ERROR, Paper Tape A parity error has occured on the paper tape reader input.

11 IC ERROR, Magtape. An incomplete character was read from the magnetic tape.

12 CHECKSUM ERROR, Magtape. The checksum generated when reading the tape does not agree with the tape checksum.

23 IC & CHECKSUM. Both errors 11 and 12 occured.

14 SEARCH ERROR. Cannot find data being search for.

Additional error messages are found in the option instruction manuals.