December 23, 1996

Command Differences VP9000 verses NF90

VP9000	NF90
IA*m*M0 Index to Absolute Zero Position	I*m*M0 Index to Absolute Zero Position
I*m*M0 Index to Positive Limit	I*m*M0 Index to Absolute Zero Position
LM0 Set Loop-to-marker at current location	L-0 Set Loop-to-marker at current location
Looping Range is 2 to 65,535	Looping Range is 2 to 255
Acceleration units are x 1000 steps/sec. ^{2}	Acceleration units are x 2000 steps/sec. ^{2}
Maximum Index distance= 16,777,215 Steps	Maximum Index distance= 1,048,575 Steps

The Following Commands are for the VP9000 Only

IA*m***M**x	Set **A**bsolute **I**ndex distance, m=motor# (1,2,3,4), x= ±1 to ±16,777,215 steps
IA*m***M-0**	Zero motor position for motor# *m*, *m*= 1,2,3,4
I*m***M0**	**I**ndex motor until positive limit is encountered, *m*=motor# (1,2,3,4)
I*m***M-0**	**I**ndex motor until negative limit is encountered, *m*=motor# (1,2,3,4)
LM-0	Resets the **L**oop-to-marker to the beginning of the current program
LA*x*	**L**oop **A**lways from beginning or Loop-to-marker *x*-1 times (*x*=2 to 65,535)
LA-*x*	**L**oop **A**lways from beginning or Loop-to-marker *x*-1 times, alternating direction of motor 1
PA*x*	**P**ause *x* tenths of a second **A**ltering output 1 high for duration of the pause, tenths of a millisecond when *x* is negative
PM*x*	Select **P**rogra**m** number *x*, *x*= 0 to 30
PM-*x*	Select and clear all commands from **P**rogra**m** number *x*, *x*= 0 to 30
PM	Request the number of the current **P**rogra**m**
J*x*	**J**ump to the beginning of program number *x*, *x*= 0 to 30
JM*x*	**J**ump to the beginning of program number *x* and come back for **M**ore after program *x* ends, *x*= 0 to 30
U10	Wait for a low on input 2
U11	Wait for a low on input 2, holding user output 2 high while waiting
U12	Wait for a keyboard key to jump to a program or continue: ← key to jump to program #21, → key to jump to program #22, ↓ key to jump to program #23, ↑ key to jump to program #24, 0 or SEL key to proceed in current program
U13	Wait for a button on Remote Control to jump to a program or continue: 1- key to jump to program #25, 1+ key to jump to program #26, 2- key to jump to program #27, 2+ key to jump to program #28, ▭ key to proceed in current program
U14	**U**ser output 2 low (reset state)
U15	**U**ser output 2 high
U16	Wait for ← key to be pressed on front panel keyboard
U17	Wait for → key to be pressed on front panel keyboard
U22	Wait for a keyboard key to jump to a program and come back, or continue: ← key to jump to program #21 and come back, → key to jump to program #22 and come back, ↓ key to jump to program #23 and come back, ↑ key to jump to program #24 and come back, 0 or SEL key to proceed in current program
U23	Wait for a button on Remote Control to jump to a program and come back, or continue: 1- key to jump to program #25 and come back, 1+ key to jump to program #26 and come back, 2- key to jump to program #27 and come back, 2+ key to jump to program #28 and come back, ▭ key to proceed in current program
U30	Wait for a high to low transition on user input 1
U31	Wait for a high to low transition on user input 1, holding user output 1 high while waiting
U32	Wait for 1- button to be pressed on Remote Control
U33	Wait for 1+ button to be pressed on Remote Control
U40	Wait for a low to high transition on input 2
U41	Wait for a low to high transition on input 2, holding user output 2 high while waiting
U50	Wait for a high and low on user input 1 with debouncing for a mechanical push-button switch
U51	Wait for a high and low on user input 1 with debouncing for a mechanical push-button switch, holding user output 1 high while waiting
U60	Wait for a low and high on user input 2 with debouncing for a mechanical push-button switch
U61	Wait for a low and high on user input 2 with debouncing for a mechanical push-button switch, holding user output 2 high while waiting
U70	Zero counts in user programmable counter
U71	Display user programmable counts "on"
U72	Display user programmable counts "off"
U73	Increment user programmable counter by one
U90	Wait for a high and low on the Run connection I/O,4 with debouncing for a mechanical push-button switch
G	Enable On-Line mode with echo off **G**rouping a <cr> with "**^**", "**<**", "**>**", "**=**", "**:**", "**W**", "**O**" responses; Also **G**o after waiting or holding
T	Send position of motor 4 to host
M	Request **M**emory available for currently selected program
#	Request the **number** of the currently selected motor
%	Requests encoder/motor status from last index, VP9000 sends: **=** when motor counts and encoder counts are **equal**, **<** when encoder counts are **less** than motor counts (motor stall), **>** when encoder counts are **greater** than motor counts (motor overshoot)
*	Request the position when the last motor that was interrupted by a "**D**" command or user input 4