

MDI Program Samples

Motion Sample

CODE	DESCRIPTION
PG 100	'Enter program mode at address 100
LB M1	'Label program M1
P=0	'Set position counter to 0
MR 200000	'Set motion mode to relative, move relative 200000
H	'Hold until motion is complete
PR P	'Print position to terminal
E	'End program
PG	'Exit program mode

IF THEN Sample

CODE	DESCRIPTION
PG 100	'Enter program mode at address 100
LB SU	'Label using SU to start program upon power up
S1=0,1	'Sets I/O 1 as a general purpose input, active high
LB M1	'Label to begin motion profile 1
MR 200000	'Move relative 200000 steps
H	'Hold until motion is complete
BR M1, I1=1	'If input is true then branch to label M1
LB M2	'Label to begin motion profile 2
MR -100000	'Move relative 100000 steps
H	'Hold until motion is complete
BR M1, I1=1	'If input is true then branch to label M1
BR M2	'Unconditional brance to label M2
E	'End program
PG	'Exit program mode

Use of I/O Sample

CODE	DESCRIPTION
PG 100	'Enter program mode at address 100
LB SU	'Label using SU to start program upon power up
A=1000000	'Sets acceleration to 1000000 steps/second square
S1=0,1	'Sets I/O 1 as a general purpose input, active high
S5=9	'Sets I/O 5 as a 0-5V analog input
LB WT	'Label WT for setting up wait loop
BR M1,I1=1	'If input is active then branch to label M1
SL 0	'Sets slew speed to 0
BR WT,I1=0	'If input is inactive then branch to label WT
LB M1	'Label for motion profile 1
SL 15*100	'Slew the axis at the rate of 0-1024 times 100
BR WT	'Unconditional branch to wait loop to scan input
E	'End program
PG	'Exit program mode

MDI Program Samples

Integer Math Only (No Decimals)

CODE	DESCRIPTION
VA V1=10	'declare global variable V1
VA V2=3	'declare global variable V2
VA V3	'declare global variable V3
PG 1	'enter program mode at address 1
LB aa	'label aa
V3=V1+V2	'add values in V1 and V2 put result into V3
PR "V3= " V3	'print the value in V3 to the terminal
H 1000	'hold program execution for 1000 milliseconds
V3=V1*V2	'multiply values in V1 and V2 put the result into V3
PR "V3= " V3	'print the value of V3 to the terminal
E	'end program
PG	'exit program mode

'program action
'V3=V1+V2 produces an answer of 13
'V3=V1*V2 produces an answer of 30
'V3=V1/V2 produces an answer of 3
'try typing different values of V1 and V2 and "ex aa"
'V1=35<enter>
'V2=4<enter>
'EX aa<enter>
'results
'V3=V1+V2=39
'V3=V1*V2=140
'V3=V1/V2=8

Limit Switch Test Program

S1=2,0	'set input 1 as positive limit, low true
S2=3,0	'set input 2 as negative limit, low true
PG 1	'enter program mode at address 1
LB aa	'label aa
VM=51200	'set max velocity to 51200 steps/sec
MR 102400	'move positive 102400 steps
	'close switch at input 1 to
	'stop motion in the positive direction
	'and cause an error 83
	'NOTE: PROGRAM CONTINUES TO EXECUTE
	'leave switch open and program runs normal
H	'hold program execution until motion complete
H 250	'hold program execution for 250 millisecond
PR Er	'print error number to terminal window
MR -102400	
	'close switch at input 2 to
	'stop motion in the negative direction
	'and cause an error 84
	'NOTE: PROGRAM CONTINUES TO EXECUTE
	'leave switch open and program runs normal
H	'hold program execution until motion complete
H 250	'hold program execution for 250 millisecond
PR Er	'print error number to terminal window
BR aa	'branch to label aa
E	'end program
PG	'exit program mode
	'hit "Esc" key to stop program

MDI Program Samples

Speed Control Via Analog Input

'This program allows the user to perform speed control using the analog input on the MDI.

CODE	DESCRIPTION
S5=9	'sets analog input to accept 0-5V
S1=0,1	'sets I/O point 1 to general purpose output
A=2000000	'acceleration set to 2000000 microsteps/sec ²
D=2000000	'deceleration set to 2000000 microsteps/sec ²
R4=20	'deadband value
PG 1	'initiate program at address 1
LB SU	'startup label, program executes on power up
LB ZZ	'label called ZZ
R1 = I5	'register 1 set to analog input value
CL ab	'computes new velocity by calling subroutine at label ab
SL R3	'slew at the value of Register 3
H 10	'wait 10 milliseconds
BR ZZ	'branch to the label called ZZ
E	'end of ZZ routine
LB ab	'routine to determine direction and remove deadband
R1 = R1-508	'offset from joystick center
R2 = 1	'positive direction
BR a1, R1>=0	'get direction of R1
R2 = -1	'negative direction
R1 = R1 * R2	'get absolute value of R1
LB a1	
R1 = R1 - R4	'remove deadband
R1 = R1 * 1000	'scale multiplier
R3 = R1 * R2	'resultant R3 value for slew speed
RT	'return to command below call in ZZ routine
E	'end
PG	'exit program space