7i76 MPG Connection LinuxCNC Lathe Custom.hal file

7i76 Set to Mode 2 creates Encoder 0 on input 16-17 (A and B) and another on inputs 18-19 (A and B) which is not used here since there is only one encoder

This is the style of encoder, ad said 4 axis but actually 6 showed up

colour	signal	7i76 Pin	7i76 desc
red	Vcc	TB3 pin 20	5V+
black	GND	TB3 pin 23	GND
green	A	TB5 Pin 1	Input 16
white	B	TB5 Pin 2	Input 17
purple	A\	No Connection	
purple\black	B\	No Connection	
yellow	X	TB5 Pin 5	Input 20
yellow\black	Y	No Connection	
brown	Z	TB5 Pin 7	Input 22
brown\black	4	No Connection	
pink	5	No Connection	
pink\black	6	No Connection	
gray	X1	TB5 Pin 9	Input 24*
gray\black	X10	TB5 Pin 10	Input 25
orange	X100	TB6 Pin 11	Input 26
orange\black	COM	TB1 Pin 4	Field Power 24+
green\black	LED +	TB3 pin 20	5V+
white\black	LED -	TB3 pin 23	GND
blue	stop-c	TB1 Pin 4	Field Power 24+
blue\black	stop-cn	TB6 Pin 1	Input 0

*Input 24 is the default so I'm not sure this is even required to be connected but doesn't seem to hurt anything

NOTE: The GND from TB3 (pin 24) needed to be connected to the GND on TB1 (pin 7, even though the MESA paperwork says pin 6 and 7 are NC they are both connected to pin 8 GRD, Vin, Field CMN) without this the encoder signal is not consistent at all and didn't actually work in one direction at all

CUSTOM.HAL file contents

```
# Include your custom HAL commands here
# This file will not be overwritten when you run PNCconf again

# Jog Pendant XYZ456 X1 X10 X100 With NC E-stop and Enabled button

loadrt mux4 count=1
loadrt and2 count=1
addf mux4.0 servo-thread
addf and2.0 servo-thread


# For position mode (the default), set to 0
# In position mode the axis will move exactly jog-scale
# units for each count, regardless of how long that might take,
setp    axis.0.jog-vel-mode 0   # X AXIS
setp    axis.2.jog-vel-mode 0   # Z AXIS


# This sets the increments that will be used based on the input to the mux4
setp mux4.0.in0 0.0001
setp mux4.0.in1 0.001
setp mux4.0.in2 0.01


# The inputs to the mux4 component, default is not required so Input 24 isn't referenced
net scale1 mux4.0.sel0 <= hm2_5i25.0.7i76.0.0.input-25
net scale2 mux4.0.sel1 <= hm2_5i25.0.7i76.0.0.input-26


# The output from the mux4 is sent to each axis jog scale
net mpg-scale <= mux4.0.out
net mpg-scale => axis.0.jog-scale
net mpg-scale => axis.2.jog-scale


# The Axis select inputs
net axis-select-x   <=  hm2_5i25.0.7i76.0.0.input-20
net axis-select-z   <=  hm2_5i25.0.7i76.0.0.input-22


# --- ESTOP-EXT ---
net estop_jog     <=  hm2_5i25.0.7i76.0.0.input-00


#  MAPPING MPG ENCODER signals to jog counters
# using one encoder for both axis (X,Z)
net jog-counter <= hm2_5i25.0.7i76.0.0.enc0.count
```

```
#  MAPPING INPUT signals to linux cnc signals / gui
# Since there is only one encoder it is connected to both axis. Only the one that is selected
# actually moves
net jog-counter => axis.0.jog-counts
net jog-counter => axis.2.jog-counts


# Send input signal to linux cnc action (enable axis selection)
net axis-select-x => axis.0.jog-enable
net axis-select-z => axis.2.jog-enable


# ESTOP button trigger GUI control
net estop_prog <= iocontrol.0.user-enable-out
net estop_prog => and2.0.in0
net estop_jog => and2.0.in1
net and2out <= and2.0.out => iocontrol.0.emc-enable-in


#net temp <= passthrough.in
```