Traverse Winder Application

Abstract

Many applications require a set of motions to be executed successively, one immediately following the other. These motions can be either time based or slaved to the motion of another axis. The REPEAT command can be used with both the MOVE FOR and GEAR FOR commands to achieve a continuous sequence of motions. A continuous motion Traverse Winder application is used to demonstrate the use of the REPEAT GEAR command.

Description

A Traverse Winder system is used to evenly wrap material around a core with a width greater than that of the material being wound (examples: fishing line, spooled wire). A system of this type consists of two axes, the Winder and the Traverse.

The Winder axis wraps the material around the core, and the Traverse axis guides the material back and forth along the core’s length. In order for the material to be evenly wrapped around the core a fixed relationship must exist between the motions of the Winder and Traverse axes. It is also important that the Traverse axis change its direction of motion at the end of its travel at points in the Winder’s rotation that are offset from each other.

For our example the both axes will be servo controlled motors, however, the Winder axis could be a pacer encoder.

Implementation

Assume that the user units have been configured such that the Winder position units are degrees, and the Traverse position units are inches.

The following code is a table of operator configurable parameters which define the operation of the system:

```
OPERATOR.CONFIG:
  CYCLE OFFSET  =120  'offset per cycle in master degrees
  TRAVERSE ACCEL =45  'master degrees for acceleration
  TRAVERSE DIST! =1.000  'Traverse travel in inches
  WINDER REVS  =1  'numbers of Winder revs per Traverse
  CYCLES  =10  'cycle = forward index & reverse index
RETURN
```

Each cycle of the Traverse axis consists of two passes, forward and reverse, between 0” 1.000”. At the end of each Traverse axis cycle the Winder axis is 120 degrees out of phase from the end of the previous cycle.

The following code is the calculations required for the Traverse axis motion. NOTE: The position User Units are scaled up by 1000 for the Traverse axis, and 10 for the Winder axis, for better resolution.
CALCULATIONS:
INDEX& = TRAVERSE.DIST! *1000
' index distance scaled for user units
WINDER.DIST& = (WINDER.REVS * 360 + CYCLE.OFFSET / 2) * 10
'total Winder distance for Traverse motion
ACCEL.DIST& = TRAVERSE.ACCEL * 10
' Winder distance for Traverse acceleration
RETURN

The following code is the program that is executed to operate the system:

MAIN:

WINDER = 1
TRAVERSE = 2

MP.CONFIG ' configure controller parameters
OPERATOR.CONFIG ' configure system
CALCULATIONS ' calculate motion parameters

AXIS.SET@ = AXIS.LIST@
AFAULT@ = 0 : FAULT@ = 0 : WAIT 300 : MODE@ = 5 : WAIT 500
AXIS.SET@ = TRAVERSE ' command Traverse axis motions

REPEAT GEAR FOR INDEX& IN WINDER.DIST&, ACCEL.DIST&
REPEAT GEAR FOR -INDEX& IN WINDER.DIST&, ACCEL.DIST&
' command Winder axis motion

MOVE WINDER AT 60 IN 250

This part of the program keeps track of where the axis is in its motion so that the queue is interrupted at the desired point in the index sequence.

CYCLE.COUNT = 0 ' count the number of cycles completed

WHILE CYCLE.COUNT < CYCLES
  WAIT UNTIL POS.ACT@ > INDEX& / 2
  WAIT UNTIL POS.ACT@ < INDEX& / 2
  CYCLE.COUNT = CYCLE.COUNT + 1
WEND

In order to interrupt the motion queue a GEAR or MOVE for 0 distance must be commanded. This will stop the motion sequence after the motion in progress.

GEAR FOR 0 IN 10 ' interrupt the DSP motion queue
WAIT UNTIL DSP.DONE@
HALT WINDER IN 250
RETURN
END

Performance Considerations

The motions in the queue are initiated one immediately after the other, without missing a DSP tick.
Additional Information

For additional information on the MotionBASIC REPEAT command refer to the MotionBASIC Hypertext Manual.