

Doc No.: AN-217

Version: 1.0

Date: 08 September 2008

Subject: TABLE_POINTER command

APPLICATION NOTE

1. TABLE_POINTER

Type: Axis Parameter(Read Only)

Syntax: value=TABLE_POINTER

Where value is returned of type X.Y where X is the current TABLE location and Y represents the interpolated distance between the start and end location of the current TABLE location.

Description:

The ability to adjust a CAM based profiles from within the Trio BASIC program adds more flexibility to Trio's Motion Coordinators. Using the TABLE_POINTER command it is possible to determine which TABLE memory location is currently being used by the CAM allowing the user to load new CAM data into previously processed TABLE location ready for the next CAM cycle. This is ideal for allowing a technician to finely tune a complex process, or changing recipes on the fly whilst running. TABLE_POINTER returns the current table location that the CAM function is using. The returned number contains the table location and divides up the interpolated distance between the current and next TABLE location to indicate exact location.

Example:

In this example a CAM profile is loaded into TABLE location 1000 and is setup on axis 0 and is linked to a master axis 1. A copy of the CAM table is added at location 100. The Analogue input is then read and the CAM TABLE value is updated when the table pointer is on the next value.

```
'CAM Pointer demo
' store the live table points
TABLE(1000,0,0.8808,6.5485,19.5501,39.001,60.999,80.4499,93.4515)
TABLE(1008,99.1192,100)
' Store another copy of original points
TABLE(100,0,0.8808,6.5485,19.5501,39.001,60.999,80.4499,93.4515)
TABLE(108,99.1192,100)
```

```
' Initialise axes
BASE (0)
WDOG=ON
SERVO=ON

' Set up CAM
CAMBOX(1000,1009,10,100,1, 4, 0)

'Start Master axis
BASE (1)
SERVO=ON
SPEED=10
FORWARD

'Read Analog input and scale CAM based on input
pointer=0
WHILE 1
'Read Analog Input (Answer 0-10)
scale=AIN(32)*0.01
'Detects change in table pointer
IF INT(TABLE_POINTER)<>pointer THEN
  pointer=INT(TABLE_POINTER)
  'First value so update last value
  IF pointer=1000 THEN
    TABLE(1008,(TABLE(108)*scale))
  'Second Value, so must update First & Last but 1 value
  ELSEIF pointer=1001 THEN
    TABLE(1000,(TABLE(100)*scale))
    TABLE(1009,(TABLE(109)*scale))
  'Update previous value
  ELSE
    TABLE(pointer-1, (TABLE(pointer-901)*scale))
  ENDF
ENDIF
WEND
STOP
```