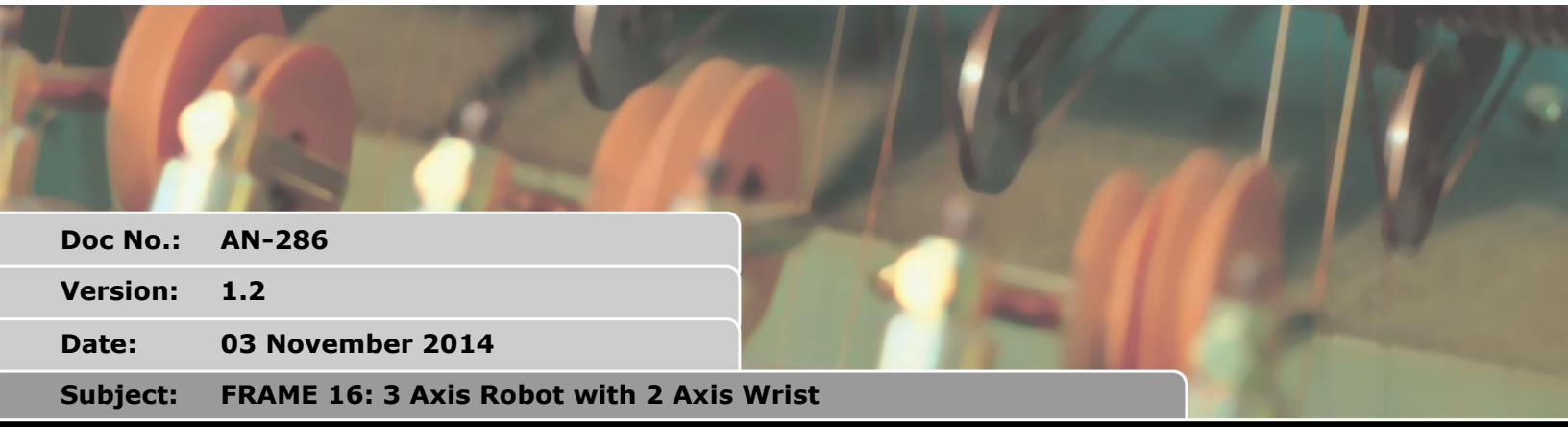
Trio Motion Technology Ltd.
Shannon Way, Tewkesbury,
Gloucestershire. GL20 8ND
United Kingdom
Tel: +44 (0)1684 292 333
Fax: +44 (0)1684 297 929

1000 Gamma Drive
Suite 206
Pittsburgh, PA 15238
United States of America
Tel: +1 412 968 9744
Fax: +1 412 968 9746

B1602 Tomson Centre
188 Zhang Yang Rd.,
Pudong New Area, Shanghai,
Postal code: 200122
CHINA
Tel/Fax: +86 21 587 97659



Doc No.: **AN-286**

Version: **1.2**

Date: **03 November 2014**

Subject: **FRAME 16: 3 Axis Robot with 2 Axis Wrist**

APPLICATION NOTE

1. Revision history

1.0	First draft	1st October 2012
1.1	Changes as detailed in section 2 below.	29th October 2012
1.2	Minor corrections to TABLE and axes order.	3rd November 2014

2. Changes:

1, Positive direction of B axis is changed. (2.0200 or newer)

NOTE: this update will require any motion on the B axis to be modified. The change was made to ensure compatibility of FRAME = 16 and USER_FRAME and TOOL_OFFSET

2, Axes order changed from X Y Z C B TO X Y Z B C. (2.0200 or newer)

NOTE: this update till require any motion on the B and C axes to be checked. The change was made to ensure compatibility of FRAME = 16 and USER_FRAME and TOOL_OFFSET

3, When FRAME = 16 then the B and C axes are programmed in Nanoradians. (2.0200 or newer)

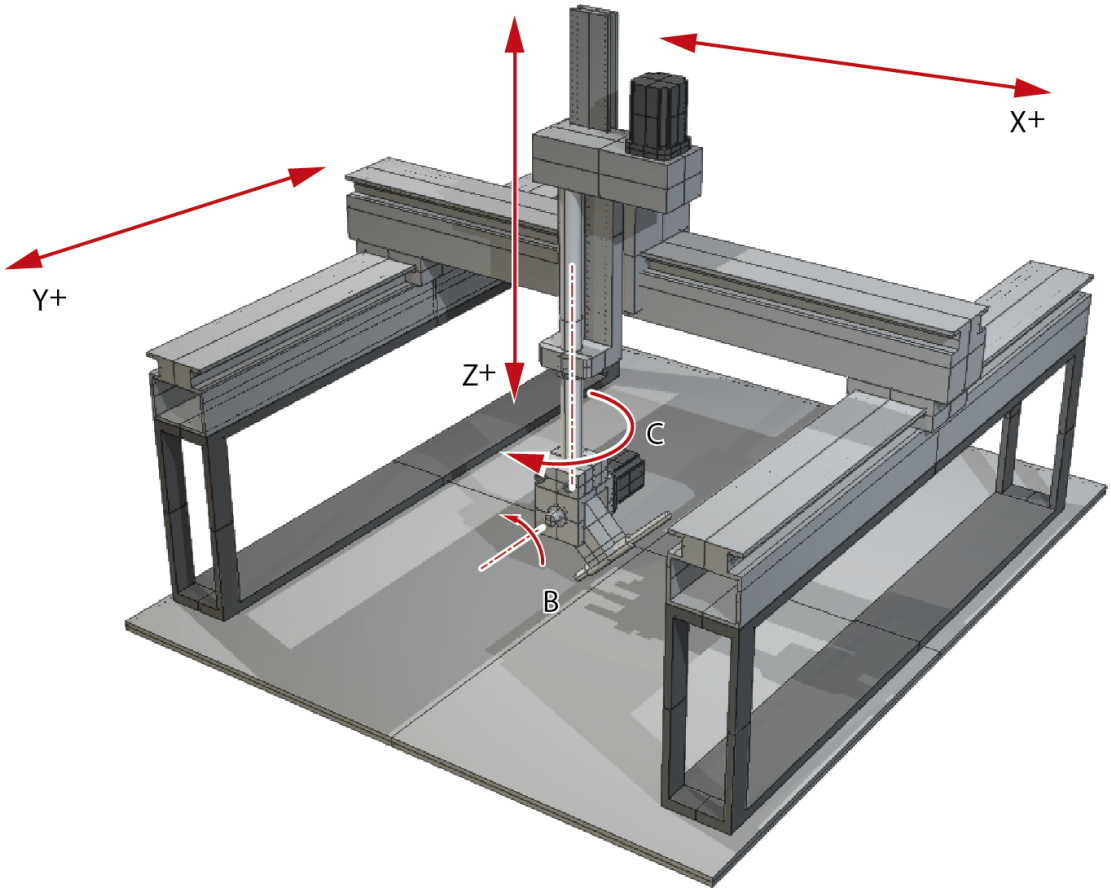
NOTE: this change will require UNITS on axis B and C to be changed. The change was made to ensure compatibility of FRAME = 16 and USER_FRAME and TOOL_OFFSET

3. Introduction

The FRAME 16 transformation allows an XYZ Robot with 2 axis wrist to be easily programmed. The transformation function provides compensation in XYZ when the 2 wrist axes are rotated. The FRAME 16 transformation is available for MC4XX controllers (including MC464, EURO408 and MC405) with the Robotic Function FEC (feature enable code) fitted.

4. Diagram

The following picture shows the type of machine:

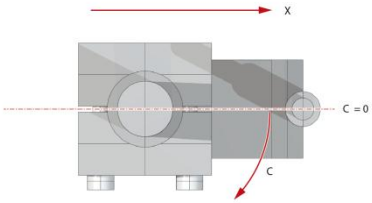


Once the frame is enabled DPOS on the X, Y and Z axis are measured in axis counts. The wrist axis is set to use Nanoradians. You can of course set UNITS for all axis to any suitable scale.

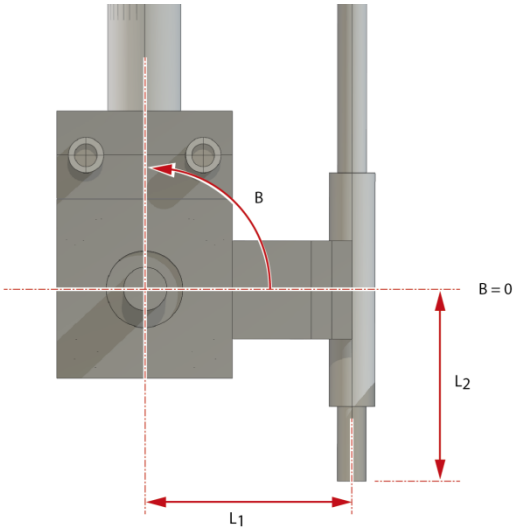
5. Datuming

Both wrist axes MUST be datumed to the correct zero position for the FRAME 16 transformation to operate. The zero position of the XYZ axes is not used by the transformation.

- The zero position on the C axis (rotation about Z) is when the offset arm is in line with the X axis. The diagram below is drawn from above looking down on to the X-Y plane.



- The zero position on the B axis(rotation about Y) is when the offset arm is the “straight down” position shown in the diagram.



The direction of motion on all 5 axes MUST match the diagram for the FRAME 16 transformation to operate. If an axis direction of motion is inverted it can be reversed either:

- Using the facility of the servo/stepper driver to invert the motion direction
- On stepper axes using STEP_RATIO function inside the Motion Coordinator
- On servo axes using ENCODER_RATIO / DAC_SCALE functions inside the Motion Coordinator

6. Parameterization

7 parameters needed to be set into a sequence of TABLE values for use by the transformation mathematics. The base TABLE position is set using FRAME_GROUP.

Offset		
0	Wrist joint to control point X offset (mm)	L1
1	Wrist joint to control point Z offset (mm)	L2
2	Wrist B axis encoder edges / radian	
3	Wrist C axis encoder edges / radian	
4	X axis encoder edges / mm	
5	Y axis encoder edges / mm	
6	Z axis encoder edges / mm	

6.1. Example

```
' Wrist offsets: 60mm in X and 90 mm in Z
' XYZ pulses/mm 1600,1600,2560
' B and C axes pulses radian = 3200 * 16 / (2 * PI)

TABLE(100,60,90,3200 * 8 / PI, 3200 * 8 / PI,1600,1600,2560)
```

```
' Set FRAME_GROUP zero using axes 0,1,2,3,4

FRAME_GROUP(0,100,0,1,2,3,4)

FRAME=16

… program moves in XYZBC with tool angle compensation

FRAME=0

… program axes
```

7. FRAME_GROUP

7.1. Type:

System Command

7.2. Syntax:

FRAME_GROUP(group, [table_offset, [axis0, axis1 …axisn]])

7.3. Description:

FRAME_GROUP is used to define the group of axes and the table offset which are used in a FRAME or USER_FRAME transformation. There are 8 groups available meaning that you can run a maximum of 8 FRAMEs on the controller.

Note:

FRAME_GROUP requires the kinematic runtime FEC

Warning:

Although 8 FRAME's can be initialised on a controller it may not be possible to process all 8 at a given SERVO_PERIOD. The number that can be run depends on many factors including, which FRAME is selected, drive connection method, if USER_FRAME and TOOL_OFFSET are enabled and additional factory communications.

The number of axes in the group must match the number of axes used by the FRAME. The axes must also be ascending order though they do not have to be contiguous. If a group is deleted FRAME and USER_FRAME are set to 0 for those axes.

To maintain backward compatibility if the FRAME command is used on an axis that is not in a group, or no groups are configured then a default group is created using the lowest axes and table_offset=0. In this situation if FRAME_GROUP(0) is already configured it is overwritten.

7.4. Parameters:

group:	The group number, 0-7. When used as the only parameter FRAME_GROUP prints the FRAME_GROUP, the active USER_FRAME and TOOL_OFFSET information to the currently selected output channel (default channel 0)
table_offset:	-1 = Delete group data 0+ = The start position in the table to store the FRAME configuration.
axis0:	The first axis in the group
axis1:	The second axis in the group
axisn:	The last axis in the group

7.5. Example:

Configure a FRAME_GROUP for axes 1,2 and 5 using table offset 100.

```
'Initialise the FRAME_GROUP
FRAME_GROUP(0,100, 1,2,5)

'Configure the axes, FRAME table data and home the robot
 GOSUB configure_frame

'PRINT the FRAME_GROUP information to the command line
 FRAME_GROUP(0)
```

8. Appendix

This is an example program that can be run on 5 virtual axes in the MC400 simulator. It can also be run on a real machine with some changes to the table data.

```
' set up frame 16 and move some axes.

  FRAME AXIS(1)=0
  FOR a=1 TO 5
    BASE(a)
    SPEED=100000
    ACCEL=SPEED*10
    DECEL=SPEED*10
    SERVO=ON
    DEFPOS(0)
    UNITS=1
  NEXT a
  WDOG=ON
  WA(10)

' Example:
' Wrist offsets: 60mm in X and 90 mm in Z
' XYZ pulses/mm 1600,1600,2560
' B axis pulses radian = 3200 * 16 / (2 * PI)
' C axis pulses radian = 4000 * 16 / (2 * PI)

  TABLE(100,60,90,3200 * 8 / PI, 4000 * 8 / PI,1600,1600,2560)

' Set FRAME_GROUP zero using axes 1,2,3,4,5
  FRAME_GROUP(0,100,1,2,3,4,5)

  BASE(1)
  UNITS AXIS(1)=TABLE(104)
  UNITS AXIS(2)=TABLE(105)
  UNITS AXIS(3)=TABLE(106)
  UNITS AXIS(4)=FRAME_SCALE AXIS(4)
  UNITS AXIS(5)=FRAME_SCALE AXIS(5)

  ' move to starting position
  MOVEABS(-60,0,-90)
  WAIT IDLE

  FRAME=16

  ' program moves IN xyzbc with tool angle compensation
  BASE(1,2,3)
  'TRIGGER
```

```
SPEED=100
ACCEL=SPEED*10
DECEL=SPEED*10
MOVECIRC(0,0,50,0,0)
WAIT IDLE

TRIGGER
BASE(5) ' change to base 4 to test B axis
SPEED=PI/4 ' take 8 seconds to move round one turn
ACCEL=SPEED*10
DECEL=SPEED*10
MOVE(-PI/2)
WAIT IDLE

FRAME=0
' program axes
```