A P P L I C A T I O N   N O T E

Doc No.:	**TN30-28**
Date:	**December 7, 2007**
Version:	**1.0**
Subject:	**CANopen DS402 control of two axes Elmo Bassoon demo**

Overview:

This document describes how the Trio Motion Coordinator with P293 CAN daughter board is initialized to control two Elmo Bassoon servo drives, using the CANopen DS402 protocol. This technique can be used with any other servo drive that supports the DS402 standard. In this mode the traditional +/- 10VDC analog velocity command and encoder feedback are replaced with a digital CAN network. The DS402 axes can be fully commanded as a traditional Trio axes. Each P293 can control up to 4 axes. The TrioBASIC program to initialize this is included at the end of this document. It is provided for example purposes and no guarantee is made as to its suitability for any particular application.
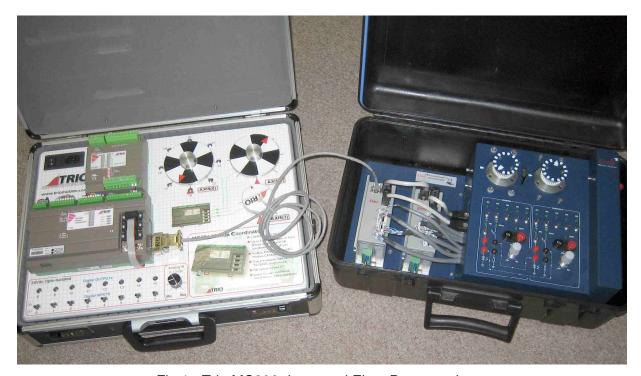

Fig.1 - Trio MC206 demo and Elmo Bassoon demo

Setup:

1. The Bassoon drives are configured as normal using the Elmo Composer software; this would include any servo loop tuning. The drive is set to operate in "Single Loop Position Control" mode (UM=5). The CAN-ID is set to 1 (PP[13]) for the first drive and 2 for the second drive, and the baudrate is set to 1MHz (PP[14]).

2. Make the CAN bus connection between the Trio MC and the Bassoon drives.

3. Load the "ELMO_DS402.BAS" program file into the Trio MC.

CANopen Operation:

The Trio MC CAN buffers are initialized to address a particular object in the drive. The MC firmware expects certain buffers to be used for a particular purpose. The object numbers are defined by the CANopen protocol. There are two types of communication used SDO and PDO. Here SDO is used by TrioBASIC to configure the drive. The PDO is used by the MC firmware to handle the cyclic communication of servo command and feedback.

Program Listing:

```
'=============================================================================
' Project:  Initialize CANopen DS402 control for two Elmo drives.
' Filename: ELMO_DS402.BAS
' Platform: (1)MC206X v1.6629 with P293 CAN daughterboard
'           (2)Bassoon 2.02.07.00 10May2006
'-----------------------------------------------------------------------------
' Revisions:
'   07 Dec 2007 - Initial release.
'-----------------------------------------------------------------------------
' Notes:
' Drives need to have CANID 1 and 2, baudrate 1MHz.
' MC206 AXIS(5) and (6) used to control drive.
' Drive is enabled/disabled via CAN SDO commands.
' Open MP2 Terminal #0 for debug messages.
' See TrioBASIC Help for CAN() command syntax.
'=============================================================================

cid1=1 : ax1=5 'can-id and axis number for drive #1
cid2=2 : ax2=6 'can-id and axis number for drive #2
slt=0 'P293 slot location

'--For smooth motor operation must use SERVO_PERIOD=2000.
IF SERVO_PERIOD <> 2000 THEN PRINT "SERVO_PERIOD=2000 REQUIRED" : STOP

WDOG=OFF
SERVO AXIS(5)=OFF
SERVO AXIS(6)=OFF
DRIVE_ENABLE AXIS(5)=OFF
DRIVE_ENABLE AXIS(6)=OFF

PRINT "Init CANopen"

CAN(slt,2,0) 'set baudrate to 1M
CAN(slt,5,15,$80,0,1) 'SYNC message in buffer 15
CAN(slt,5,14,$0,2,1) 'NMT message in buffer 14
CAN(slt,5,4,$380+cid1,8,0) 'drive 1 Tx PDO3 position feedback
CAN(slt,5,5,$380+cid2,8,0) 'drive 2 Tx PDO3 position feedback
CAN(slt,5,8,$400+cid1,4,1) 'drive 1 Rx PDO3 velocity command
CAN(slt,5,9,$400+cid2,4,1) 'drive 2 Rx PDO3 velocity command

'--Set all drives to pre-operational mode for mapping.
CAN(slt,7,14,128,0)
```

```
'--Setup drive #1.
drv=cid1 : ax=ax1
GOSUB setup_drive

'--Setup drive #2.
drv=cid2 : ax=ax2
GOSUB setup_drive

'RUN "STARTUP" 'sets axis tuning parameters

WDOG=ON

'--Enable drive #1.
drv=cid1 : ax=ax1
GOSUB enable_drive

'--Enable drive #2.
drv=cid2 : ax=ax2
GOSUB enable_drive

PRINT "Complete"
STOP

'-----------------------------------------------------------------------------
setup_drive:
  PRINT "Setup drive ";drv[0];

  CAN(slt,5,1,$580+drv,8,0) 'Tx SDO
  CAN(slt,5,13,$600+drv,8,1) 'Rx SDO

  '--Set drive speed for profiled velocity mode.
  IF CAN(slt,9,13,1,8,$6060,0,$0003,$0000)=0 THEN PRINT "$6060" : STOP

  '--Set PDO3 Rx off.
  IF CAN(slt,9,13,1,8,$1602,0,$0000,$0000)=0 THEN PRINT "RPDO OFF" : STOP

  '--Map PDO3 Rx to velocity command.
  IF CAN(slt,9,13,1,32,$1602,1,$0020,$60ff)=0 THEN PRINT "RPDO MAP" : STOP

  '--Set PDO3 Rx for SYNC controlled.
  IF CAN(slt,9,13,1,8,$1402,2,$0001,$0000)=0 THEN PRINT "RPDO SYNC" : STOP

  '--Set PDO3 Rx on.
  IF CAN(slt,9,13,1,8,$1602,0,$0001,$0000)=0 THEN PRINT "RPDO ON" : STOP

  '--Set PDO3 Tx off.
  IF CAN(slt,9,13,1,8,$1a02,0,$0000,$0000)=0 THEN PRINT "TPDO OFF" : STOP

  '--Map PDO3 Tx to position feedback.
  IF CAN(slt,9,13,1,32,$1a02,1,$0020,$6063)=0 THEN PRINT "TPDO MAP" : STOP

  '--Set PDO3 Tx for SYNC controlled.
  IF CAN(slt,9,13,1,8,$1802,2,$0001,$0000)=0 THEN PRINT "TPDO SYNC" : STOP

  '--Set PDO3 Tx on.
  IF CAN(slt,9,13,1,8,$1a02,0,$0001,$0000)=0 THEN PRINT "TPDO ON" : STOP

  '--Set operational mode in drive.
  CAN(slt,7,14,1,drv)

  ATYPE AXIS(ax)=19 'set axis type for CANopen speed
  AXIS_ADDRESS AXIS(ax)=drv 'map trio axis to drive address
  OUTLIMIT AXIS(ax)=5000000 'allow large dac_out values
  DRIVE_ENABLE AXIS(ax)=ON 'fires up sync from firmware ds402
```

```
   PRINT " done"
RETURN

'------------------------------------------------------------------------------
enable_drive:
  CAN(slt,5,1,$580+drv,8,0) 'Tx SDO
  CAN(slt,5,13,$600+drv,8,1) 'Rx SDO
  IF CAN(slt,9 ,13,1,16,$6040,0,$0000,$0000)=FALSE THEN STOP 'disable
  IF CAN(slt,9 ,13,1,16,$6040,0,$0006,$0000)=FALSE THEN STOP '
  IF CAN(slt,9 ,13,1,16,$6040,0,$0007,$0000)=FALSE THEN STOP 'enable voltage
  IF CAN(slt,9 ,13,1,16,$6040,0,$000f,$0000)=FALSE THEN STOP 'enable operation
  SERVO AXIS(ax)=ON
RETURN

'------------------------------------------------------------------------------
disable_drive:
  CAN(slt,5,1,$580+drv,8,0) 'Tx SDO
  CAN(slt,5,13,$600+drv,8,1) 'Rx SDO
  IF CAN(slt,9 ,13,1,16,$6040,0,$0000,$0000)=FALSE THEN STOP 'disable
  SERVO AXIS(ax)=OFF
RETURN
'==============================================================================
```