BALDOR MOTION PRODUCTS

Mint™ Multi-Tasking MT Application Note

AN00108-002 - Virtual Axes

Related Applications or Terminology

- ***Simulation without drives and motors.***
- ***Cascading motion***
- ***Adding axes motion together***
- ***Line encoder simulation***
- ***Virtual Master***

Supported Controllers

*NextMove*PCI	☑
*NextMove*BX^{II}	☑
*NextMove*ST	☑
*NextMove*ES	☑
*NextMove*ESB	☑
*NextMove*e100	☑
MintDrive^{II}	☑
Flex+Drive^{II}	☑

Relevant Keywords

CONFIG

AUXENCODERMODE

AUXENCODERSPEED

Overview - What is a Virtual Axis?

A virtual axis does not physically exist, it is imaginary! The motion controller will produce no demand output, nor will it require any physical feedback (position or velocity). The instantaneous position and velocity of a virtual axis is internally generated and is equal to the demand values produced by the motion profiler.

Axes Simulation

There are many uses for these virtual axes, for instance it allows aspects of the software to be developed and tested without the machine mechanics or drives and motors through simulation. Applications can be tested on a controller before completion of the machine itself.

Line Encoder Simulation or 'Virtual Master'

When commissioning or proving the operation of a system which is designed to follow a master axis or encoder, it is often useful to be able to simulate the master to test the functionality of the slave axes without the need to run the entire machine. In this case, a virtual axis can be used to replace or simulate the master axis for a slave or group of slave axes.

This has the added advantage that that the virtual axis will be "ideal" in that it will not suffer wiring problems, noise, mechanical or electrical problems. This can be used to prove the operation of slave axes or diagnose potential problems with the real encoder, or master axis.

It is for the very "ideal" characteristics of a virtual axis that many multi-axis synchronizing applications use the virtual master concept in normal operation. Where traditional line shafts have been replaced or a master encoder was once used, a virtual axis is used instead, to which all the slave axes synchronize.

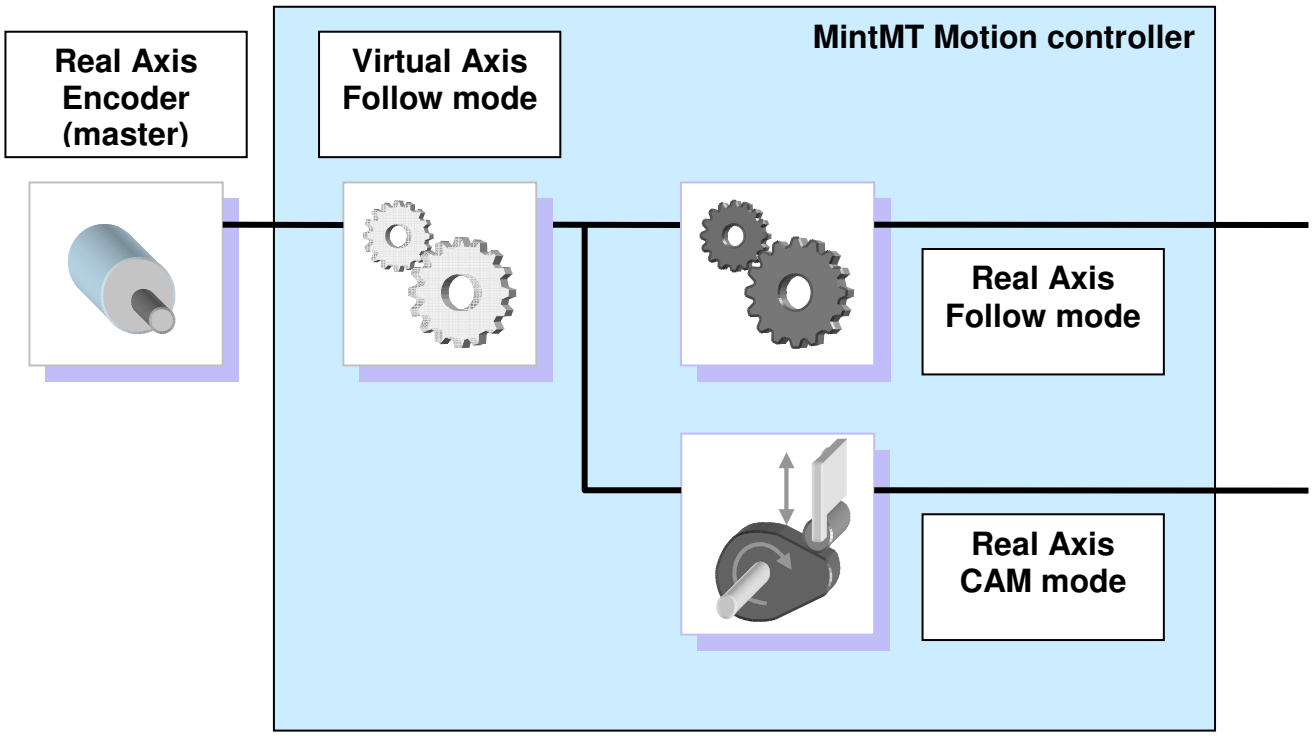
In the original scheme, where a real axis was followed, the slaves would have lagged the master. In the virtual master scheme, the original master also becomes a slave and is subject to the same time lag as the other slaves, hence eliminating lag between real axes.

'Cascading Motion' or 'Adding motion together'

Virtual axes provide an interesting mechanism for adding motion types together. This is achieved by configuring an axis to follow a virtual axis or vice versa. A motion sequence is then initiated on the master axis, which the slave will follow according to the type of process motion programmed. Mint provides keywords to allow either normal geared motion (FOLLOW), cam profiles (CAM) or flying shears (FLY) to be setup very quickly and easily.

For example, a real axis may be configured to execute a CAM profile following a virtual axis. The virtual axis can then execute any form of motion command itself such as JOG, MOVER, MOVEA, FLY, CAM, SPLINE etc and the slave will follow this motion in the relationship defined by the CAM table. If the virtual axis were executing a FOLLOW then registration corrections using OFFSET on the virtual axis would produce a corresponding registration adjustment on the real axis performing a CAM.

More than one real axis can be grouped to follow the same virtual axis as shown below. This allows groups of axes to be controlled synchronously, perhaps when the real master is not operating, for machine start-up purposes at a product changeover or during commissioning.



How to configure a virtual axis

NextMove products running MintMT allow virtual axes to be set-up in exactly the same way as normal axes, simply by configuring the axis or axes as virtual, as shown below.

```
'Example – Virtual axes with Mint MT NextMove products
CONFIG([0,1,2]) = _cfServo,_cfServo,_cfVirtual    'Axes 0 and 1 are real, Axis 2 is virtual
RESETALL                                          'Clear errors and enable all axes
MOVER(2) = 1000:GO(2)                             'Move the virtual axis forward 1000 units
```

Note: If using NextMove e100; the CONFIG keyword is no longer supported and the virtual axis is instead configured via the System Configuration wizard. Also CANCEL([axes list]), PAUSE IDLE([axes list]), DRIVEENABLE([axes list]) = 1; must be used to replace RESETALL

*Flex+*Drive ^{II} and MintDrive ^{II} allow the master encoder input to be 'disabled' so that a virtual auxiliary encoder signal can be used as a 'Virtual Master' instead.

This is achieved by setting bit 3 of the AUXENCODERMODE keyword and clearing bit 4.

Bit 0 of this keyword is used to set the direction of the auxiliary encoder hence a combination of these two bits can be used to configure a virtual master running in either direction.

The AUXENCODERSPEED keyword sets the speed of the virtual master (in User Units per second) scaled according to the value of AUXENCODERSCALE.

```
'Example – Virtual axes with MintDrive II
AUXENCODERMODE(0) = 8          'Auxiliary encoder is configured as virtual encoder
AUXENCODERSCALE(0) = 4096      'Virtual master units are revs/sec (4096 counts/rev)
AUXENCODERSPEED(0) = 10        'Virtual master speed = 10 revs/sec
RESET(0)                       'Clear errors and enable real axis
FOLLOW(0) = 2                  'Real axis geared to virtual master at 2:1 ratio
```