Introduction
Step and direction configuration is a simple control format which allows motion to be controlled from two inputs on the drive, one used to represent a requirement for a step change in the axis position and one used to represent the direction of travel. In essence the step and direction outputs from the motion controller / PLC are connected to one of the available encoder input channels on the drive and the drive is configured to follow this encoder, but the drive decodes the encoder count in a “step and direction mode” rather than the usual A/B quadrature decoding method.

Both MicroFlex e190 and MotiFlex e180 support the following encoder channels that can be used for step and direction input:

- Encoder channel 1 (24Vdc single ended encoder formed from fast digital inputs 1 and 2)
- Encoder channel 2 (RS422/5V differential encoder input)

Because this form of control is usually associated with low cost systems it is unusual for the drive’s simulated encoder output to be used by the motion controller to close the position loop, instead the motion controller usually treats the axis as an “open loop” stepper axis (relying on the drive itself to ensure the motor position matches the demand set by the incoming step/direction signals). This is not always the case though, sometimes the controller may want to perform “closed loop control”, in which case the drive’s simulated encoder output will be used to feedback actual position (when using MicroFlex e190 this would mean that a digital encoder or resolver must be used on the motor to allow use of encoder channel 2 via the Universal Encoder Input for the step and direction inputs).

Configuration of step and direction mode just requires a few simple parameters to be configured via Mint Workbench. As a result there is no need to run a Mint program and hence no need for the Mint memory module option for these drives.

NOTE: Neither drive supports 5V single ended step and direction inputs. Should your motion controller / PLC provide 5V TTL single ended step and direction signals it will be necessary to either include an intermediate signal conversion unit from a third party or you will need to use encoder channel 2 and bias the A- and B- encoder input lines to 2.5Vdc using a resistor network and the drive’s 5V encoder supply output – please contact your local ABB support office for further information if required.

Pre-requisites
You will need to have the following to work through this application note:

- Mint Workbench build 5854 or later (see new.abb.com/motion for latest downloads and support information)
- A MicroFlex e190 or MotiFlex e180 drive with build 5868.7.0 or later firmware, tuned and configured for position control as per the AN00250 Drive Tuning application note
- Appropriate hardware connected to the drive to provide suitable step and direction signals. If using encoder channel 2 for example then an ABB NextMove e100 or NextMove ESB-2 with differential stepper outputs or an ABB PLC with an FM562 PTO module would be suitable. If using encoder channel 1 then connection of two switches to digital inputs 1 and 2 will be sufficient for testing the basic operation
It is assumed the reader has a basic working knowledge of Mint Workbench and that the drive has already been tuned for the application. Throughout this document we will assume the use of a MicroFlex e190, but the process is identical for a MotiFlex e180 drive.

**Installation requirements**

Because this type of system is usually operated “open loop” the positional accuracy achieved is very dependent on the number of steps detected by the drive being accurate. As a result it is critical that all recommended installation practices relating to EMC performance are followed exactly…

- Drives should be mounted onto non-painted metal backplates
- An earth connection from the main panel earth star point should be made on the drive
- Motor power cable shields must be p-clipped to earth close to the drive
- Control signals must be routed away from power cables
- Differential step and direction signals must be wired using twisted pair cable with an overall shield
- Single ended step and direction signals must be wired as a twisted pair with the return signal and with an overall shield
- Functional earth connections on the device providing the signals must be installed
- Typically the 0Vdc of the control power supply should be linked to earth

Please refer to the relevant drive’s installation manual for full details.

It is highly recommended that RS422 / 5V differential line driver step and direction signals are used if at all possible (via encoder channel 2) for the greatest noise immunity and therefore the highest probability of reliable and accurate operation.

When using encoder channel 1 (the 24Vdc encoder channel formed by digital inputs 1 and 2) the controller providing the step and direction signals may use current sourcing / open emitter outputs (providing 24Vdc to DIN1+ and DIN2+, with DIN1- and DIN2- connected to 0Vdc) or it may use current sinking / open collector outputs (pulling DIN1- and DIN2- down to 0Vdc with DIN1+ and DIN2+ connected to 24Vdc).

In both cases it is still vital to ensure twisted pairs are used for these signals (twist the signal for DIN1+ with the signal for DIN1- and twist the signal for DIN2+ with the signal for DIN2-).

In the case of a PLC/controller using open collector outputs it is also necessary to install pull-up resistors in the circuit as shown in the diagram below to achieve a bandwidth of 250 kHz:
Resistor Rp | Bandwidth
---|---
None | Low
470 or 1k ohms | 250 kHz

It is also important to ensure these resistors have the correct (minimum) power rating as shown below:

<table>
<thead>
<tr>
<th>Resistor Rp</th>
<th>Power rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>470 ohms</td>
<td>1.5 W</td>
</tr>
<tr>
<td>1k ohms</td>
<td>1 W</td>
</tr>
</tbody>
</table>

Setting the configuration

As we mentioned previously, this application note assumes the drive has already been commissioned/tuned and is ready to operate in position control mode at power up. The quickest and easiest way to configure the drive to operate in step and direction mode is via the Mint Workbench command window. This is accessed via the ‘Edit & Debug’ area of Mint Workbench. For the rest of this document we will assume all drive parameters are set via the command window, but it should be noted that it is also possible to use the Parameter viewer within Mint Workbench to make the necessary configuration settings too should the user prefer a more “graphical” way to configure the drive. Remember to use Tools>Store Drive Parameters to save any changes in configuration.

Configuring step and direction inputs

The step and direction inputs are processed via encoder channels 1 and 2 on the drive. For firmware versions earlier than 5868, encoder channels 1 and 2 are automatically enabled. However, for firmware version 5868 onwards these encoders have been disabled by default (ENCODERTYPE for these channels is set to 22, to represent “None”). To allow these encoder channels (and hence the step and direction inputs) to work you must set ENCODERTYPE(1) and/or ENCODERTYPE(2) to 0...

\[
\text{e.g. ENCODERTYPE(1) = 0 (this would enable the encoder input formed by digital inputs 1 and 2)} \\
\text{ENCODERTYPE(2) = 0 (this would enable encoder channel 2 on connector X7)}
\]

As described earlier, the step and direction signals will either be connected to encoder channel 1 (if using 24Vdc single ended signals) or Encoder channel 2 (if using RS422/5V differential signals). In order for the A channel of these encoder inputs to be decoded as a ‘Step’ signal and the B channel of these encoder inputs to be decoded as a ‘Direction’ signal (instead of using a quadrature decoding scheme as would usually be the case with an encoder input) it is necessary to set bit 2 of ENCODERMODE for the relevant channel.
e.g. ENCODERMODE(1) = 4  (this would set the fast digital inputs to operate as step and direction inputs)
ENCODERMODE(2) = 4  (this would set encoder channel 2 to operate as step and direction inputs)

Note that the Mint constant _emSTEP_AND_DIR_INPUTS is pre-defined (with a value of 4) should you wish to enter this instead…
ENCODERMODE(1) = _emSTEP_AND_DIR_INPUTS

Although it is possible to change the direction of rotation for a step input by setting or clearing bit 0 of ENCODERMODE it is recommended that the MOTORDIRECTION parameter is used to configure the direction of rotation to suit the incoming step and direction signals.
e.g. MOTORDIRECTION(0) = 0 or MOTORDIRECTION(0) = 1

**Configuring the master reference**
The drive needs to be configured to ‘follow’ the encoder input channel we are using for the step and direction signals. This is achieved using the MASTERSOURCE and MASTERCHANNEL parameters.

Examples:
To configure the drive to use Encoder channel 1 (the 24Vdc digital inputs) as the encoder input for following we would enter…
MASTERSOURCE(0) = 1
MASTERCHANNEL(0) = 1

To configure the drive to use Encoder channel 2 (the RS422/5V differential encoder) as the encoder input for following we would enter…
MASTERSOURCE(0) = 1
MASTERCHANNEL(0) = 2

Note that the Mint constant _msENCODER is pre-defined (with a value of 1) should you wish to enter this instead…
MASTERSOURCE(0) = _msENCODER

**Configuring the follow ratio**
As we’ve already described, when operating in step and direction mode the drive is simply ‘following’ a defined encoder input. We have already shown how we used ENCODERMODE to ensure the drive decodes the relevant encoder input using a step and direction scheme instead of the usual quadrature encoder decoding. At this point we now need to configure the required ratio between the number of ‘steps’ (user units) the motor will move for each ‘step’ on the defined encoder input. By default the drive will count one ‘step’ for every rising edge of the corresponding step input (this can be modified using ENCODERMODE).

The follow ratio (defined using the FOLLOWRATIO command when using step and direction) is derived from…

Required axis movement (in scaled user units) for each change in relevant encoder input (in scaled encoder units).

This is best explained further via some examples, in these examples we’ll assume encoder 2 is being used for the step and direction inputs):

Example 1
POSSCALEFACTOR(0) = 1  (or SCALEFACTOR(0) = 1)
ENCODERSCALE(2) = 1

In this case the axis is scaled for a user unit of ‘encoder counts’ (i.e. there is 1 encoder count for each user unit) and encoder channel 2 is also scaled into counts (i.e. one count on encoder channel 2 is one user unit).
If we were to enter FOLLOWRATIO(0) = 10 at the command window then, when the drive is enabled and enters following mode using the step and direction signals, the axis would travel 10 user units (i.e. 10 counts) for every 1 user unit (i.e. every 1 count/step) on encoder channel 2.

**Example 2**
P OSSCALEFACTOR(0) = 131072 (or SCALEFACTOR(0) = 131072)
ENCODERSCALE(2) = 10000

In this case the axis is maybe using an ABB ESM servo motor with Smartabs encoder feedback (with resolution of 131072 counts per revolution) and so the axis has been scaled into ‘revs’ (i.e. there are 131072 counts per user unit). Encoder channel 2 has been scaled such that 1 ‘unit’ on this encoder channel corresponds to the encoder input changing by 10000 counts.

If we were to enter FOLLOWRATIO(0) = 1 at the command window then, when the drive is enabled and enters following mode using the step and direction signals, the axis would travel 1 user unit (i.e. 1 revolution) for every 1 user unit (i.e. every 10000 counts/steps) on encoder channel 2.

**Configuring how the drive is enabled**
In this type of application the motion controller or PLC (or maybe even some simple external logic) is usually used to provide the drive with a drive enable signal. We are not using a Mint program so there will be no code available to issue the DRIVEENABLE command, instead we use the following parameters to ensure the drive is automatically enabled/disabled from a defined digital input…

DRIVEENABLEINPUT(0) = x   (where x is the required digital input channel)
DRIVEENABLEINPUTMODE(0) = 0   (to ensure the drive doesn’t generate an error when the enable input is removed)
DRIVEENABLEMODE(0) = 2  (to set a hardware enable mode of operation)

**Automatically starting in step and direction mode**
At power up we need to ensure the drive automatically follows the defined encoder input when it enables. This is achieved by entering…

AUTOSTARTMODE(0) = 4

The Mint pre-defined constant _asFOLLOW (with a value of 4) can be used instead if required.

**Completing the set up**
Once all of the relevant settings have been made you must save the current configuration using the Tools>Store Drive Parameters menu option (or you can use the ‘Store drive parameters’ toolbar button instead – usually at the top right corner of the Mint Workbench environment). Once stored use the Tools>Reset Controller… menu option (or physically power cycle the drive) to reset it and ensure that the defined auto start mode works as expected when you operate the drive’s enable input (the seven segment display on the drive will indicate _FOLLOW to show it has entered a following mode of operation.

**Examples**
We can summarise the preceding sections into the following sequences of commands/parameters:

*Step and direction via encoder channel 1 (24Vdc)*
SCALEFACTOR(0) = x   (enter value for axis scale factor)
ENCODERTYPE(1) = _entROTARY_ENCODER
ENCODERSCALE(1) = x   (enter value for encoder scale factor)
ENCODERMODE(1) = _emSTEP_AND_DIR_INPUTS
MASTERSOURCE(0) = _msENCODER
MASTERCHANNEL(0) = 1
FOLLOWRATIO(0) = x   (enter calculated follow ratio)
AUTOSTARTMODE(0) = _asFOLLOW
DRIVEENABLEMODE(0) = 2
DRIVEENABLEINPUT(0) = x   (enter input number to be used as enable input)
DRIVEENABLEINPUTMODE(0) = _emIGNORE
MOTORDIRECTION(0) = x  (0 or 1 as required)
Tools>Store Drive Parameters (from Mint Workbench menu)
Power cycle the drive

Step and direction via encoder channel 2 (5Vdc/RS422)
SCALEFACTOR(0) = x   (enter value for axis scale factor)
ENCODERTYPE(2) = _entROTARY_ENCODER
ENCODERSCALE(2) = x   (enter value for encoder scale factor)
ENCODERMODE(2) = _emSTEP_AND_DIR_INPUTS
MASTERSOURCE(0) = _msENCODER
MASTERCHANNEL(0) = 2
FOLLOWRATIO(0) = x   (enter calculated follow ratio)
AUTOSTARTMODE(0) = _asFOLLOW
DRIVEENABLEMODE(0) = 2
DRIVEENABLEINPUT(0) = x   (enter input number to be used as enable input)
DRIVEENABLEINPUTMODE(0) = _emIGNORE
MOTORDIRECTION(0) = x   (0 or 1 as required)
Tools>Store Drive Parameters (from Mint Workbench menu)
Power cycle drive

Testing and monitoring operation
Again using the Mint Workbench ‘Edit and Debug’ window we can use this to monitor the axis position and the relevant encoder channel being used for the step and direction signals. To do this we can go to the ‘Spy’ window area to the right of the ‘Edit & Debug’ screen and select the ‘Monitor’ tab. In the top data area select ‘Position’ from the first drop down box and ‘Axis 0’ and in the second drop down box select ‘Encoder’ and then select the relevant channel according to whether you are using the 24V digital inputs (channel 1) or the RS422/5V differential encoder (channel 2).

As we operate the step input we should see a change in the encoder channel in the spy window, even if the drive is disabled. The encoder input always decodes the step and direction inputs regardless of the enabled status of the drive.

With the drive enabled and following the defined encoder channel we should be able to confirm that as the encoder input changes by a known amount the axis also moves by the amount defined by our POSSCALEFACTOR, ENCODERSCALEFACTOR and FOLLOWRATIO settings.

If something does not work as expected then double check the correct settings have been made. If you make any changes be sure to store drive parameters and power cycle before re-testing.

Contact Us
For more information please contact your local ABB representative or one of the following:

new.abb.com/motion
new.abb.com/drives
new.abb.com/drivespartners