

# Application note

## Moving to NextMove e100

AN00162

Rev A (EN)

Take advantage of the enhanced capabilities of the NextMove e100 motion controller. This handy product migration guide illustrates how it may be possible to update existing NextMove applications to suit the latest generation real-time Ethernet controller.



### Introduction

With the introduction of NextMove e100, a number of improvements and changes have been made in the firmware to allow for distributed real-time Ethernet systems and control. Whilst ABB always strives for backwards compatibility with previous generation products, sometimes this is not possible. NextMove e100 firmware contains a number of changes from previous generation controllers that enhance the product functionality, but at the same time means some changes to existing programs are necessary.

This application note highlights these differences to make moving from previous generation NextMove controllers, such as NextMove ESB, as easy as possible. Mint programs will inevitably require some degree of modification before use on NextMove e100.

### Hardware differences

The following table summarises the hardware features of boxed NextMove products compared with NextMove e100:

	BX <sup>1</sup>	ESB	ESB-2	e100
Maximum software axes	8	8	8	16 (+14 drive profiled)
Maximum local servo axes	4	3	4	3
Maximum stepper axes	0	4	4	4
Total encoder inputs	5	4 <sup>1</sup>	6 <sup>1</sup>	3
Analog outputs	4	4	4	4
Analog inputs	8	2	2	2
Digital inputs	16	20	20	20
Digital outputs	8	11	12	12
Fast inputs	4 dedicated	4 shared with digital inputs	4 shared with digital inputs	4 shared with digital inputs
Serial connections	RS232 and RS422	RS232 or RS422 (specified at order time)	RS232 or RS422 (specified at order time)	RS232 or RS422 (selector switch)
USB connection	no	yes	yes	yes
CAN	CANopen and BaldorCAN	CANopen or BaldorCAN (selected with firmware)	CANopen or BaldorCAN (selected with firmware)	CANopen
Mint program space	256Kb	1024Kb	1024Kb	1024Kb

<sup>1</sup> One encoder channel is a slow speed channel realised via three digital inputs.

Battery backed RAM	yes	no	no	no
Non-volatile storage	-	16Kb	16Kb	16Kb

### Understanding the components

Although not essential, some grounding in the firmware structure is useful.

Firmware is composed of several components. The two components of most interest are the 'Mint Motion Library' (MML) and the 'Mint Virtual Machine' (MVM).

The MML is the real-time portion of the firmware and defines all the available Mint motion keywords, such as Speed, Accel etc. The MVM describes the features of the Mint language itself, such as structures and keywords like Loop, While etc. The MVM is versioned to show the level of functionality supported. This is known as the Target Format. There have been a number of Target Formats released on NextMove: 8, 9, 10, 13 and 14. On NextMove e100, there have been formats 13 and 14. The term Mint v5.0 is taken to mean all firmware builds between 5000 and 5499. The term Mint v5.5 is taken to mean all firmware builds above 5500.

### Mint WorkBench compatibility modes

When a Mint program is compiled in Mint WorkBench, the compatibility mode affects a number of options that control the Mint compiler behavior. The compatibility mode is found on the Tools menu in Mint WorkBench. Select Options and then click on the Compiler Options tab.

Mode 5000 is for Target Format 10 and below (firmware before Build 5400). This will reduce the number of warnings produced and restrict the use of new language features introduced in later firmware.

Mode 5400 is for Target Format 13 (firmware after Build 5400). This allows use of new language features of Target Format 13 such as structures and the Time data type.

Mode 5500 is for Target Format 13 (firmware after Build 5400) and Target Format 14 (firmware after build 5615). As well as allowing the use of new language features, the semantic checking is increased and compiler warnings will be generated for the use of Mint syntax that has been deprecated.

The default compatibility mode for older NextMove controllers is 5000 or 5400 depending on firmware version. For NextMove e100, the default compatibility mode is 5500.

For further information, see the Mint Help file topic: **Porting from Mint v5 (MT) to Mint v5.5**.

### Axis syntax

The most common problem encountered when moving to NextMove e100 is axis syntax. The following examples show axis syntax that has now been deprecated.

```
SPEED[0] = 10
SPEED.0 = 10
SPEED[0,1] = 10;
Pause IDLE[0,1]
CONTOURPARAMETER[0]._ctpSTOP_ANGLE = 45
```

These will generate a 'Legacy Parameters ...' warning. The correct syntax for Nextmove e100 is shown below:

```
SPEED(0) = 10
SPEED(0) = 10
SPEED([0,1]) = 10;
Pause IDLE([0,1])
CONTOURPARAMETER(0,_ctpSTOP_ANGLE) = 45
```

Warnings do not stop the operation of a Mint program though they can indicate where unexpected behavior may result. Ideally, the Mint code should be modified, removing the warnings. However, if porting an existing Mint application to NextMove e100, this may not be practical. Changing the compatibility mode to 5400 will stop the warnings from being generated.

Another example is with keywords that have a variable number of axes. These will also generate a 'Legacy Parameters ...' warning.

```
MOVEA[0,1] = 10;  
VECTORA[0,1] = 10;  
GO[0,1]
```

The correct syntax for NextMove e100 is:

```
MOVEA([0,1]) = 10;  
VECTORA(0,1) = 10;  
GO(0,1)
```

Note the difference between the MOVEA and VECTORA syntax. Normally, the [] syntax is used as a shortcut to call the Mint keyword for each of the axes listed between the []. For example, `SPEED([0,1]) = 10, 20` means `SPEED(0) = 10` and `SPEED(1) = 20`.

Some keywords always require multiple axes to be specified and some keywords can take a variable number of arguments. These keywords cannot use the [] syntax. Deciding whether to embed the square brackets inside round brackets, or to simply change the square brackets to round ones can cause some confusion, but the rule is that if the function takes a variable number of arguments (indicated by a ... in the help file and the hover-over help) or requires multiple axes to be specified then the square brackets should be converted into round ones; otherwise put round brackets around the [] parameters present.

The following keywords do not use the [] syntax:

```
CANCEL  
CIRCLEA / CIRCLER  
DEFAULT  
GO  
HELIXA / HELIXR  
VECTORA / VECTORR  
STOP
```

### Optional parameters

A number of keywords have optional parameters where a default parameter is used when omitted from the Mint code. The default is specified with AXES, BANK, BUS and TERMINAL. This functionality has been deprecated in Mint v5.5 and will generate warnings.

```
SPEED = 10  
INPUTACTIVELEVEL = 0x1001  
BUSBAUD = 500  
Print myVariable
```

These will generate the following warnings: 'Default axes ...', 'Default bank ...', 'Default bus ...' and 'Default terminal ...'.

The correct syntax is to explicitly call the parameters:

```
SPEED(0) = 10  
INPUTACTIVELEVEL(0) = 0x1001  
BUSBAUD(_busCANOPEN) = 500  
Print #_TERM2, myVariable
```

If the Mint code cannot be modified then changing the compatibility mode to 5400 will stop the warnings from being generated.

## Axis Configuration

All axis configuration on NextMove e100 is performed using the System Configuration Wizard in Mint WorkBench. This produces a device configuration file (DCF) that is stored in the file system on the controller and executed at startup. It is not possible to change axis configurations in a Mint program. This affects the following keywords:

Mint v5.0 Keyword	Suggested Mint v5.5 Replacement	Comments
AXISCHANNEL	AXISDAC, AXISPDOUTPUT, AXISPOSENCODER, AXISVELENCODER	Read only, assigned in the System Configuration Wizard.
CHANNELTYPE	-	Not supported
CONFIG	CONFIG	Read only, assigned in the System Configuration Wizard.
LOOPTIME	CONTROLRATE	Read only, assigned in the System Configuration Wizard.
PROFLETIME	CONTROLRATE	Read only, assigned in the System Configuration Wizard.

## Reset

The RESET and RESETALL keywords are no longer supported in Mint v5.5. The following code replicates its functionality:

```
CANCEL(axis)
Pause IDLE(axis)
POS(axis) = 0
POSROLLOVER(axis) = 0
DRIVEENABLE(axis) = 1
```

## Cancel

In Mint v5.0, the CANCEL keyword would zero the size of the axis following error. This functionality is not DS402 compliant so does not occur in Mint v5.5. There are two methods to zero following error on an axis.

1. If the axis is idle then writing to position will clear following error. See the Idle keyword in the Help file for details on the conditions of being idle. For example:

```
POS(0) = 0
POS(2) = POS(2)
```

If the axis is not idle, change the IdleMode to allow the axis to become idle. Then write to position.

```
IDLEMODE(0) = _idmIGNORE_IDLEPOS + _idmIGNORE_IDLEVEL
POS(0) = POS(0)
IDLEMODE(0) = 0
```

2. Disabling the axis will clear following error.

```
DRIVEENABLE(0) = 0
```

## Auxiliary Encoders

There are no auxiliary encoders on NextMove e100. The following keywords are not supported:

```
AUXENCODER
AUXENCODERMODE
AUXENCODERPRESCALE
AUXENCODERSCALE
AUXENCODERVEL
AUXENCODERWRAP
AUXENCODERZLATCH
```

## Error Handling

Support for Ethernet Powerlink required a number of changes to error handling so Mint v5.5 has a new error handling scheme. See the Mint Help file topic '**Error categories (e100)**' for full details.

In Mint v5.0 firmware, errors are reported in a number of keywords where bit within some of these keywords represented an error. In v5.5 firmware, all errors have a unique error code and a set of new keywords provide a consistent and simplified interface to access those errors. The errors are read from a buffer called the Error List. Each time an error is read, it is removed from the list. Mint WorkBench also has a feature known as the Error Log. The Error Log is a historical record of errors, whereas the Error List is designed to be manipulated and handled in a Mint program.

When an error occurs, the ONERROR event is called. Errors from the list can either be read sequentially in the order they were added (ERRORREADNEXT) or individual errors can be checked for (ERRORREADCODE). All existing error handling functionality can be ported to the new scheme. Examples of error handlers can be seen in the Help file topic '**Error Event: ONERROR**'. The Mint Library included with Mint Workbench also includes some sample code that can be dragged into the program editor.

The following keywords are affected:

Mint v5.0 Keyword	Suggested Mint v5.5 Replacement	Comments
ASYNCEERRORPRESENT	ERRORPRESENT	Read with group _egAXIS_ERROR
AXISERROR	ERRORREADNEXT ERRORREADCODE	Read with group _egAXIS_ERROR Use individual error codes
AXISSTATUS	-	Not Supported
AXISWARNING	ERRORREADNEXT ERRORREADCODE	Read with group _egAXIS_ERROR Use individual error codes
AXISWARNINGDISABLE	ERRORCODEENABLE	Use individual error codes
ERL	ERRLINE	Primed with call to ERRORREADNEXT or ERRORREADCODE
ERR	ERRCODE	Primed with call to ERRORREADNEXT or ERRORREADCODE
ERRAXIS	ERRAXIS	Primed with call to ERRORREADNEXT or ERRORREADCODE
ERRORMASK	ERRORCODEENABLE	Use individual error codes
ERRSTR	ERRSTRING	Primed with call to ERRORREADNEXT or ERRORREADCODE
INITERROR	ERRORREADNEXT ERRORREADCODE	Read with group _egCONTROLLER_ERROR Use individual error codes
INITWARNING	ERRORREADNEXT ERRORREADCODE	Read with group _egCONTROLLER_WARNING Use individual error codes
MISCERROR	ERRORREADNEXT  ERRORREADCODE	Read with group _egCONTROLLER_ERROR or _egCONTROLLER_WARNING Use individual error codes
MISCERRORDISABLE	ERRORCODEENABLE	Use individual error codes

In Mint v5.0, the error registers ERR, ERL etc were automatically primed on entry to the ONERROR event handler. By default in Mint v5.5, the call to ERRORREADNEXT or ERRORREADCODE primes the registers ERRCODE, ERRLINE etc. Changing compatibility mode to 5400 or below will prime the error registers on entry to the handler.

## Fast Position Capture

To increase the flexibility and simplify fast position capture, this functionality has been re-written. This is explained in full in the Mint Help file topic '**Fast latching (e100)**'.

As there are no auxiliary encoders on NextMove e100, the following keywords are not supported:

[FASTAUXENABLE](#)  
[FASTAUXENCODER](#)  
[FASTAUXLATCH](#)  
[FASTAUXLATCHMODE](#)

## FASTAUXSELECT

The FASTxxx keywords are axis based. The LATCHxxx keywords in Mint v5.5 are channel based where any channel can be freely configured. The following keywords are affected:

Mint v5.0 Keyword	Suggested Mint v5.5 Replacement	Comments
FASTENABLE	LATCHENABLE	Set to 1 to enable
FASTENCODER	LATCHVALUE	
FASTIN event	LATCHx event	One event per channel
FASTIN0..x event	LATCHx event	
FASTLATCH	LATCH	
FASTLATCHDISTANCE	LATCHINHIBITVALUE LATCHINHIBITTIME	
FASTLATCHEDGE	LATCHTRIGGEREDGE	
FASTLATCHMODE	LATCHMODE	Mode _fLATCH_SMART has no direct replacement. Manually enable the latch on completion of the latch event.
FASTPOS	LATCHVALUE	
FASTSELECT	LATCHTRIGGERCHANNEL LATCHTRIGGERMODE	
FASTSOURCE	LATCHSOURCE LATCHSOURCECHANNEL	

In an Ethernet Powerlink system, if the data to be latched is not available on the NextMove e100, then the latching must be configured on the controlled node (i.e. the remote drive). In the System Configuration Wizard, the latch channel on the remote Ethernet Powerlink device must be mapped to a latch channel on the NextMove e100. The actual configuration of the latch on the drive can either be done directly on the drive (i.e. using Workbench to edit the drive's parameters) or via redirected Mint calls in the Mint program running on the NextMove e100 (see the "Redirection" help file topic for further details on how to configure and implement redirected Mint calls to remote axes).

## CAN Bus

There is no support for BaldorCAN nodes on NextMove e100. The following keywords are not supported:

KEYS  
 READKEY  
 REMOTEBAUD  
 REMOTEDEBOUNCE  
 REMOTEESTOP  
 REMOTEINPUTACTIVELEVEL  
 REMOTENODE  
 REMOTEOUTPUTACTIVELEVEL  
 REMOTEOUTPUTERROR  
 REMOTERESET

With CANopen, the Group functionality is not supported. Groups allow COMMS data to be sent to a number of nodes simultaneously with broadcast messages. This must be replaced with separate calls to REMOTECOMMS for each node. This affects the following keywords:

GROUP  
 GROUPCOMMS  
 GROUPMASTER  
 GROUPMASTERSTATUS  
 GROUPSTATUS

## Relay

NextMove e100 has a relay output but this is now controlled directly as a digital output channel. The RELAY keyword is not supported. Digital output channel 12 on NextMove e100 controls the relay.

For example:

RELAY(0) = 1 on NextMove e100 is OUTX(12) = 1

DRIVEENABLEOUTPUT(2) = \_RELAY0 on NextMove e100 is DRIVEENABLEOUTPUT(2) = 12

GLOBALERROROUTPUT = \_RELAY0 on NextMove e100 is GLOBALERROROUTPUT = 12

The relay is also affected by the OUTPUTACTIVELEVEL keyword.

## Homing

For local axes, there is no change in the homing on NextMove 100.

For Ethernet Powerlink axes, although homing is initiated by the NextMove e100, the Ethernet Powerlink device must have the home input wired to it and the homing parameters are also configured on the Ethernet Powerlink device (although again these can be set via redirection from the Mint program running on the NextMove e100).

## Other Keywords

A number of other keywords are also not supported or have modified functionality as listed below:

Mint v5.0 Keyword	Suggested Mint v5.5 Replacement	Comments
ACTIVERS485NODE	-	No RS485 master support
ADCERROR	-	Not supported. The functionality can be achieved with Mint code.
ADCERRORMODE	-	Not supported
ADCMAX	-	Not supported
ADCMIN	-	Not supported
ADCMONITOR	-	Not supported
AUXDAC	-	No auxiliary DAC outputs
BLEND	-	Not supported
BLENDDISTANCE	-	Not supported
BLENDMODE	-	Not supported
BOOST	-	No Boost outputs
CAPTUREEVENT	CAPTURETRIGGERxxx	Capture triggering allows a Capture to triggered automatically.
CAPTUREEVENTAXIS	CAPTURETRIGGERxxx	Capture triggering allows a Capture to triggered automatically.
CAPTUREEVENTDELAY	CAPTURETRIGGERxxx	Capture triggering allows a Capture to triggered automatically.
CAPTUREINTERVAL	CAPTUREPERIOD	The capture period is specified in microseconds.
COMMSRETRIES	-	No RS485 master support
COMPARELATCH	-	Use OUTX to check the state of the compare output.
COMPAREMODE	COMPAREENABLE COMPAREOUTPUT	
DACMONITORAXIS	-	Not supported. The functionality can be achieved with Mint code.
DACMONITORGAIN	-	Not supported. The functionality can be achieved with Mint code.
DACMONITORMODE	-	Not supported. The functionality can be achieved with Mint code.
DACRAMP	-	Not supported
DPRFLOAT	-	No Dual Port RAM interface.
DPRLONG	-	No Dual Port RAM interface.
LED	-	No LED display
LEDDISPLAY	-	No LED display
MAXSPEED	-	Not supported
NODE	BUSNODE	Each communication channel can have a unique node number
POSROLLOVERDEMAND	-	Not supported



PRECISIONINCREMENT	-	Not supported
STOPINPUTMODE	STOPMODE	Also see SUSPENDINPUT
TERMINALADDRESS	-	No BaldorCAN so not supported
TERMINALDEVICE	-	All channels are VT100
TERMINALPORT	-	Terminal channels cannot be reassigned
TORQUE	TORQUEREF	

### Contact us

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

[new.abb.com/motion](http://new.abb.com/motion)  
[new.abb.com/drives](http://new.abb.com/drives)  
[new.abb.com/drivespartners](http://new.abb.com/drivespartners)  
[new.abb.com/PLC](http://new.abb.com/PLC)

© Copyright 2014 ABB. All rights reserved.  
Specifications subject to change without notice.