

**Mint**™ Multi-Tasking  
**MT Application Note****AN00197-002 – Replacing BaldorCAN Keypads****Overview**

BaldorCAN is a proprietary fieldbus protocol that operates on CAN. The protocol is based on the CAN Application Layer for Industrial Applications messaging protocol (CAL), as specified by the CAN in Automation group (CiA). There is a large installed base of BaldorCAN peripherals such as keypads (KPD002-501, KPD002-502, KPD002-505) and I/O modules like the I/O Node 24 (ION004-501). These peripherals allowed the expansion of features on motion controllers such as Smartmove, NextMoveBX and NextMovePCI (amongst others). These BaldorCAN peripherals are now obsolete and are no longer manufactured.

As Baldor products have developed over the last 10 years the use of BaldorCAN has been phased out with a move towards the widely available CANopen protocol being favoured instead. During this period some products have supported both BaldorCAN and CANopen (either together or as an either/or option).

Baldor has for some time offered programmable keypads/HMIs suitable for use with CANopen but have had none that offer simple text display using only Mint code on board the motion controller. Now Baldor are able to offer the KPD202-501 operator panel which allows for RS232/485 serial or CANopen communication.

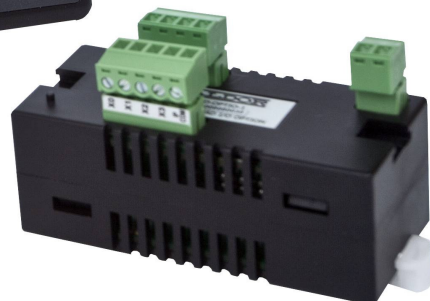


The new **KPD202-501** replaces the following obsolete BaldorCAN operator panels:

- KPD002-501 - 3 axes model, 27 keys
- KPD002-505 - 4 axes model, 41 keys

The new **KPD202-501** most closely resembles the KPD002-501 but has controls for 4 axes and 29 keys.

This device shown on the right is the **KPD-OPTIO-1** which adds 4 digital inputs and 4 digital outputs when fitted to the new Baldor keypad operating on the CANopen network.



This application note details the differences and changes required to convert a system from an existing BaldorCAN keypad to a new Baldor keypad. If you are designing a new system using this type of keypad please refer to the manual 'MN1960 – Baldor Keypad' for full details.

# Mint<sup>MT</sup> Multi-Tasking Application Note

**Important:** There is no replacement for a BaldorCAN keypad utilizing the BaldorCAN protocol so applications using a BaldorCAN keypad must be assessed to see whether all of the BaldorCAN peripherals can be replaced with CANopen based peripherals (this will only be possible where the motion controller supports CANopen of course) or alternatively switch to using a new Baldor keypad which can be used over serial (only possible where the motion controller supports serial). If the user needs to update all peripherals to CANopen then please refer to MN1960 and MN1962 for details on how to use the new keypad and modules with the CANopen protocol, 'AN00196 – Replacing BaldorCAN Peripherals' along with this application note also provide details on replacing BaldorCAN devices.

## Existing Products Supporting BaldorCAN

For legacy applications requiring replacement of existing BaldorCAN peripherals (due to obsolescence) there are some decisions to be made.

If the motion controller only supports BaldorCAN then it will not be possible to replace any BaldorCAN keypad that might be present (as there is no replacement for the BaldorCAN keypad that still supports the BaldorCAN protocol).

If the motion controller supports CANopen then the BaldorCAN keypad can be replaced with the new Baldor keypad and any existing I/O modules on BaldorCAN can be replaced by the latest I/O modules that support CANopen (refer to MN1962 for details). If you only require 4 digital inputs and 4 digital outputs then there is the option of using the KPD-OPTIO-1 which plugs in to the back of the keypad. Please refer to the manual and Baldor website for further information on the KPD-OPTIO-1.

If the motion controller supports serial communication then the new Baldor keypad can be used as a replacement in its serial configuration.

The table below summarizes where BaldorCAN replacement may be possible or not:

Product Family	BaldorCAN support?	CANopen Support?	Does CANopen port need powering?	CANopen keypad support possible?	Firmware version required	Serial support possible?
Smartmove	Yes (later models only)	No	-	No	Contact <a href="mailto:motionsupport.uk@baldor.com">motionsupport.uk@baldor.com</a> for details	
NextMoveBX	Yes	Yes	No	No	No firmware support	Yes
NextMovePC	Yes	No	-	No	-	No
NextMovePCI	Yes	Yes	No	<b>Yes</b>	5414	No
MintDrive	Yes (on Dual CAN variant only)	Yes	No	No	No firmware support	Yes
MintDrive <sup>II</sup>	Yes (on Dual CAN variant only)	Yes	No	<b>Yes</b>	5383	Yes
Flex+Drive <sup>II</sup>	No	Yes	No	No	No firmware support	Yes
NextMoveBX <sup>II</sup>	Yes	Yes	Yes	No	No firmware support	Yes

# Mint<sup>MT</sup> Multi-Tasking Application Note

Product Family	BaldorCAN support?	CANopen Support?	Does CANopen port need powering?	CANopen keypad support possible?	Firmware version required	Serial support possible?
NextMove ES, ST	Yes (firmware variant)	Yes (firmware variant)	Yes	Yes	5414	Yes
NextMove ESB, ESB-2	Yes (firmware variant)	Yes (firmware variant)	Yes	Yes	5414	Yes
NextMove PCI-2	Yes (firmware variant)	Yes (firmware variant)	Yes	Yes	5414	No
e100 products	No	Yes	Yes	Yes	5621	NextMove e100 only

## CANopen connection

The new Baldor keypad has two types of CANopen connection shown below:

### CANopen port - 9 pin D-type



Pin	Name	Description
1	-	(NC)
2	CAN-	CAN channel negative (CAN_L)
3	CAN GND	Ground/earth reference for CAN signals
4	-	(NC)
5	Shield	Shield connection
6	CAN GND	Ground/earth reference for CAN signals
7	CAN+	CAN channel positive (CAN_H)
8	-	(NC)
9	-	(NC)
<b>Details</b>	Opto-isolated CANopen communication port.	

# Mint<sup>MT</sup> Multi-Tasking Application Note

## CANopen port - RJ45



Pin	Name	Description
1	CAN+	CAN channel positive (CAN_H)
2	CAN-	CAN channel negative (CAN_L)
3	-	(NC)
4	CAN GND	Ground/earth reference for CAN signals
5	-	(NC)
6	-	(NC)
7	-	(NC)
8	-	(NC)
<b>Details</b>		Opto-isolated CANopen communication port.

**Note:** The D-type and RJ45 CANopen ports are internally connected so can be used to connect to the neighbouring devices (on the same CANopen bus) on each side of the keypad (i.e. 'daisy-chaining'). DIP switch 1 may be used to connect an internal 120 Ω termination resistor between the CAN+ and CAN- signals

When deciding which port on the new keypad to use it is worth considering whether you need to power the CANopen port of the Motion controller. The 'Existing Products' table shown previously listed whether or not the CANopen port on the motion controller requires powering.

This is something that needs considering now because the older BaldorCAN keypads fed the 24V supply applied on the keypad through into the BaldorCAN connection and then to the motion controller. The new CANopen keypad does not do this so requires that you supply power to the motion controller by another means. One way of achieving this is to use an **OPT-CNV002** DB9 Female to RJ45 converter.



This device converts from a female D-type to a RJ45 socket and has flying leads for a 24V supply. It also has a 120 Ω termination resistor fitted inside. If the device to which the adapter is connected is not at the end of the CAN network, the termination resistor must be removed. This can be achieved by opening the adapter's case and disconnecting or removing the resistor.

By using one of these converters and one of the RJ45 cables listed below any motion controller's CANopen port can be powered. For further details on connections see MN1960.

# Mint<sup>MT</sup> Multi-Tasking Application Note

**Pre-made RJ45 to RJ45 cables:**

Part Number	Length
CBL0025CM-CSS2	0.25m
CBL005CM-CSS2	0.50m
CBL010CM-CSS2	1.00m
CBL020CM-CSS2	2.00m
CBL050CM-CSS2	5.00m
CBL100CM-CSS2	10.00m
CBL200CM-CSS2	20.00m

## Serial port



Pin	RS232 name	RS485 name
1	(NC)	(NC)
2	RXD	RX- (input)
3	TXD	TX- (output)
4	(NC)	(NC)
5	0V / GND	0V / DGND
6	(NC)	(NC)
7	RTS	TX+ (output)
8	CTS	RX+ (input)
9	(NC)	(NC)
<b>Details</b>	Serial communication port supporting RS232 or RS485. Max baud rate: 115.2 Kbit/s	

The serial connector is a 9-pin female D-type connector. This port can be configured as either RS232 or RS485, using the serial mode switch on the keypad. When using RS485 mode, DIP switch 3 may be used to connect an internal 120 Ω termination resistor between the RX+ and RX- signals (for further info please refer to MN1960).

For details on mounting the Baldor keypad as well as further hardware info refer to MN1960 or MN1963 Baldor keypad quick start guide.

## Software Changes

When a new Baldor Keypad is used to replace an existing BaldorCAN keypad then some software changes will be required to the existing Mint application code. Please note that configuration code should be included in the startup block or a subroutine called from the startup block. Full details on the CANopen keywords supported by the new Baldor keypad are listed in manual MN1960.

### CANopen

#### Node ID, Node type and Baud rate

The default Node ID for both the new and old keypads is 14 so you will probably not need to make any changes for this. Refer to MN1960 if you need to modify the Node ID of the keypad. The Node type of the BaldorCAN keypad was `_ntKEYPAD = 9`. The new Baldor keypad has a node

# Mint<sup>MT</sup> Multi-Tasking Application Note

type of `_ntBALDOR_KEYPAD = 46` or `_ntBALDOR_KEYPAD_IO_EXPANSION = 47` when the optional IO (KPD-OPTIO-1) is fitted.

The default baud rate for a BaldorCAN keypad was 125kbit/s, the default baud rate for the new Baldor keypad is 500kbit/s. This may be defined using code within a Mint program or within the motion controller setup. If you would like to change the Baud rate of the keypad refer to MN1960.

## Terminal settings

The following changes have occurred relating to the TERMINAL settings:

The TERMINALPORT when using CANopen will change to `_tpCANOPEN` (or `_tpBUS1`) = 1.

The TERMINALDEVICE when using the new Baldor keypad will change to `_tdBALDORKEYPAD` = 4.

## Serial

### Baud rate

The default baud rate of the new Baldor keypad is 57600.

### Terminal settings

If using a non-e100 device you can choose a suitable terminal number for the application and use the following terminal settings. TERMINALPORT should be `_tpSERIAL1 = 0` and TERMINALDEVICE should be `_tdVT100 = 0`

If using an e100 device there is a pre-defined terminal object for VT100 devices on serial. By using `_TERM1` as the terminal you will interface with the Baldor keypad over serial. There are no settings to consider other than the baud rate as mentioned previously.

## CANopen and Serial

### Line feeds

After using a PRINT or LINE statement to display characters on the keypad display a line feed is automatically issued to position the cursor at the first position of the next line. If you do not wish this to happen you can use a `,` comma to suppress the line feed leaving the cursor at the end of the previous line. The example below will suppress the line feed leaving the cursor on line 2 (the comma at the end of the line of code suppresses the line feed):

Line 2,"Display example",

The way in which line feeds work on the new panel has changed. When you fill a line with 20 characters it automatically line feeds to the next line, it will then perform the usual line feed too. This can cause the cursor to be positioned 2 lines below where the text was just entered. Because of this it is now necessary to add a comma at the end of the third line of text if you are printing 20 characters as well as always on the fourth to stop the page from scrolling. For example the code below prints some text to the keypad display on lines 3 and 4. On line 3 we will print 20 characters (including spaces) and on line 4 only 14.

Line 3,"Welcome to Baldor UK",

Line 4,"Bristol Office",

Comma's are required on both line 3 and line 4 to stop the page (current 4 lines visible) being lost as they scroll off the screen.

Due to this same dissimilarity displaying 20 characters on line 2 without suppressing the line feed will cause the cursor to move to the beginning of line 4. If your next line of mint code is a PRINT that expects the cursor to be positioned at the beginning of line 3 like it did on the old keypad you will need to add in a comma at the end of writing to line 2 or locate the cursor in the correct place before writing to line 3 (using Locate or Line).

**Mint<sup>MT</sup> Multi-Tasking** Application Note**Example code for non-e100 devices**

Note – This code should be included in the startup block

**Old - BaldorCAN**

```
'Set the node type for node 14  
NODETYPE(2, 14) = _ntKEYPAD
```

```
'Wait for node 14 to become live  
Pause NODELIVE(2, 14)
```

```
'Terminal configuration  
TERMINALPORT(_TERM3) = _tpBUS2  
TERMINALDEVICE(_TERM3) = _tdCALKEYPAD  
TERMINALADDRESS(_TERM3) = 14
```

**New - Serial**

```
'Set serial baud rate  
SERIALBAUD(_TERM3) = 57600
```

```
'Terminal configuration  
TERMINALPORT(_TERM3) = _tpSERIAL1  
TERMINALDEVICE(_TERM3) = _tdVT100
```

**New - CANopen**

```
'Set up the node on CANopen  
BUSNODE(_busCANOPEN) = 1  
BUSBAUD(_busCANOPEN) = 500  
BUSRESET(_busCANOPEN)  
Wait(1000)
```

```
'Scan the bus for Mint node 14  
NODESCAN(_busCANOPEN, 14)
```

```
'Wait for node 14 to become live  
Pause NODELIVE(_busCANOPEN, 14)
```

```
'Makes a connection between node 1 and node 14  
CONNECT(_busCANOPEN, 1, 14) = 1
```

```
'Terminal configuration  
TERMINALPORT(_TERM3) = _tpBUS1  
TERMINALDEVICE(_TERM3) = _tdBALDORKEYPAD  
TERMINALADDRESS(_TERM3) = 14  
Terminal = Terminal Or _TERM3
```

**Example code for e100 devices**

Note – This code should be included in the startup block

**Old - BaldorCAN**

N/A

**New - Serial**

```
'Set serial baud rate  
SERIALBAUD(_TERM1) = 57600
```

```
'Terminal configuration  
'e100 must use _TERM1 for serial
```

**New - CANopen**

```
'Set up the node on CANopen  
BUSNODE(_busCANOPEN) = 1  
BUSBAUD(_busCANOPEN) = 500  
BUSRESET(_busCANOPEN)  
Wait(1000)
```

```
'Scan the bus for Mint node 14  
NODESCAN(_busCANOPEN, 14)
```

```
'Wait for node 14 to become live  
Pause NODELIVE(_busCANOPEN, 14)
```

```
'Makes a connection between node 1 and node 14  
CONNECT(_busCANOPEN, 1, 14) = 1
```

```
'Terminal configuration  
TERMINALADDRESS(_TERM3) = 14  
Terminal = Terminal Or _TERM3
```