Overview

ETHERNET Powerlink (EPL) is a real-time Ethernet interface, using standard 100Mbit/s (100Base-T) Fast Ethernet (IEEE 802.3u) connections. NextMove e100, MicroFlex e100 and MotiFlex e100 are the first Baldor controllers to incorporate an Ethernet interface, providing EPL and TCP/IP networking capabilities.

The EPL protocol implemented in Mint complies with the CANopen DS402 Device Profile for Drives and Motion Control.

The EPL network can be configured in Mint WorkBench using the System Configuration Wizard. This allows the remote devices (connected to the NextMove e100) to be configured. Typically, a remote device will be a drive amplifier (a remote axis), an input / output or actuator device (remote I/O), or a combination of these.

This application note describes the keywords available to determine when both the EPL network and the remote devices are operational.

The sample Mint code included with this application note illustrates how a user-defined Mint function can be used to initialize the EPL network (later in the Application Note we will also show how this function can be stored in the Mint Library within Mint Workbench so it can be dragged straight into future Mint projects).

Note: It is important that the EPL network is initialized before the Mint program makes any calls relating to remote axes (e.g. CANCEL, DRIVEENABLE). Therefore it is typical to ensure that the sample function provided with this Application Note is one of the first things (if not the first thing) called in the Mint Startup code section.

Prerequisites

Ensure that firmware version 5554 onwards is installed on the Nextmove e100 and firmware version 5553 onwards is installed on the Microflex e100 or Motiflex e100 drive.
Initialisation Sequence

The following flow chart indicates the recommended initialization sequence to be programmed into the Nextmove e100 Mint startup block. Each of the stages below is detailed further in the following sections:

Start

- Ethernet Cyclic?
  - No
  - All Nodes Live?
    - No
    - SDO Data Operational?
      - No
      - PDO Data Valid?
        - No
        - DS402 Operation Ready?
          - No
          - Drives in Remote Mode?
            - No
            - Complete
          - Yes
            - Yes
        - Yes
      - Yes
    - Yes
  - Yes
- Yes
- Yes
- Yes
- Yes
- Yes
- Complete
Although it isn’t essential for the Mint program to wait for each check to pass successfully it is usually simpler to code it this way. However, in some applications remote nodes may be added to the network at a later, random, time (e.g. EPL nodes may be fitted to an optional machine module that is occasionally fitted to the parent machine) and so it may not be acceptable to ensure all possible network nodes initialise immediately. In these instances the application must be coded in such a way that access to uninitialised nodes is not attempted (the Mint BUS5 Event can be used to detect the appearance and disappearance of Ethernet nodes on the network).

**Detecting an Operational Bus (Ethernet Cyclic?)**

When the Nextmove e100 is powered up as the manager of an EPL network (node address F0 Hex) it will take a short time for the Ethernet Powerlink controller to become operational. The Mint keyword `BUSSTATE` can be used to detect that the EPL bus (Bus number 5 or Mint constant `_busETHERNET`) is cyclic.

```plaintext
If BUSSTATE(_busETHERNET) = _bsCYCLIC Then
    'Continue...bus is cyclic
    <Code to continue with other initialization checks>
End If
```

**Ensuring EPL Nodes are Live (All Nodes Live?)**

In the same way that CANopen nodes can be detected as live on a network using `NODELIVE`, the same keyword can be used to detect that EPL nodes are live. The sample code with this application note shows how an array of node addresses can be passed to the user-defined Mint function to provide a generic solution regardless of the number of nodes fitted. The example below illustrates how a single node (node 3) might be checked...

```plaintext
IF NODELIVE(_busETHERNET,3) = _true Then
    'Continue....node is live
    <code to continue with other initialization checks>
End If
```

**Detecting Nodes are Operational (SDO Data Operational?)**

Once the remote EPL nodes are live the Nextmove e100 should check to ensure they have entered an operational state. As far as EPL communicatons are concerned this confirms that the remote nodes are able to exchange Service Data Objects (SDOs).

There is no point continuing with the main Mint program until all EPL nodes have confirmed this operational state. The Mint keyword `REMOTESTATUS` can be used for this purpose.

```plaintext
If REMOTESTATUS(_busETHERNET,3) = _rsEPL_OPERATIONAL Then
    'Continue...SDOs are operational
    <code to continue with other initialization checks>
End If
```
Waiting for Process Data Valid (PDO Data Valid?)

Having ensured SDO communication is possible the Nextmove e100’s Mint program should now check to ensure Process Data Objects (PDOs) are operational. Mint provides a REMOTEPDOVALID keyword to simplify this process...

```plaintext
If REMOTEPDOVALID(_busETHERNET,3) = _true Then
  'Continue...PDOs are valid
  <code to continue with other initialization checks>
End If
```

At this point an EPL device other than a drive (e.g. an encoder) can be considered fully initialized and ready for operation.

If, however, a DS402 compliant drive such as a Microflex e100 or Motiflex e100 is present on the network then there are some further checks necessary that relate specifically to these drives. These additional checks are detailed in the following section.

**Initialisation of a DS402 Remote Node**

DS402 (Device Specification) is a standardized CANopen device profile for digital controlled motion for products like servo controllers, frequency converters and stepper drives.

The Microflex e100 and Motiflex e100 drives are DS402 compliant and as a result their operating states/modes are as defined by the DS402 state machine:

![DS402 State Machine Diagram]

The drive may be forced into different parts of the state machine (subject to interlocks) via the EPL control word (passed from Nextmove e100 to the remote drive). The user does not need to
worry about how to do this – state machine control is inherently handled by Mint (e.g. If the user issues DRIVEENABLE.0 = 1 and the remote drive is in the ‘Ready to Switch On’ state then it will automatically enter the ‘Switched On’ state). The Nextmove e100 can detect which state the remote drive is in via the Mint keyword AXISSTATUSWORD. This keyword returns a bit pattern corresponding to the current DS402 state (note that this keyword takes an axis as a parameter and not a node address as we’ve used earlier). The sample code with this application note shows how an array of axis numbers can be passed to the user-defined Mint EPL Initialisation function to provide a generic solution regardless of the number of EPL axes fitted.

The various DS402 states are described below (and where a Mint constant is available to represent this state this is also shown):

**Not Ready to Switch On:**

Low level (logic) power (e.g. 24Vdc) has been applied to the drive.
The drive is being initialized.
The drive is not (and cannot be) enabled.

**Switch On Disabled (aswSWITCH_ON_DISABLED):**

Drive Initialisation is complete.
The drive parameters have been set up (the drive parameter table has been loaded).
Drive parameters may be changed.
High (mains) voltage may not be applied to the drive, (e.g. for safety reasons).
The drive is not (and cannot be) enabled.

**Ready to Switch On (aswREADY_TO_SWITCH_ON):**

High Voltage may be applied to the drive.
The drive parameters may be changed.
The drive is not (and cannot be) enabled.

**Switched On (aswSWITCHED_ON):**

High Voltage has been applied to the drive.
The Power amplifier is ready.
The drive parameters may be changed.
The drive is not enabled.

**Operation Enable (aswOPERATION_ENABLED):**

No faults have been detected.
The drive function is enabled and power is applied to the motor.
The drive parameters may be changed.
(This corresponds to normal operation of the drive.)

**Quick Stop Active (aswQUICK_STOP):**

The drive parameters may be changed.
The Quick Stop function is being executed.
The drive function is enabled and power is applied to the motor.

**Fault Reaction Active:**
The drive parameters may be changed.
A non-fatal fault has occurred in the drive.
The Quick Stop function is being executed.
The drive function is enabled and power is applied to the motor.

**Fault (_aswFAULT):**
The drive parameters may be changed.
A fault has occurred in the drive.
The drive is not (and cannot be) enabled.

On power up a Microflex e100 or Motiflex e100 drive will eventually reach the ‘Switch on Disabled’ state (or beyond) and this is one of the conditions that our Nextmove e100 Mint program would normally detect before allowing the main body of the Mint program to execute.

The Mint keyword `AXISSSTATUSWORD` allocates a bit for each DS402 state. The relevant bits for network initialisation are detailed below:

<table>
<thead>
<tr>
<th>Bit No.</th>
<th>Decimal Value</th>
<th>Mint Constant</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>_aswREADY_TO_SWITCH_ON</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>_aswSWITCHED_ON</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>_aswOPERATION_ENABLED</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>_aswFAULT</td>
</tr>
<tr>
<td>6</td>
<td>64</td>
<td>_aswSWITCH_ON_DISABLED</td>
</tr>
</tbody>
</table>

If the remote drive is in any of these states then the EPL network initialisation may continue.

```vbnet
Const _btINITIALISED = (_aswSWITCH_ON_DISABLED Or _aswREADY_TO_SWITCH_ON Or _aswSWITCHED_ON Or _aswOPERATION_ENABLED Or _aswFAULT)

If (AXISSSTATUSWORD(3) & _btINITIALISED) Then
    'Continue axis is ready
    <code to continue with other initialization checks>
End If
```

Finally if the remote DS402 node is configured to operate in EPL mode (i.e. it is enabled/disabled via EPL and is either Manager Profiled or Controlled Node profiled) then the Nextmove e100 should wait for Bit 9 (512 or Mint constant _aswREMOTE) of the axis status word to be set.

```vbnet
If (AXISSSTATUSWORD(3) & _aswREMOTE) <> 0 Then
    'Continue... axis is ready for remote control
```

© Baldor UK Ltd 2008
AN00147-001 Initialising an EPL Network.doc
Adding the Sample Code to the Mint Library

As described earlier, the sample Mint code provided with this application note shows how a user-defined Mint function block can be used to initialise the EPL network (returning _true [1] when the initialisation has completed successfully).

Functions such as this (or even just small segments of Mint code) can be saved to the Mint library incorporated into Mint Workbench so that they may be recalled at will when writing new applications.

To add a code segment to the Mint Library follow this procedure...

(a) Open the Mint program containing the code to be stored (e.g. the sample program included with this application note)
(b) Either select the code to be saved in the right-hand editor window (e.g. using click and drag with the left mouse button) or, if the code to be saved is a complete module such as a Mint function, subroutine, task or event then just click on the module name in the Program Explorer view
(c) Now drag the select code/module to the Mint Library area of Mint Workbench
(d) A dialogue will appear asking you to enter a name for the new library item (you can call it anything you like!)

(e) Click on OK and a new entry will be made in the Mint Library

(f) To reuse this code in a new Mint program simply click on the Library entry and drag it into the program editor