Baldor Motion Controllers

Overview

When configured with **STEPware-100**, Maple Systems’ OIT Family Operator Interface Terminals (Maple OITs) can communicate with Baldor Mint Drive, Smart Move, and Next Move BX Motion Controllers. The **Maple OIT is the slave** in a point-to-point single master, single slave or single master, multiple slave format. This document describes the various STEPware-100 settings and provides some simple Mint examples.

<table>
<thead>
<tr>
<th>Compatible Controllers</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Family</strong></td>
</tr>
<tr>
<td>Baldor</td>
</tr>
</tbody>
</table>

Communications Cable

The Maple OIT should be connected to the RS232 or RS485 communication port.

Refer to Technical Note 1061 for information on communication cable part numbers and cable assembly instructions. If you will be assembling your own communications cable, cable assembly instructions are also available on our web site at www.maple-systems.com.

**WARNING:** If your communications cable is not wired exactly as shown in our cable assembly instructions, damage to the Maple OIT or loss of communications can result.
### STEPware-100 Settings

The following table lists the communications settings that must be configured in STEPware-100.

Please note:
- the Settings column lists STEPware-100’s settings; for the Baldor Controllers
- the Options column lists STEPware-100’s options; your Controller may not support every option

<table>
<thead>
<tr>
<th>Name</th>
<th>Settings</th>
<th>Options</th>
<th>Important Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baud Rate</td>
<td>9600 (All Models)</td>
<td>19200, 9600, 4800, 2400, 1200, 600, 300</td>
<td>Must match the controller port settings. Use the fastest baud rate supported by both.</td>
</tr>
<tr>
<td>Parity</td>
<td>None</td>
<td>Even, Odd, None, Mark, Space</td>
<td></td>
</tr>
<tr>
<td>Data Bits</td>
<td>8</td>
<td>7, 8</td>
<td></td>
</tr>
<tr>
<td>Stop Bits</td>
<td>1</td>
<td>1, 2</td>
<td></td>
</tr>
<tr>
<td>Operating Mode</td>
<td>Block</td>
<td>Interactive, Block, Network Modes</td>
<td>Attached sample code requires Block Mode.</td>
</tr>
<tr>
<td>Line Terminator</td>
<td>CR</td>
<td>CR, LF, CR/LF, ETX</td>
<td>Attached sample code requires CR.</td>
</tr>
<tr>
<td>Turn-around Delay</td>
<td>No Delay</td>
<td>50, 100, 250, No Delay</td>
<td></td>
</tr>
<tr>
<td>Handshaking</td>
<td>None</td>
<td>None, Xon/Xoff, RTS/CTS, both</td>
<td></td>
</tr>
<tr>
<td>Password</td>
<td>00000</td>
<td>Any 5-digit value</td>
<td>Application specific</td>
</tr>
<tr>
<td>Local Echo</td>
<td>Enabled</td>
<td>Enabled/Disabled</td>
<td></td>
</tr>
<tr>
<td>Local Setup</td>
<td>Enabled</td>
<td>Enabled/Disabled</td>
<td></td>
</tr>
<tr>
<td>Local Keyboard</td>
<td>Enabled</td>
<td>Enabled/Disabled</td>
<td></td>
</tr>
<tr>
<td>Key Click</td>
<td>Enabled</td>
<td>Enabled/Disabled</td>
<td></td>
</tr>
<tr>
<td>Block Echo</td>
<td>Enabled</td>
<td>Enabled/Disabled</td>
<td></td>
</tr>
<tr>
<td>Delayed Line Feed</td>
<td>Disabled</td>
<td>Enabled/Disabled</td>
<td></td>
</tr>
<tr>
<td>Append Line Feed</td>
<td>Disabled</td>
<td>Enabled/Disabled</td>
<td></td>
</tr>
<tr>
<td>Use 3-Wire RS485</td>
<td>Disabled</td>
<td>Enabled/Disabled</td>
<td>Available only in Network Mode.</td>
</tr>
</tbody>
</table>

**NOTE:** STEPware provides a full range of display and control functions. See STEPware Help, under *ESCape Control Commands*, for a complete list.
Sample Programs

Below are fragments of Mint code for performing common tasks.

In each of the following examples, the OIT should be set for Block mode. The Line Terminator should be set as CR.

**Setting/Showing Drive Speed**

```
PRINT “New Speed: “
INPUT NewSpd
JOG = 10
GO
PRINT “Speed is “NewSpd
WAIT = 2000
STOP
```

**Setting/Showing Drive Position**

```
HOME = 1
PRINT “New Position: “
INPUT NewPos
SPEED = 250
MOVEA = NewPos
GO
PRINT“Position is “NewPos
```

**Starting/Stopping the Motor from the OIT**

Using Function Key ASCII Strings, the OIT can send messages to Start, Stop, and adjust the motor speed.

The OIT’s Function Keys should be programmed as follows:

- **F1**: 1{CR}
- **F2**: 2{CR}
- **F3**: 3{CR}
- **F4**: 4{CR}

Each key should have the **Send Immediately** option checked.
Starting/Stopping the Motor from the OIT (continued)

Use the following Mint code:

```plaintext
REM F1 is Start, F2 is Stop, F3 is Increase Speed, F4 is Decrease Speed
InitSpeed = 100 :REM Initial Speed
SpdChange = 25 :REM Speed Increment/Decrement Amount
MaxSpeed = 500 :REM Max Speed
SPEED = InitSpeed
LOOP
KEY = 0 :REM Wait for key to be pressed
WHILE KEY = 0
KEY = INKEY :REM Read keyboard
IF KEY = '1' THEN GOSUB START_MOTOR
IF KEY = '2' THEN GOSUB STOP_MOTOR
IF KEY = '3' THEN GOSUB INCR_SPD
IF KEY = '4' THEN GOSUB DECR_SPD
ENDW
ENDL
END :REM all done

#START_MOTOR
REM Start Motor Subroutine
SPEED = InitSpeed
J O G=1 0
GO
RETURN

#STOP_MOTOR
REM Stop Motor Subroutine
SPEED = 0
STOP
RETURN

#INCR_SPD
REM Increase Motor Speed
InitSpeed = InitSpeed + SpdChange :REM get new speed
IF InitSpeed <= MaxSpeed THEN :REM don’t exceed max speed
SPEED = InitSpeed :REM set new speed
ENDIF
RETURN

#DECR_SPD
REM Decrease Motor Speed
InitSpeed = InitSpeed - SpdChange :REM get new speed
IF InitSpeed >= 0 THEN :REM don’t allow 0 or negative speed
SPEED = InitSpeed :REM set new speed
ENDIF
RETURN

Displaying a Pre-Programmed Message

The Motion Controller can instruct the OIT to display a pre-programmed message. This can be useful for showing alarm or status messages, without having to place code in the Motion Controller for a lot of text messages.

The following Mint code will display message numbers 1-25.

```plaintext
FOR MSG = 1 TO 25
REM format a message to the OIT: ESC m [msg num] STX
PRINT BINARY '27';"m";MSG;BINARY '2';
WAIT 300
NEXT MSG
```

Sounding the OIT’s Built-In Buzzer

The Motion Controller can instruct the OIT to sound its buzzer for a specified number of seconds.

The following Mint code will sound the OIT’s buzzer for 2 seconds.

```plaintext
'RE format a message to the OIT: ESC g [seconds] STX
PRINT BINARY '27';"g2";BINARY '2';
```

NOTE: STEPware provides a full range of display and control functions. See STEPware Help, under ESCape Control Commands, for a complete list.