Configuring digital I/O with MATLAB® Data Acquisition Toolbox

These instructions explain how to set the direction of digital ports and bits from a Measurement Computing (MCC) data acquisition device using MATLAB Data Acquisition Toolbox*. The Measurement Computing brand miniLAB 1008 is used in this example.

*This example was tested with MATLAB version 7.

1. Run InstaCal and note the board number assigned to the miniLAB 1008.

   In this example, the miniLAB 1008 is assigned Board# 0.

   Perform the following steps to control the miniLAB 1008 device's digital ports.

2. Run MATLAB and enter \texttt{dio = digitalio ('mcc',0)} at the >> prompt in the Command Window.

   This command creates a DIO object using board# 0 — the miniLAB 1008 — as an MCC-type device in MATLAB.

   After you enter this command, the screen updates with miniLAB 1008 device information:

   ![Command Window](image)

   ```
   >> dio = digitalio ('mcc',0)
   Display Summary of DigitalIO (DIO) Object Using 'miniLAB 1008'.
   
   Port Parameters: Port 0 is line configurable for reading and writing. Port 1 is port configurable for reading and writing. Port 2 is port configurable for reading and writing. Port 3 is port configurable for reading and writing. Port 4 is port configurable for reading and writing.
   
   Engine status: Engine not required.
   
   DIO object contains no lines.
   >>
   ```

3. At the >> command prompt, enter \texttt{addline(dio,4:27, 'out')} The Command Window updates with the port information.
This command adds digital I/O lines and configures them for output (‘out’).

The `addline` method is zero-based — a device with 28 bits is written as 0 to 27. The miniLAB 1008 has 28 digital I/O lines: four DIO channels are on the top screw terminals (channels 0 to 3), and 24 DIO channels are on the 37-pin D connector (channels 4 to 27).

On this window, indices 1 to 8 are `FirstPortA`, indices 9 to 16 are `FirstPortB`, indices 17 to 20 are `FirstPortCL`, and indices 21 to 24 are `FirstPortCH`.

4. Use the `putvalue` command to set the logic state of a bit, port, or any number of bits you choose to control.

The example commands below control `FirstPortA`, bit 0.

The `putvalue` command is one-based — a device with 28 bits is written as 1 to 28.
To turn FirstPortA bit 0 on, enter `putvalue(dio.line(1),1)`.

FirstPortA bit 0 is index number 1.

To turn FirstPortA bit 0 off, enter `putvalue(dio.line(1),0)`.

To turn all 8 bits contained in FirstPortA on, enter `putvalue(dio.line(1:8), 255)`.

To turn all 8 bits contained in FirstPortA off, enter `putvalue(dio.line(1:8), 0)`.

You control the other ports in the same manner.

To control the individual bits contained in FirstPortB, but still update the entire port at once, enter `putvalue(dio.line(9:16),logical([111000101]))`.

Remember, indices 9 to 16 are FirstPortB.

5. When you are done controlling the digital channels, enter the following commands at the `>>` prompt to remove the DIO object from memory and from the MATLAB workspace.

```matlab
delete(dio);
clear('dio');
```

The full MATLAB **Command Window** display is shown here.
Controlling digital I/O with MATLAB with MATLAB® Data Acquisition Toolbox

For more information on digital I/O operations with the MATLAB Data Acquisition Toolbox, refer to Chapter 7 of the Data Acquisition Toolbox User’s Guide (Version 2) at