

Simple Analog Input Acquisition Using a Data Translation DT9836 module and MATLAB R2017b

After successfully configuring a Windows system to use a Data Translation DT9836 USB module with MATLAB R2017b (reference KB article - <http://kb.mccdaq.com/KnowledgebaseArticle50741.aspx>), the following lines were used in a .m script to acquire and display data of an analog input signal.

```
% Begin by creating a session.
s = daq.createSession('dt');

% Add an analog input channel using Data Translation's Open Layers device ID and channel
ID (0),
% with the measurement type set to 'Voltage'.
ch0 = addAnalogInputChannel(s,'DT9836(00)', '0', 'Voltage');

% Configure analog input channel properties
% Reference the DT device's user manual to verify supported terminal configurations (i.e.
'SingleEnded', 'Differential').
ch0.TerminalConfig = 'SingleEnded';
ch0.Range = [-10,10];

% Set the acquisition rate (Hz).
s.Rate = 10000;

% Set the input event handler by adding a listener for 'DataAvailable' events.
% The callback function plots the acquired data against time.
lh = addlistener(s,'DataAvailable', @(src,event) plot(event.TimeStamps, event.Data));

% Set the acquisition to run continuously in the background for a specified duration.
s.IsContinuous = true;
startBackground(s);
durationInSeconds = 3;
localTimer = tic;
while s.IsRunning && toc(localTimer) < durationInSeconds
    pause(1)
    fprintf('Scans Acquired = %d\n', s.ScansAcquired)
end

% Cleanup
stop(s);

% Delete the listener.
delete(lh);
delete (s)
```

Simple Analog Input Acquisition Using a Data Translation DT9836 module and MATLAB R2017b

*NOTE: This script may also work with other Data Translation devices or else require minimal code changes in order to acquire and display data.

Measurement Computing Data Acquisition Knowledgebase

<https://kb.mccdaq.com/KnowledgebaseArticle50744.aspx>