



plug & play instruments  
oscilloscopes

Phone +64 9 524 7456  
Fax +64 9 524 7457  
Email info@cleverscope.com  
28 Ranfurly Rd, Epsom  
P.O. Box 26-527  
Auckland 1023  
New Zealand

3 Nov 2010

## Cleverscope – use with Matlab

### Summary

Cleverscope includes the Maths Equation Builder, and Maths graph to calculate and display waveforms derived from the measured signals. The Maths Equation builder is used to define up to 10 linked Maths Equations which are then executed whenever a new signal is measured. The Maths Equation builder is used to define an **equation**, which executes on every sample separately, and a **process** which operates longitudinally over the entire waveforms' sample set. Example processes are filters, integration, differentiation and signal wide derived values such as RMS, power factor etc.

Cleverscope includes 4 processes that can be used to call external Matlab .m functions, considerably extending the capabilities of Cleverscope in analysing waveforms.

### Executing Matlab Functions

The maths Equation builder is used to set up the equation line that calls the Matlab.m functions. Here are examples:



Pass channels **a** and **b** to Matlab cscope1.m function. Return the result to channel **a**.



Pass (channel **c** multiplied by 3.2 added to sine(channel **a** times 12)) and digital input 3 to Matlab cscope2.m function. Return the result to channel **b**.

### Matlab .m functions

Cleverscope assumes the existence of 4 Matlab .m functions used for interface. These functions can in turn call other Matlab functions, including user interface activity, if needed. The 4 functions are named **cscope1.m** to **cscope 4.m**.

The functions each have a common header:

Function  $y = \text{cscope}(a,b,n,T0,dt)$

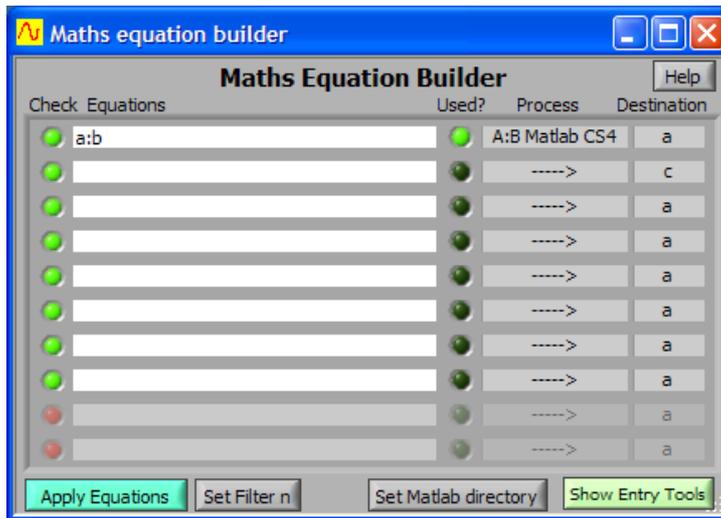
This function expects the following values:

- **a** – an array of values input as A in the A:B
- **b** – an array of values input as B in the A:B
- **n** – the number of samples in the arrays A and B, a scalar.
- **T0** the start time of the Maths graph as a scalar. For example -0.023 means the graph starts at -23ms.
- **dt** – the sample interval, in s, as a scalar. For example 0.000005 means a sample interval of 5us.
- **y** is the array passed back to the Cleverscope maths equation builder.

# Setting up for Matlab Use

Before using Matlab you need to do these things:

1. Install the Matlab environment on your PC.
2. Write the cscope1.m .. cscope4.m files you will use.
3. Set the directory in which the cscope1.m .. cscope4.m files can be found. Click on the 'Set Matlab directory' button, navigate to the directory, and choose 'Select Current Directory'



4. Have the Maths graph visible. On the first execution of the Maths Equations, Matlab will open, and there will be a considerable delay to execution of the .m function. After that execution is much faster.
5. When the Cleverscope application is closed, Matlab will also be closed (if it was opened by Cleverscope).
6. You can obtain help by clicking on the Help button.
7. **Important:** Matlab caches the cscope1..4.m files internally to improve speed. Any edits of the .m file after execution has started will have no effect until after Cleverscope is closed and re-opened. Cleverscope does plan to introduce a button to update the cache in the near future.

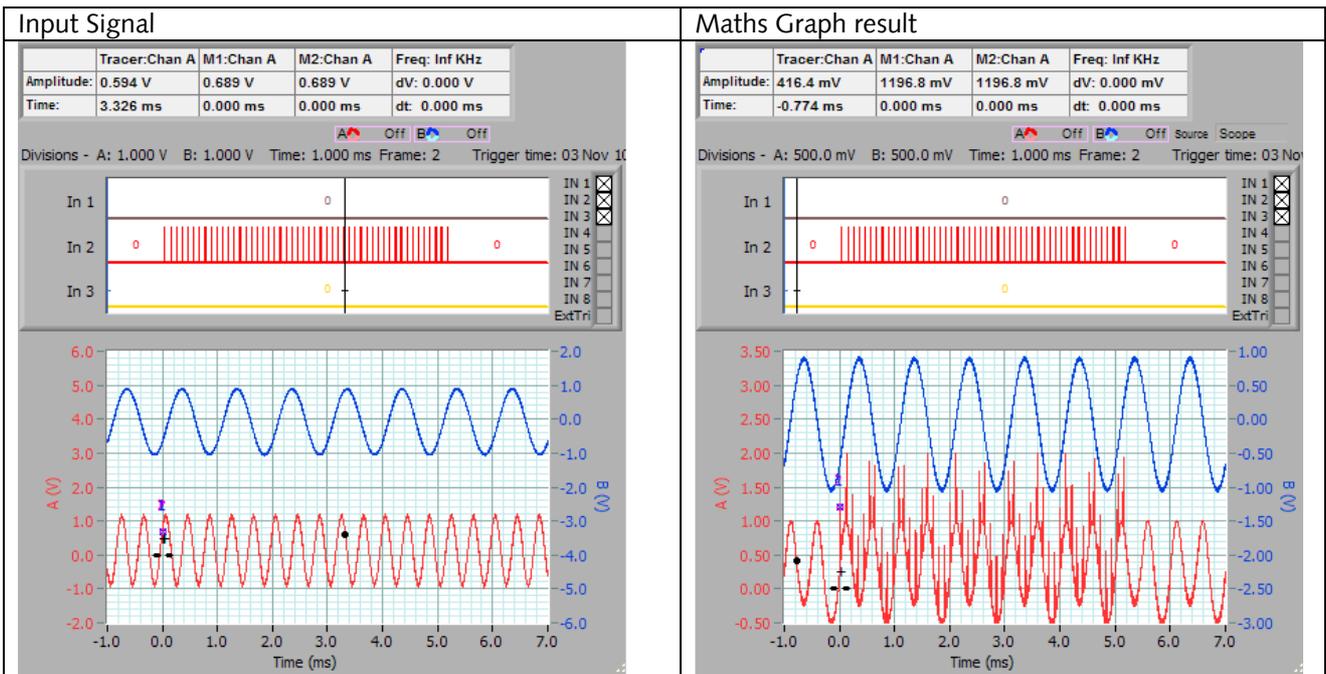
# Examples

These examples illustrate use of the Matlab functionality:

**Cscope2.m** - adds two channels together and returns result

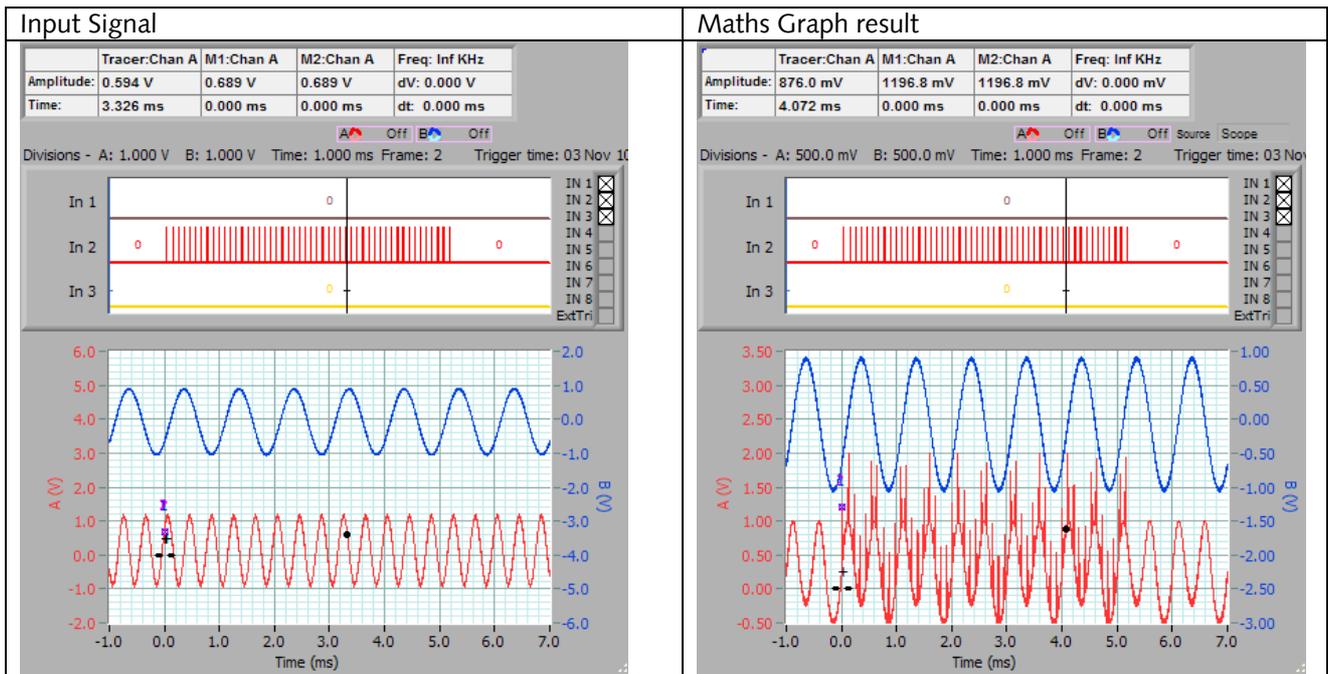
function  $y = \text{cscope}(a,b,n,T0,dt)$

$y = a+b$  %add inputs together

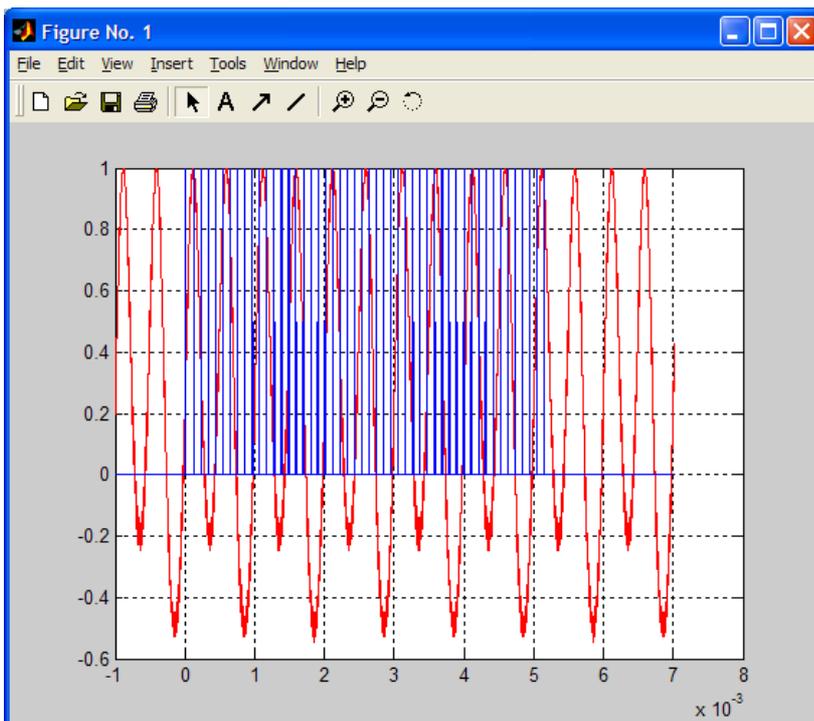


**Cscope1.m** – adds two channels together, returns result, and plots **a** and **b** in red and blue in a Matlab Figure.

```
function y = cscope(a,b,n,T0,dt)
y=a+b; %add the signals together
time=((0:n-1)*dt)+T0; % Calc graph time axis
figure (1) % Plot data to fig 1
plot(time, a, 'r', time, b, 'b') %plot the data
grid on
```



Matlab output:



## Using Matlab for further processing

We have cscope4.m defined as:

**Cscope4.m** – plots the spectrogram of channel a

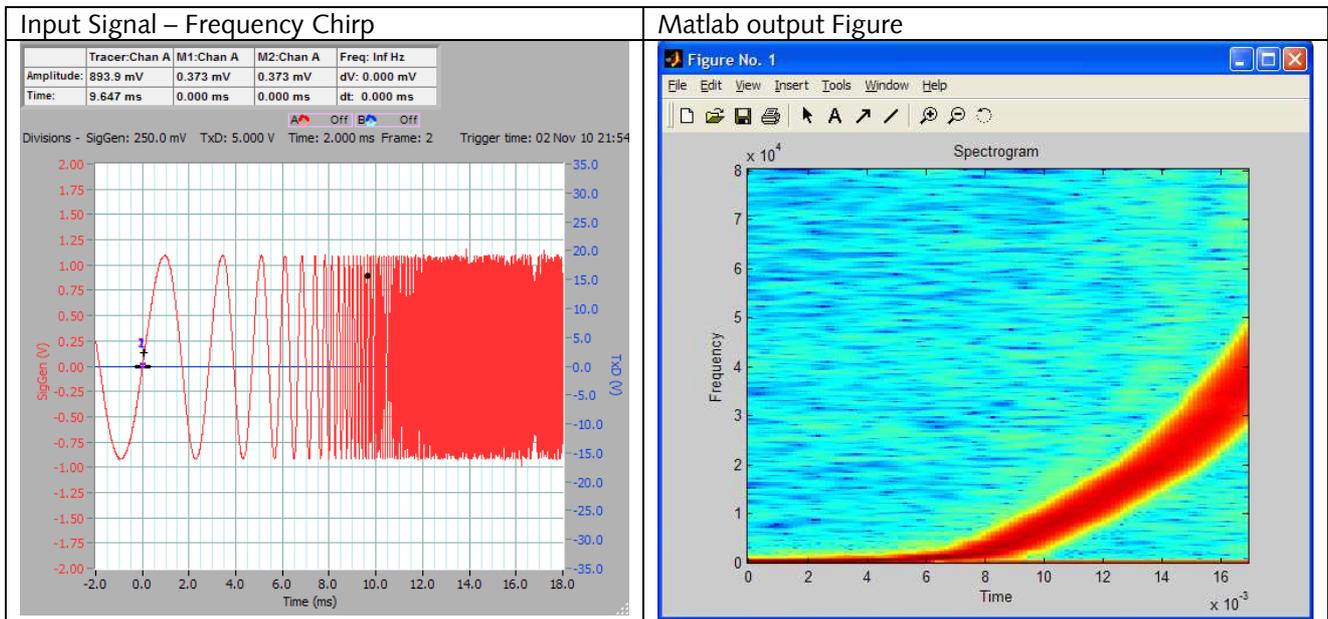
function y = cscope(a,b,n,T0,dt)

y = a

figure (1) % Plot data to fig 1

specgram(a,512,1/dt,500,475)

title('Spectrogram')



## Conclusion

With up to 4 processes to call Matlab with, and fully flexibility on the data sent to Matlab, you have a great deal of scope in using Matlab to extend the already high Cleverscope functionality.

Good luck with your new Matlab extensions!