Sysblocks

Systems, signals, DSP, FFT

Teacher Guide

MATRIX

CP2398

Teacher’s notes

Learning objectives

On successful completion of this course, the student will be able to:

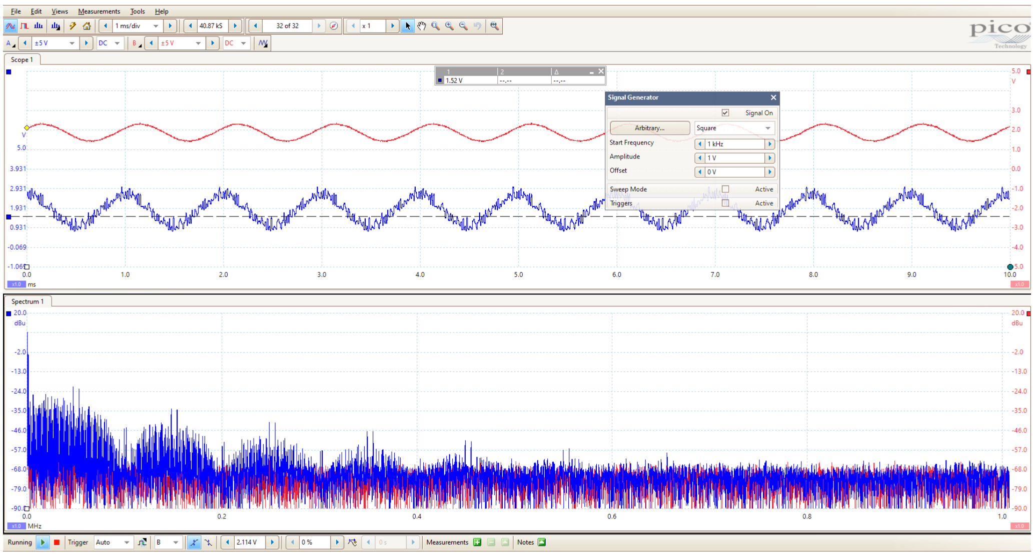
- explain what is meant by sampling an analogue signal;
- describe the trade-off between sample-rate and file size;
- explain the meaning of the term ‘quantisation error’;
- explain the significance of Nyquist’s theorem to sampling an analogue signal;
- describe what is meant by ‘aliasing’ in connection with sampling a signal;
- describe the following techniques:
  - Analogue-to-digital conversion;
  - Digital-to-analogue conversion;
- create a ‘heartbeat’ flashing LED to show that the program is running;
- use an oscilloscope to observe the waveform of a signal;
- use a spectrum analyser and analyse the resulting trace;
- explain the need for analogue-to-digital conversion when processing signals;
- state the meaning of the term ‘resolution’ applied to analogue-to-digital conversion;
- count up using binary numbering;
- interpret voltage measurements to obtain the quantisation ‘step’ voltage;
- describe the factors that determine the size of the quantisation ‘step’ voltage;
- identify three common sources of electrical noise;
- distinguish between ‘white’ noise and ‘pink’ noise;
- describe the significance of signal-to-noise ratio (SNR);
- give two ways in which SNR can be increased;
- state the meaning of the term phase difference applied to two sinusoidal signals;
- describe the use of an ‘X-Y’ plot to identify this phase difference;
- change the configuration of a DSP wave generator to adjust the frequency and phase of the signal produced;
- use DSP blocks to add or subtract two signals and output the result;
- state the meaning of the term ‘wrap around’ and explain how it happens for a system using ‘signed int’ data type;
- outline the significance of the Fourier theorem to signal synthesis;
- describe how a square wave can be made from an infinite series of sinusoids of appropriate frequency and amplitude;
- distinguish between full Fourier Transform and Fast Fourier Transforms;
- state one advantage of digital over analogue filters;
- distinguish between the effect of a low-pass and high-pass filter on a noisy signal.

Instrumentation

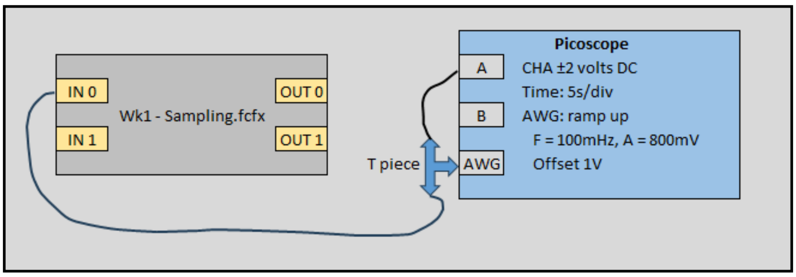
Throughout this course we have used a Picoscope. This is a PC based instrument that includes:

- Oscilloscope
- XY plotter
- Spectrum analyser
- Arbitrary waveform generator

You don’t have to use a Picoscope - there are man alternatives: separate instruments and other manufacturers of PC based instruments like Voltcraft. However the Pico scope works very well: in particular the dual views that include both Oscilloscope and Spectrum analyser are very good.

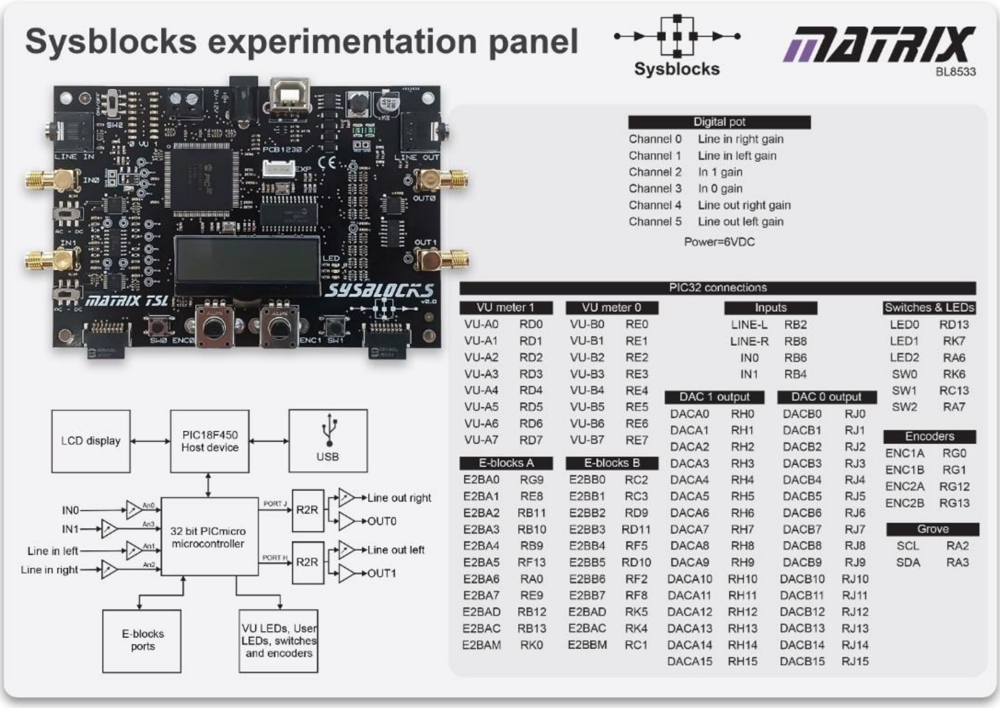


This image shows Oscilloscope, Spectrum analyser and Signal generator instruments on a Picoscope.



For each experiment we have drawn the connections simply in Excel and given an image that shows the configuration of the system as well as the settings on the Oscilloscope.

Equipment needed



To deliver this course you will need:

Picoscope or other instrument which includes:

- Oscilloscope
- Spectrum analyser
- Arbitrary waveform generator
- XY plotter

And the items in the table which are included in Matrix product code BL8386 - Standard Sysblocks Experimentation panel.

Qty	Code	Description
1	HP4039	Tray Lid
1	HP2045	Shallow tray
1	HP8600	Crash Foam
1	SUB8801	Standard sysblocks panel with PCB
1	HPUSB	USB lead
3	BL4585	BNC-SMA adaptor
2	BL6889	SMA to SMA lead
1	BL6374	BNC splitter
1	HP6401	3.5mm jack speaker
1	HP6400	Stereo 3.5mm plut to plug lead 1m

Teacher's notes

Systems and Signals

	Notes

Worksheet 1
Sampling

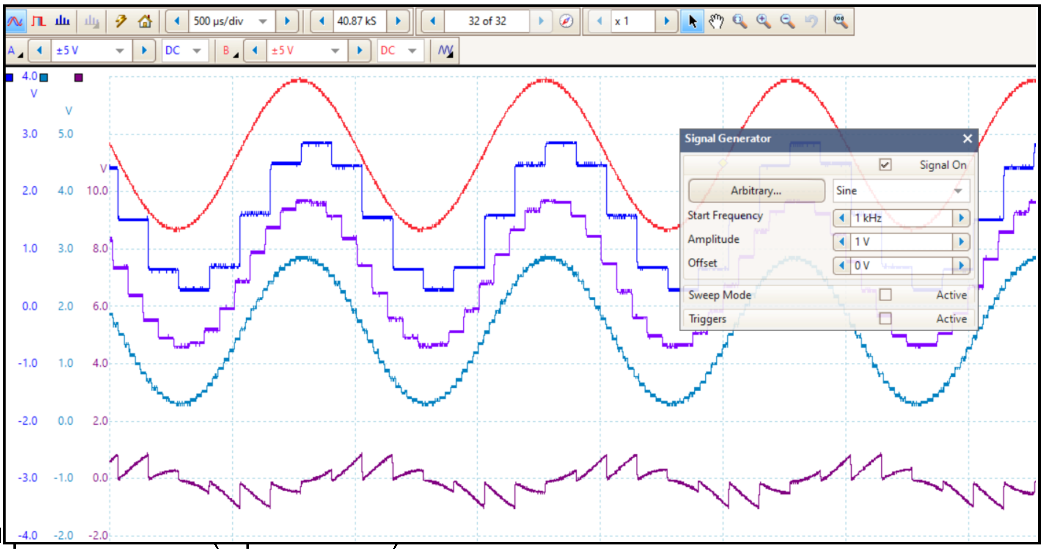
Concepts involved:

sampling *sample rate* *file size* *quantisation* *interrupt*

The program takes the sinusoidal signal from a source such as the Picoscope AWG and samples it at various sample rates, set by encoder 'ENCO'. Students may need help in setting up the AWG or other signal source.

Students are asked to listen to and compare the quality of what they hear. It may be desirable to use headphones, rather than speakers, depending on how many groups are involved in the activity at the time!

Typical results are shown below, together with the AWG settings panel:

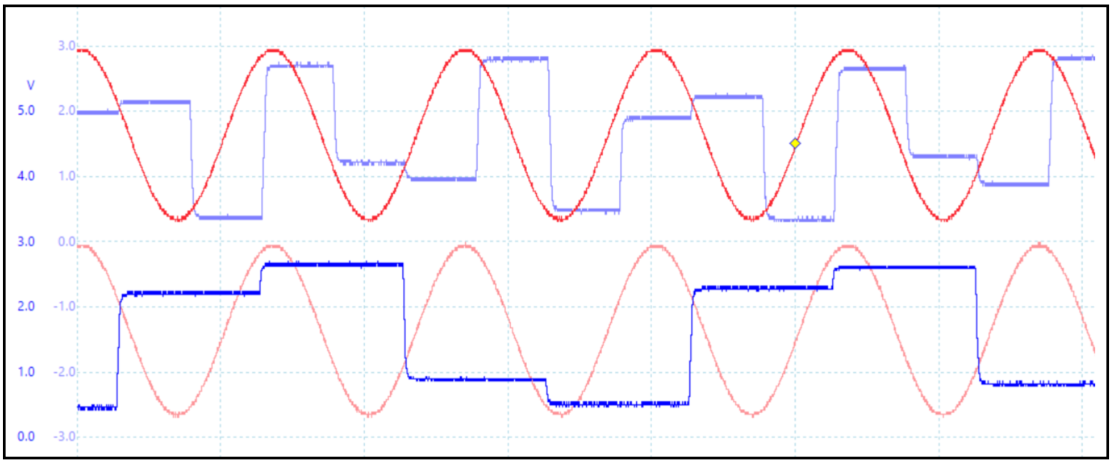
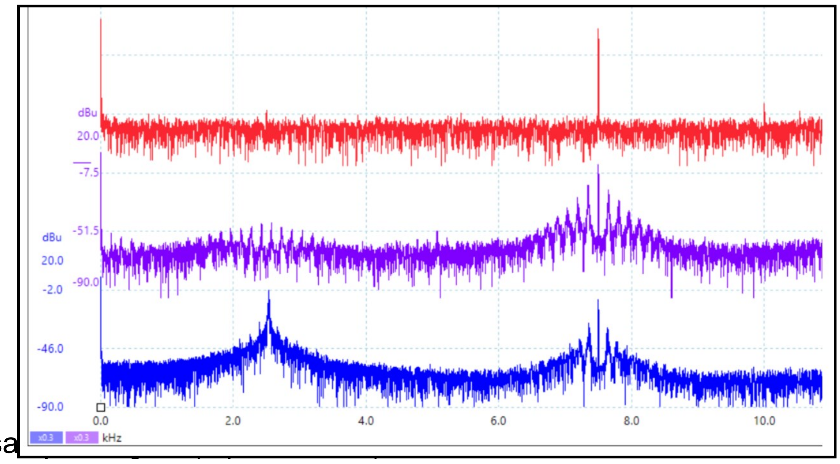


Ou

- Input sine wave
- Output at 8 kHz sample rate
- Output at 16 kHz sample rate
- Output at 44.1 kHz sample rate
- Input minus output for 8 kHz - the quantisation noise.

Teacher's notes

Systems and Signals

	Notes
Worksheet 2 Nyquist's theorem	Concepts involved: *Nyquist's theorem* *aliasing* *spectrum analyser* The program builds on the previous investigation and looks at the consequences of Nyquist's sampling theorem. Students may need assistance in recognising the onset of aliasing. They could tabulate results and draw a graph of the apparent frequency of the reconstructed waveform versus the frequency of the input waveform. Putting in a square wave shows the spectrum filling as square waves are made up of odd harmonics. Typical results are shown below:  Output waveforms (top to bottom): Top: 7.5 kHz sine wave sampled at 20 kHz Bottom: Same wave sampled at 10 kHz, aliasing effect clearly seen.  Spectrum of sa Input signal at 7.5 kHz Sampled at 20 kHz showing some quantisation noise Sampled at 10 kHz showing alias around 2.5 kHz

	Notes
Worksheet 3 Analogue to digital conversion	Concepts involved: *analogue signal* *digital signal* *A to D conversion* *binary numbering* *ADC resolution* First of all, the input is an analogue signal in the form of a rising ramp. The program converts this to a digital signal, as a binary number, and displays it on the Sysblocks LED array 'VU'. Students may wish to video the sequence in slow motion to convince themselves that it produces a binary sequence. The second part of the investigation uses the AWG to input a steady DC signal. Again, students may need guidance in setting up the AWG to do this. The aim is to produce a graph from which they can determine the resolution of the ADC.

Notes

Worksheet 4 Digital to analogue conversion

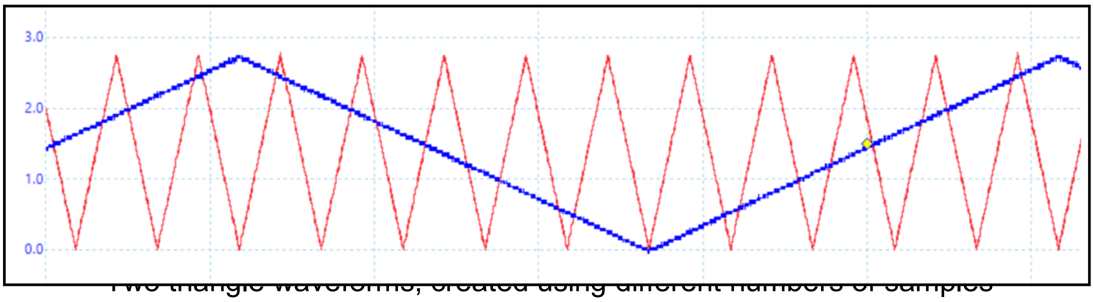
Concepts involved:

quantisation step

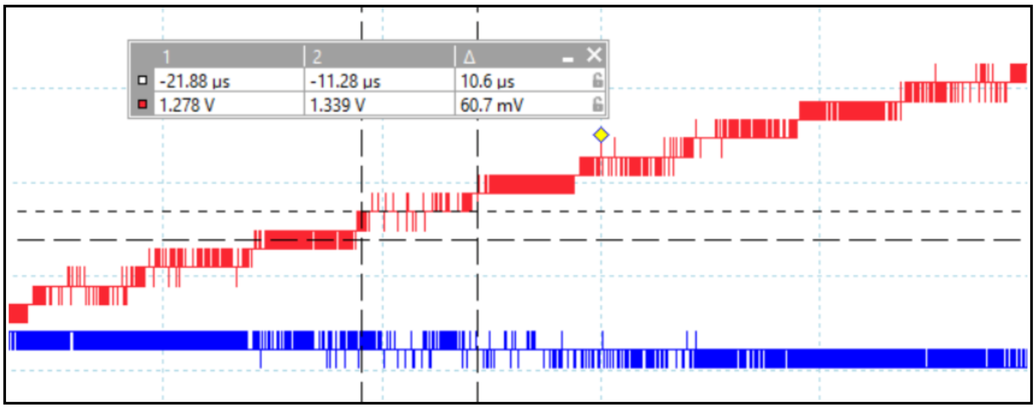
To begin with, students input a 3-bit digital number using the switches on the Sysblock board. The program generates the equivalent analogue voltage and outputs it to be viewed on the oscilloscope. In this way, students build up a table of equivalent analogue voltages for the eight binary numbers possible at the input and can determine the quantisation step for the DAC.

In the second program, students run a program which generates triangular waveform signals. It is shown this involves the same DAC conversion process.

Typical results for the second program are shown below:

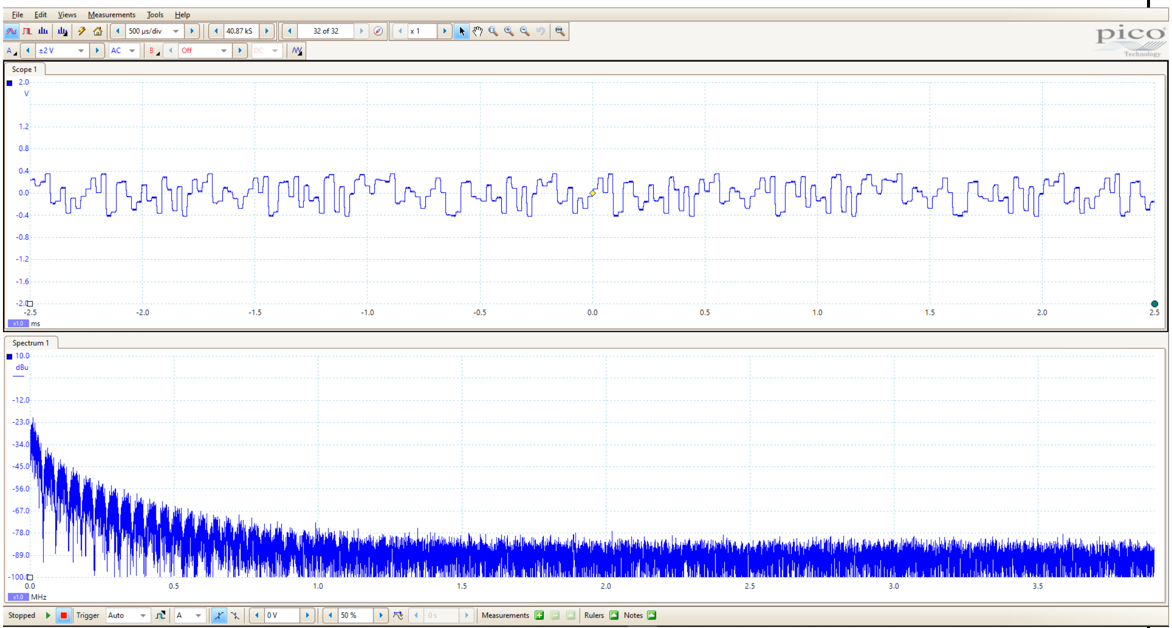


Two triangle waveforms, created using different numbers of samples



Teacher's notes

	Notes
Worksheet 5 **Noise**	Concepts involved: *sources of noise* *noise vs distortion* *noise floor* *white noise* *pink noise* The activity could begin with a discussion about the meaning of noise and common sources of noise. The students could be asked to research the difference between 'white' and 'pink' noise. Truly 'white' noise is rare and is difficult to generate. The DSP noise generator uses 'random' numbers, but truly random numbers are equally difficult to generate! The spectrum analyser trace of the noise signal should show a signal spread equally across a range of frequencies. Computer generated noise will never be equal across the spectrum.



Spectrum view of 'white' noise. (Scale is in MHz)

Teacher's notes

Systems and Signals

	Notes

Worksheet 6
Signal to noise ratio

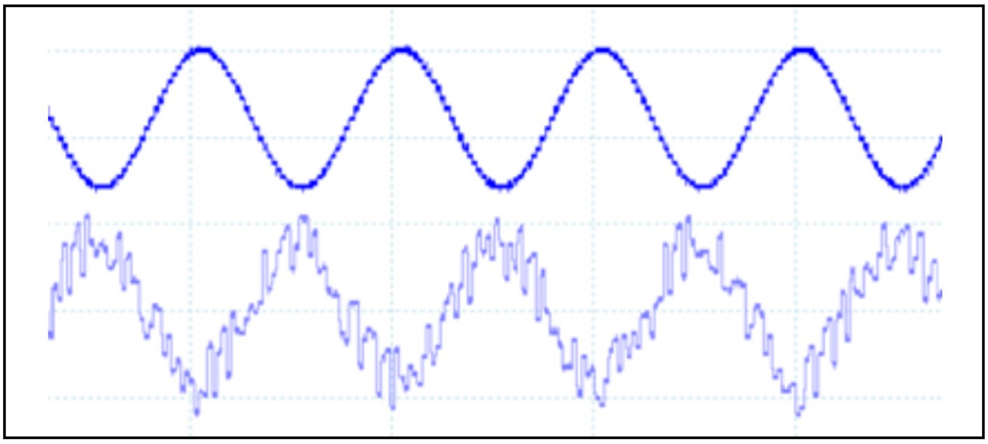
Concepts involved:

signal to noise ratio *decibel* *bandwidth*

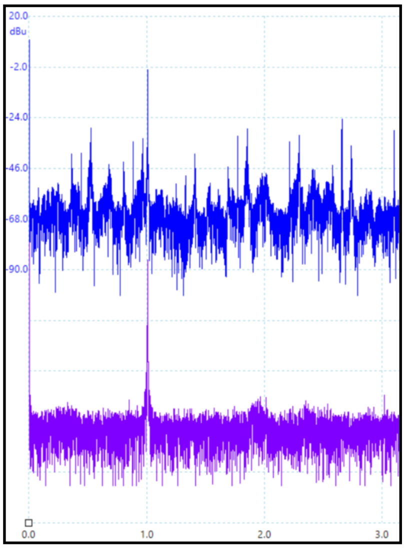
The program allows the student to see what a signal looks like with different levels of added noise. Ideally, the spectrum for the signal should show one sharp peak at the frequency of the sinusoidal signal and a broad fuzzy noise floor. As the noise level is increased, the peak should remain unaltered while the noise floor rises across the whole spectrum.

There could be a discussion about measures to improve the SNR such as controlling the bandwidth and using spread spectrum techniques.

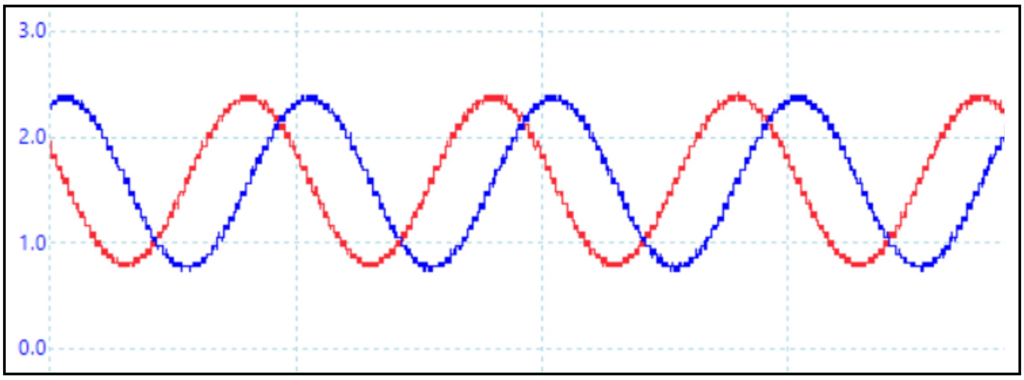
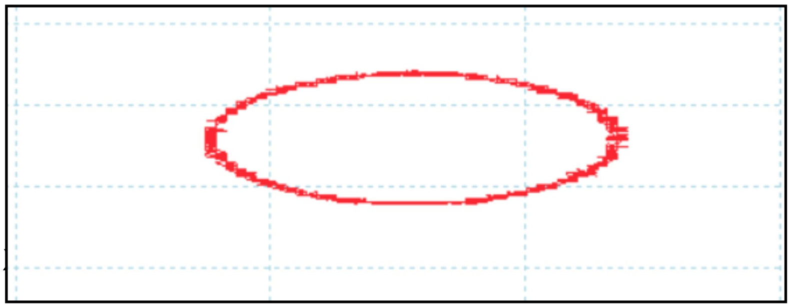
Typical results are shown below:



Sinusoidal signal with and without noise

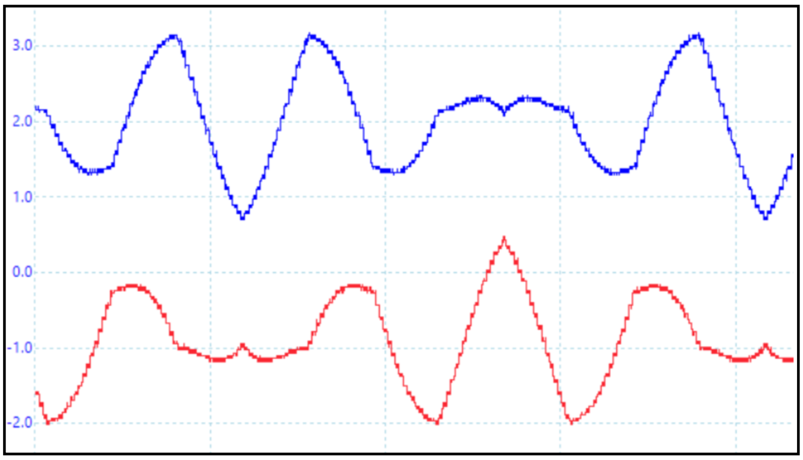


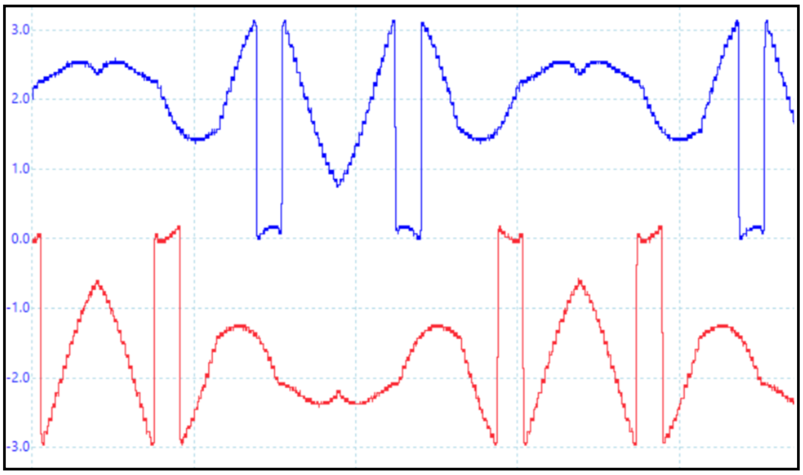
Offset Spectra of clean and noisy signals

	Notes
Worksheet 7 **Phase**	Concepts involved: *phase* *oscilloscope X-Y mode* *lissajous figures* The program generates two sinusoidal signals with the same frequency but with a phase difference between them. Seen using the ‘X-Y’ plot on the oscilloscope, this causes a circle. However, the students are more likely to create an ellipse because of differences in the vertical and horizontal scales and may need support to understand this. This effect can be seen clearly in the X-Y view shown below. They then ‘tweak’ the settings to change first of all the phase difference and then the frequency of the second signal to see the effect on the ‘X-Y’ plot. To some students it may not be obvious why changing the number of samples in each cycle of the wave alters it frequency. Typical results are shown below:  

<table>
<tr><th></th><th>Notes</th></tr>
<tr><td>Worksheet 8
Manipulating signals</td><td>
Concepts involved:
number types used in computing ‘wrap around’
examples of signal manipulation e.g. modulation, convolution, Fourier transforms
Signal manipulation is arguably the most important of techniques in DSP. Some areas involve very complex mathematical operations, but the options built into Sysblocks allow these to be investigated without mathematical ‘pain’. The first exercise shows that some unintended consequences can occur, such as ‘wrap around’. For students to appreciate fully what is happening, there may need to be a detailed discussion of number types used.
Students are challenged to explore the effects of changing the parameters used in the program.

Typical results are shown below:





</td></tr>
</table>

Teacher’s notes

Systems and Signals

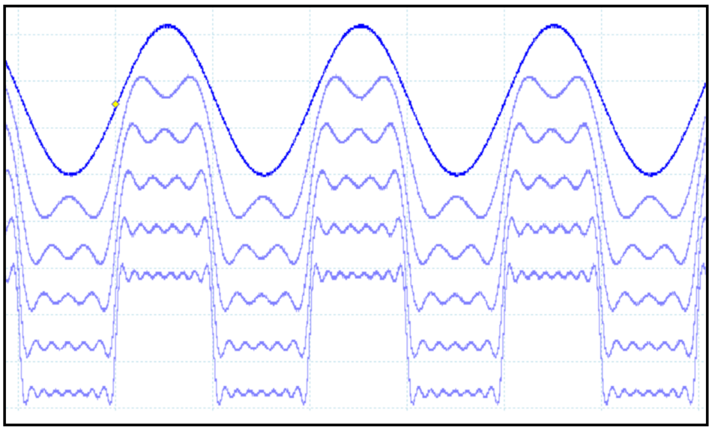
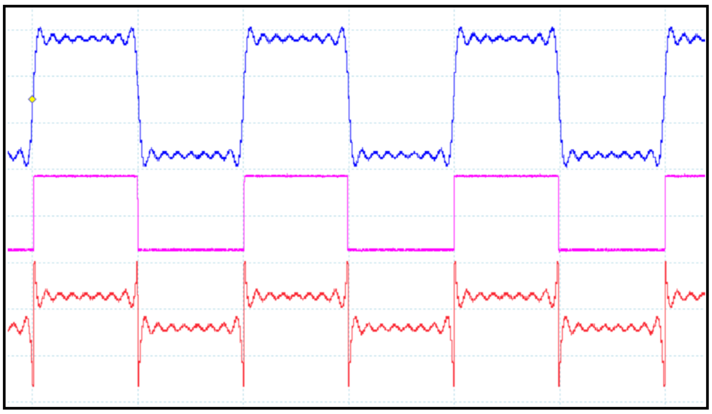
	Notes
Worksheet 9 Level detection	Concepts involved: *peak* *amplitude* *trough* *AC rms* *true rms* The investigation could be introduced with a review of applications using electrical measurements like those taken here. The student varies the amplitude of the signal, using an encoder and views the results both on the oscilloscope trace and on the LCD screen. The challenge swaps the internally generated signal for an external one. The students investigate it in similar fashion. This program uses the ‘signed int’ data type, mentioned in worksheet 8. The output port component uses ‘autoscaling’ meaning that a measurement of -32768 gives an output of approximately 0V, a measurement of +32767 gives 3.3V (the full scale value) and a measurement of zero produces a mid-point value of 1.65V.

Teacher's notes

<table>
<tr><th></th><th>Notes</th></tr>
<tr><td>Worksheet 10
Fourier
analysis</td><td>
Concepts involved:
fundamental   harmonic
Understanding Fourier's theorem involves wading through advanced mathematical operations. Using the DSP system shown here, takes all the pain out of that!
There are two versions of the program. The first demonstrates Fourier transformations directly but allows little experimentation. The second allows the student to build up the waveform, harmonic by harmonic. In the process, they use two measures of how well the constructed waveform matches a square wave:

viewing the trace from 'OUT1', the difference between the generated waveform and an actual square wave;
measuring the AC rms value of that waveform. (Students should be advised to wait a short while after each adjustment for the reading to stabilise.)

Those capable are invited to modify the waveform to try other shapes.

Typical results are shown below:

Succession of waveforms showing the effect of adding more harmonics

Square wave built from eight sine waves (top trace), ideal square wave (centre) and the error between them (bottom trace).
</td></tr>
</table>

Teacher's notes

Systems and Signals

Notes

Worksheet 11 Fourier basics

Concepts involved:

FFT *frequency bin (interval)*

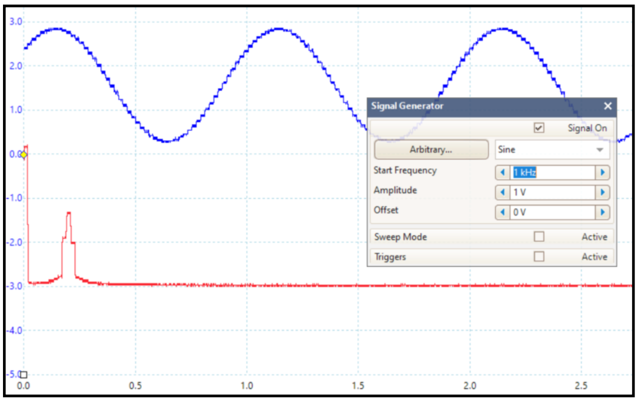
Using FFT reduces the mathematical depth. It ignores those frequency components that contribute little to the outcome. The result can reduce the number of computations by a factor of 10 in some cases, speeding up the process significantly.

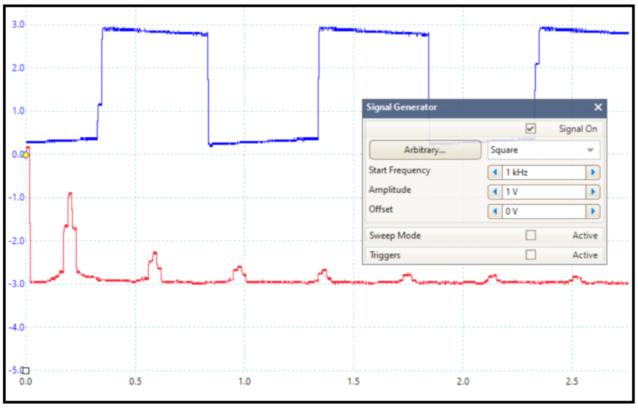
Some students may be puzzled by the oscilloscope trace. In particular, the horizontal axis appears to be a time axis, graduated in ms. It may even display negative times.

The explanation is given in the worksheet. To summarise it:

- the horizontal axis shows the time at which the sample appeared at the output;
- the buffer size (number of samples) is initially 512;
- the number of frequency bins is always half the number of samples;
- the initial sample rate is 51.2 kHz, giving a Nyquist frequency of 25.6 kHz;
- the frequency bins split the spectrum range up to the Nyquist frequency into equal values. In this case, 25.6 kHz / 256 bins gives 100 Hz range for each bin.

Typical results are shown below:





Teacher's notes

Notes

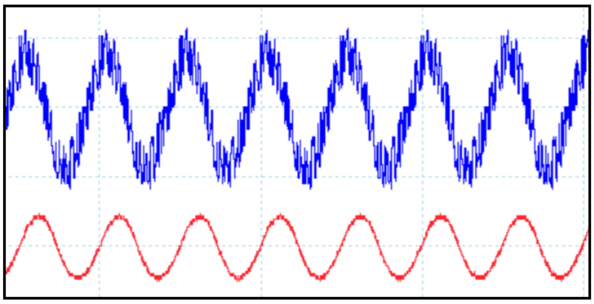
Worksheet 12
Digital filter

Using FFT reduces the mathematical depth. It ignores those frequency components that contribute little to the outcome. The result can reduce the number of computations by a factor of 10 in some cases, speeding up the process significantly.

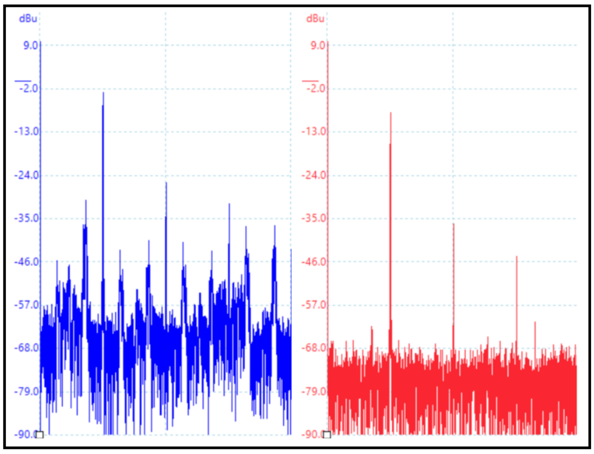
Some students may be puzzled by the oscilloscope trace. In particular, the horizontal axis appears to be a time axis, graduated in ms. It may even display negative times. The explanation is given in the worksheet. To summarise it:

- the horizontal axis shows the time at which the sample appeared at the output;
- the buffer size (number of samples) is initially 512;
- the number of frequency bins is always half the number of samples;
- the initial sample rate is 51.2 kHz, giving a Nyquist frequency of 25.6 kHz;
- the frequency bins split the spectrum range up to the Nyquist frequency into equal values. In this case, 25.6 kHz / 256 bins gives 100 Hz range for each bin.

Typical results are shown below:



Noisy sine wave (top) and result of filtering (bottom)



Version control

14 12 23 First release