# CM3106 Chapter 2: DSP, Filters and the Fourier Transform

Prof David Marshall
dave.marshall@cs.cardiff.ac.uk
and
Dr Kirill Sidorov
K.Sidorov@cs.cf.ac.uk
www.facebook.com/kirill.sidorov

School of Computer Science & Informatics
Cardiff University, UK

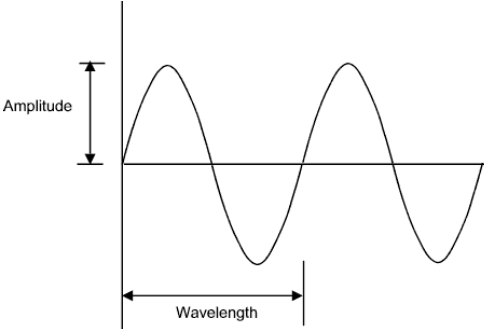
# Digital Signal Processing and Digital Audio Recap from CM2104/CM2208

## Issues to be Recapped:

- Basic Digital Signal Processing and Digital Audio
  - Waveforms and Sampling Theorem
  - Digital Audio Signal Processing
  - Filters

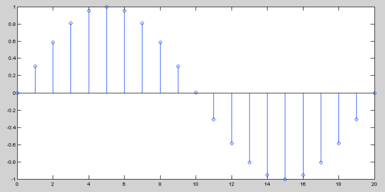
For full details please refer to last Year's **CM2208 Course Material** — **Especially detailed underpinning maths** — and also **CM2104 Notes** .

# Simple Waveforms




- **Frequency** is the number of cycles per second and is measured in Hertz (Hz)
- **Wavelength** is *inversely proportional* to frequency i.e. Wavelength varies as $\frac{1}{frequency}$

# The Sine Wave and Sound



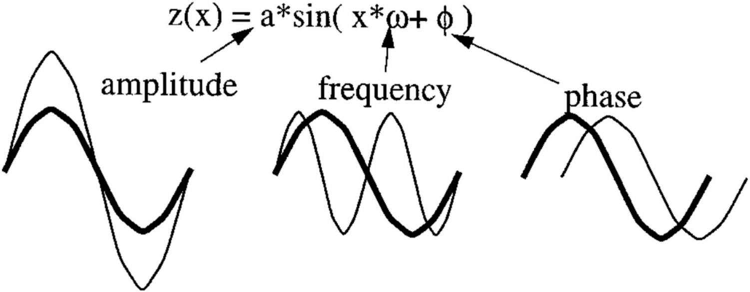

The general form of the sine wave we shall use (quite a lot of) is as follows:

$$y = A.sin(2\pi.n.F_w/F_s)$$

where:

$A$ is the amplitude of the wave,
$F_w$ is the frequency of the wave,
$F_s$ is the sample frequency,
$n$ is the sample index.

# Relationship Between Amplitude, Frequency and Phase

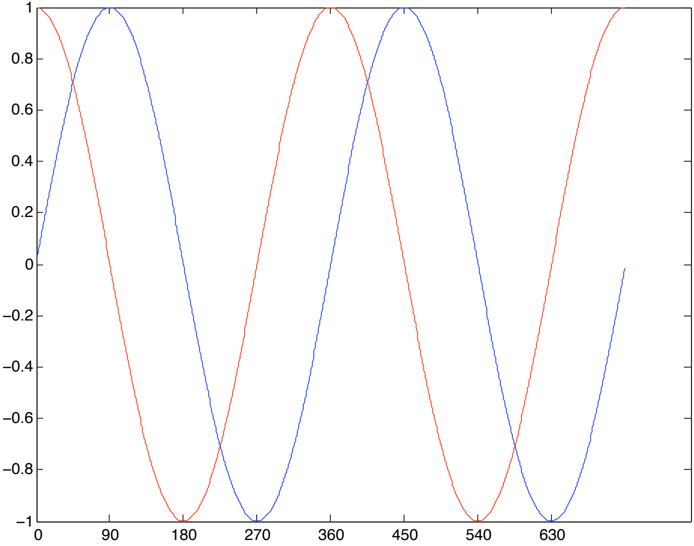


# Phase of a Sine Wave

sinphasedemo.m

```
% Simple Sin Phase Demo
samp_freq = 400;
dur = 800; % 2 seconds
amp = 1; phase = 0; freq = 1;
s1 = mysin(amp,freq,phase,dur,samp_freq);

axisx = (1:dur)*360/samp_freq; % x axis in degrees
plot(axisx,s1);
set(gca,'XTick',[0:90:axisx(end)]);

fprintf('Initial Wave: \t Amplitude = ...\n', amp, freq, phase,...);

% change amplitude
phase = input('\nEnter Phase:\n\n');

s2 = mysin(amp,freq,phase,dur,samp_freq);
hold on;
plot(axisx, s2,'r');
set(gca,'XTick',[0:90:axisx(end)]);
```

# Phase of a Sine Wave: `sinphasedemo` output



# Basic DSP Concepts and Definitions: The Decibel (dB)

When referring to measurements of power or intensity, we express these in decibels (dB):

$$X_{dB} = 10 \log_{10} \left( \frac{X}{X_0} \right)$$

where:

- $X$ is the actual value of the quantity being measured,
- $X_0$ is a specified or implied reference level,
- $X_{dB}$ is the quantity expressed in units of decibels, relative to $X_0$.
- $X$ and $X_0$ **must** have the same dimensions — they must measure the same type of quantity in the the same units.
- The reference level itself is **always at 0 dB** — as shown by setting $X = X_0$ (**note:** $\log_{10}(1) = 0$).

# Why Use Decibel Scales?

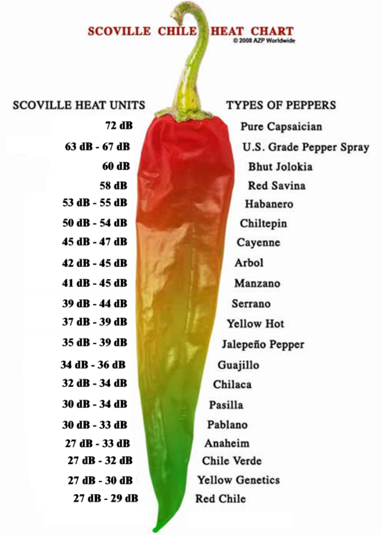
- When there is a large range in frequency or magnitude, logarithm units often used.
- If $X$ is greater than $X_0$ then $X_{dB}$ is positive (Power Increase)
- If $X$ is less than $X_0$ then $X_{dB}$ is negative (Power decrease).
- Power Magnitude $= |X(i)|^2$ so (with respect to reference level)

$$\begin{aligned} X_{dB} &= 10\log_{10}(|X(i)|^2) \\ &= 20\log_{10}(|X(i)|) \end{aligned}$$

which is an expression of dB we often come across.

# Decibel and Chillies!

Decibels are used to express wide dynamic ranges in a many applications:



# Decibel and acoustics

- dB is commonly used to quantify sound levels relative to some 0 dB reference.
- The reference level is typically set at the *threshold of human perception*
- Human ear is capable of detecting a very large range of sound pressures.

# Examples of dB measurement in Sound

## Threshold of Pain

The ratio of sound pressure that causes **permanent** damage from short exposure to the limit that (undamaged) ears can hear is above a million:

- The ratio of the maximum power to the minimum power is above one (short scale) trillion ($10^{12}$).
- The log of a trillion is 12, so this ratio represents a **difference of 120 dB**.
- **120 dB** is the quoted **Threshold of Pain** for Humans.

# Examples of dB measurement in Sound (cont.)

## Speech Sensitivity

Human ear is not equally sensitive to all the frequencies of sound within the entire spectrum:

- Maximum human sensitivity at noise levels at between 2 and 4 kHz (Speech)
  - These are factored more heavily into sound descriptions using a process called **frequency weighting**.
  - Filter (Partition) into frequency bands concentrated in this range.

  - Used for Speech Analysis
  - Mathematical Modelling of Human Hearing
  - Audio Compression (E.g. **MPEG Audio**)

**More on this Later**

# Examples of dB measurement in Sound (cont.)

## Digital Noise increases by 6dB per bit

In digital audio sample representation (**linear pulse-code modulation (PCM)**),

- The first bit (least significant bit, or LSB) produces residual quantization noise (bearing little resemblance to the source signal)
- Each subsequent bit offered by the system **doubles** the resolution, corresponding to a 6 $(= 10 * \log_{10}(4))$ dB.
- So a 16-bit (linear) audio format offers 15 bits beyond the first, for a dynamic range (between quantization noise and clipping) of (15 x 6) = 90 dB, meaning that the maximum signal is 90 dB above the theoretical peak(s) of quantisation noise.
- 8-bit linear PCM similarly gives (7 x 6) = 42 dB.
- 48 dB difference between 8- and 16-bit which is (48/6 (dB)) 8 times as noisy.

**More on this Later**

# Signal to Noise

**Signal-to-noise ratio** is a term for the power ratio between a signal (meaningful information) and the background noise:

$$SNR = \frac{P_{signal}}{P_{noise}} = \left(\frac{A_{signal}}{A_{noise}}\right)^2$$

where $P$ is average power and $A$ is RMS amplitude.

- Both signal and noise power (or amplitude) must be measured at the same or equivalent points in a system, and within the same system bandwidth.

Because many signals have a very wide dynamic range, SNRs are usually expressed in terms of the logarithmic decibel scale:

$$SNR_{dB} = 10\log_{10}\left(\frac{P_{signal}}{P_{noise}}\right) = 20\log_{10}\left(\frac{A_{signal}}{A_{noise}}\right)$$

# System Representation: Algorithms and Signal Flow Graphs

It is common to represent digital system signal processing routines as a visual **signal flow graphs**.

We use a simple *equation* relation to describe the **algorithm**.

### Three Basic Building Blocks

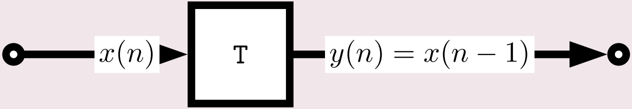
We will need to consider *three* processes:

- Delay
- Multiplication
- Summation

# Signal Flow Graphs: Delay

## Delay

- We represent a delay of **one sampling interval** by a block with a `T` **label**:




- We describe the algorithm via the equation:
$y(n) = x(n-1)$

# Signal Flow Graphs: Delay Example

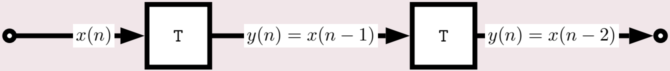
## A Delay of 2 Samples

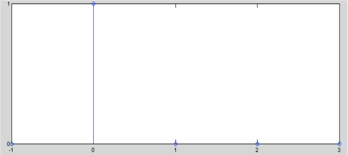
A delay of the input signal by **two** sampling intervals:

- We can describe the **algorithm** by:

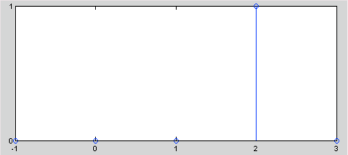
$$\mathbf{y(n) = x(n-2)}$$

- We can use the block diagram to represent the **signal flow graph** as:




$x(n)$

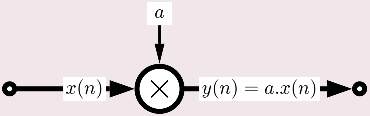



$y(n) = x(n-2)$

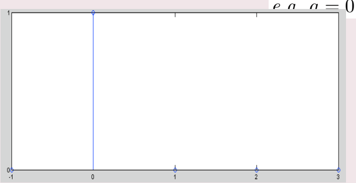
# Signal Flow Graphs: Multiplication

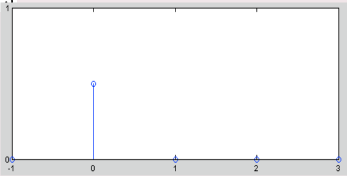
## Multiplication

- We represent a multiplication or weighting of the input signal by **a circle with a $\times$ label** .
- We describe the algorithm via the equation: $\mathbf{y(n) = a.x(n)}$




e.g. $a = 0.5$



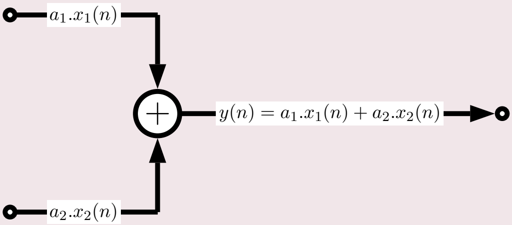

$x(n)$ $y(n) = 0.5x(n)$

# Signal Flow Graphs: Addition

## Addition

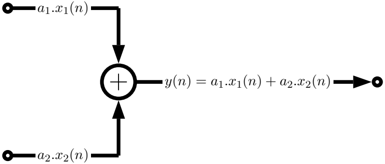
- We represent a addition of two input signal by **a circle with a + label** .
- We describe the algorithm via the equation:

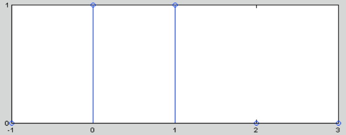
$$\mathbf{y(n) = a_1.x_1(n) + a_2.x_2(n)}$$

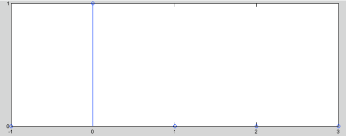


# Signal Flow Graphs: Addition Example

In the example, set $a_1 = a_2 = 1$:

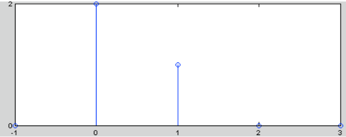


$x_1(n)$

$x_2(n)$




$y(n) = x_1(n) + x_2(n)$

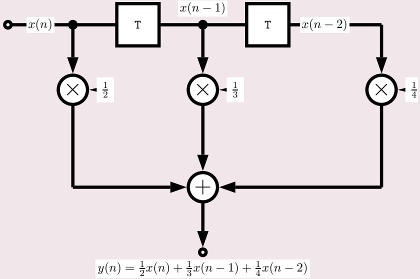
# Signal Flow Graphs: Complete Example

## All Three Processes Together

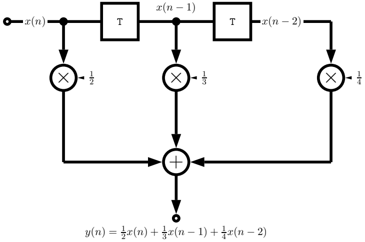
We can combine all above algorithms to build up more complex algorithms:

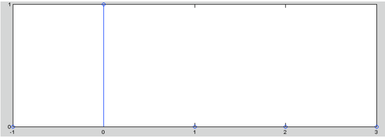
$$\mathbf{y(n) = \frac{1}{2}x(n) + \frac{1}{3}x(n-1) + \frac{1}{4}x(n-2)}$$

- This has the following signal flow graph:

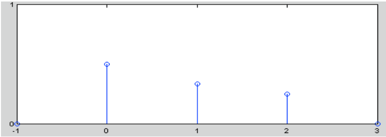


# Signal Flow Graphs: Complete Example Impulse Response



$x(n)$



$y(n) = \frac{1}{2}x(n) + \frac{1}{3}x(n-1) + \frac{1}{4}x(n-2)$