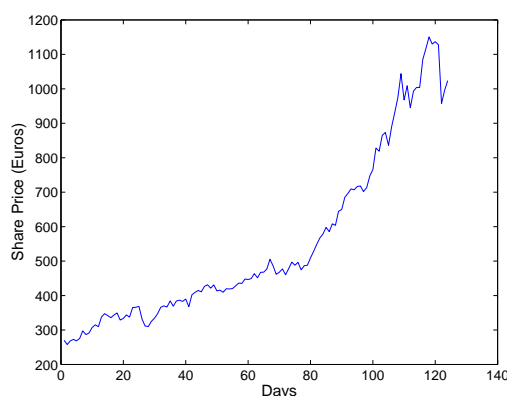
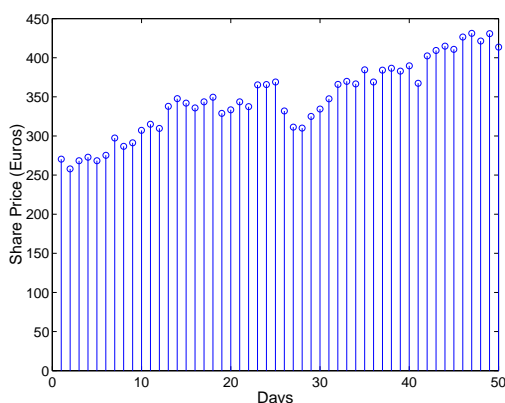


DIFFERENCE EQUATIONS AND DIGITAL FILTERS

- The last topic discussed was A-D conversion. This handout explores what becomes possible when the digital signal is processed.
- DSP (Digital Signal Processing) rose to significance in the 70's and has been increasingly important ever since. The first enabling breakthrough was the discovery of the Fast Fourier Transform in the last 1960's (enabling calculation of spectra on digital signals) and then the rise of the transistor and the CPU in the 1970's and 1980's.
- Its rapid rise to significance culminated in the advent of Digital Television/Media and Mobile Communications toward the end of the 20th century.
- The study of DSP now lies somewhere between Electronic Engineering (where it started) and Statistics, Mathematics, Computer Science and even Statistical Physics. The computer and computing science is to DSP engineers what the hammer and nail is to a carpenter.
- In the early part of the 1990's DSP chips for implementation of DSP algorithms were extremely popular: enabling faster computation times than general purpose CPUs like the Intel 808x series or Motorola, SUN or Alpha chips. But as clock speeds have increased and reconfigurable hardware has developed, the lines between DSP chips, CPUs and IC chips, have blurred.
- DSP Engineers find that because they deal with data processing and understand the subtleties of extracting useful information from poor measurements, the demand for them is always good.

1 Introducing digital filters

- A digital filter is a construct used in exactly the same way as an analogue filter.
- Consider the figure below showing the share price of Iona over a number of days. This is a digital signal.

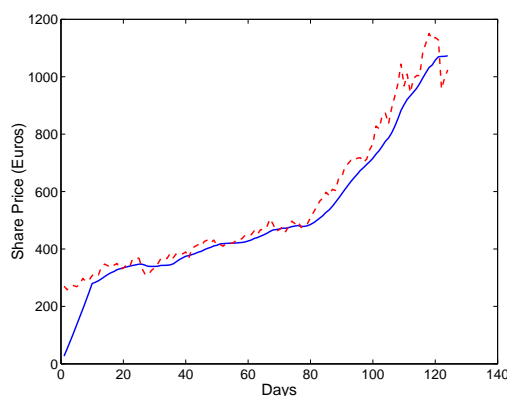
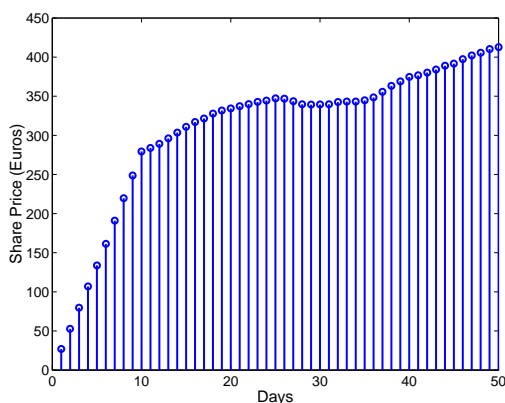


- To calculate the average share price over the last 10 days you could do something like

$$y_n = \frac{1}{10} \left[x_n + x_{n-1} + x_{n-2} + \dots + x_{n-9} \right]$$

where n is the day, x_n is the share price at day n , and y_n is the average price over 10 days.

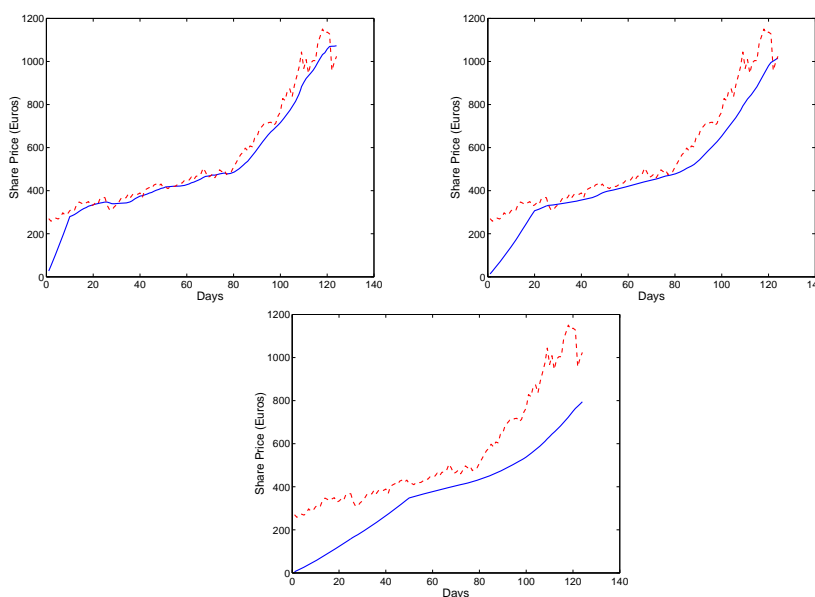
- This is a digital filter called the moving average filter. It can be applied to every day n to yield a value for the average share price over the last 10 days.
- Applying this filter to the Iona signal yields the following.



- Consider what the shape of the output signal looks like compared to the input. What is the difference? What has been removed? The signal is smoothed. The moving average filter is a low pass filter.
- What happens if we average over more days?

$$y_n = \frac{1}{N} \left[x_n + x_{n-1} + x_{n-2} + \dots + x_{n-(N-1)} \right]$$

- For $N = 10, 20, 50$ we get



- What is happening? Why?
- We can apply this filter $N = 5, 10$ to the rows of a picture to see a similar effect.



2 The Difference Equation: Impulse and Step Response

- The moving average filter is an example of a difference equation. The difference equation is to discrete signal processing what the differential equation is to analogue signal processing. This difference equation represents a Linear Time Invariant system and obeys all the usual properties.
- Given the input sequence $x_n = 270.2806, 257.9545, 268.2720, 272.8098, 0, 0$, calculate the output sequence y_n , for $n = 0 \dots 4$ when x_n is put through the filter below (assume zero initial conditions).

$$y_n = \frac{1}{3} \left[x_n + x_{n-1} + x_{n-2} \right]$$

$$\begin{aligned}
 y_0 &= \frac{1}{3} [x_0 + x_{0-1} + x_{0-2}] \\
 &= \frac{1}{3} [x_0 + x_{-1} + x_{-2}] \\
 &= \frac{1}{3} [270.2806 + 0 + 0] \\
 &= 90.09
 \end{aligned}$$

$$\begin{aligned}
 y_1 &= \frac{1}{3} [x_1 + x_{1-1} + x_{1-2}] \\
 &= \frac{1}{3} [x_1 + x_0 + x_{-1}] \\
 &= \frac{1}{3} [257.9545 + 270.2806 + 0] \\
 &= 176.08
 \end{aligned}$$

$$\begin{aligned}
 y_2 &= \frac{1}{3} [x_2 + x_{2-1} + x_{2-2}] \\
 &= \frac{1}{3} [x_2 + x_1 + x_0] \\
 &= \frac{1}{3} [268.2720 + 257.9545 + 270.2806] \\
 &= 265.50
 \end{aligned}$$

$$\begin{aligned}
 y_3 &= \frac{1}{3} [\quad \quad \quad] \\
 &= \frac{1}{3} [\quad \quad \quad] \\
 &= \frac{1}{3} [272.81 + 268.2720 + 257.9545] \\
 &= 266.34
 \end{aligned}$$

$$\begin{aligned}
 y_4 &= \frac{1}{3} [\quad \quad \quad] \\
 &= \frac{1}{3} [\quad \quad \quad] \\
 &=
 \end{aligned}$$

2.1 Impulse Response

$$y_n = \frac{1}{3} \left[x_n + x_{n-1} + x_{n-2} \right]$$

- What is the impulse response of this filter? Well, just put an impulse in and see what comes out!
- $x_n = \delta_n = 1, 0, 0, 0, 0, \dots$

$$y_0 = \frac{1}{3} \left[x_0 + x_{-1} + x_{-2} \right]$$

$$= \frac{1}{3} \left[1 + 0 + 0 \right]$$

$$= 1/3$$

$$y_1 = \frac{1}{3} \left[x_1 + x_0 + x_{-1} \right]$$

$$= 1/3$$

$$y_2 = \frac{1}{3} \left[x_2 + x_1 + x_0 \right]$$

$$= 1/3$$

$$y_3 = \frac{1}{3} \left[x_3 + x_2 + x_1 \right]$$

$$= 0$$

$$y_4 = \frac{1}{3} \left[\right]$$

$$= 0$$

So the impulse response $h_n = (1/3), (1/3), (1/3), 0, 0, 0, \dots$

- Because the impulse response has a finite duration (the non-zero values do not last forever), this kind of filter is called a Finite Impulse Response filter. (FIR Filter). Another way to remember this is that the output of the filter is a function only of past inputs.

3 FIR Filters and Impulse Response

- In the moving average FIR filter, the values that multiply the input values x_{n-1} etc, are all the same. That is the coefficients of the filter are all the same.
- We can generalise the idea of the moving average filter to a kind of moving weighted average filter. For example

$$y_n = \frac{1}{4}x_n + \frac{1}{2}x_{n-1} + \frac{1}{4}x_{n-2}$$

This can be expressed as

$$y_n = \sum_{k=0}^2 b_k x_{n-k}$$

where the filter length is 3 and coefficients are $b_k = [1, 2, 1]/4$

- In general we can write an FIR filter in that form i.e. the output is a linear combination of past inputs. Causal FIR filter:

$$y_n = \sum_{k=0}^{N-1} b_k x_{n-k}$$

- The impulse response of an FIR filter is in fact a sequence made up from its coefficients. We can see that by setting $x_n = \delta_n = 1, 0, 0, 0$ and working as follows.

$$h_0 = \sum_{k=0}^{N-1} b_k x_{0-k} = b_0 x_0 + b_1 x_{-1} + \dots = b_0$$

$$h_1 = \sum_{k=0}^{N-1} b_k x_{1-k} = b_0 x_1 + b_1 x_0 + \dots = b_1$$

$$h_2 = \sum_{k=0}^{N-1} b_k x_{2-k} = b_0 x_2 + b_1 x_1 + b_2 x_0 + b_3 x_{-1} + \dots = b_2$$

4 IIR Filters and Impulse Response

- Recursive digital filters can be designed in which the output of the filter depends both on current and previous inputs as well as previous outputs. For such filters, the impulse response has infinite duration and they are called Infinite Impulse Response (IIR) filters.
- The general form of a causal IIR filter is as follows.

$$\sum_{m=0}^{M-1} a_m y_{n-m} = \sum_{k=0}^{N-1} b_k x_{n-k}$$

- Consider calculating a moving average of a bunch of samples using $y_n = (1/3)(x_n + x_{n-1} + x_{n-2})$. This same average can be calculated by re-using previous filter outputs (averages) with a difference equation as follows

$$y_n = \frac{1}{3} \left[3y_{n-1} - x_{n-3} + x_n \right] \quad (1)$$

Thus to calculate the current average value y_n given the previous average value y_{n-1} and the new input x_n , we need to calculate the previous total, subtract the input at x_{n-3} and add in the new input x_n , before finding the new average by dividing by 3.

- This is a recursive filter. It is an IIR filter.

- Given the input sequence $x_n = 270.2806, 257.9545, 268.2720, 272.8098, 0, 0$, calculate the output sequence y_n , for $n = 0 \dots 4$ when x_n is input to the filter with difference equation as below (assume zero initial conditions).

$$y_n - y_{n-1} = \frac{1}{3}x_n - \frac{1}{3}x_{n-3}$$

Thus

$$y_n = y_{n-1} + \frac{1}{3}x_n - \frac{1}{3}x_{n-3}$$

$$y_0 = y_{-1} + \frac{1}{3}x_0 - \frac{1}{3}x_{-3}$$

$$= \frac{1}{3}270.2806$$

$$= 90.09$$

$$y_1 = y_0 + \frac{1}{3}x_1 - \frac{1}{3}x_{-2}$$

$$= 90.09 + 85.98 - 0$$

$$= 176.08$$

$$y_2 = y_1 + \frac{1}{3}x_2 - \frac{1}{3}x_{-1}$$

$$= 176.08 + 89.42 + 0$$

$$= 265.50$$

$$y_3 = y_2 + \frac{1}{3}x_3 - \frac{1}{3}x_0$$

$$= 265.50 + 90.94 - 270.28/3$$

$$= 266.34$$

$$y_4 =$$

$$=$$

$$=$$

4.1 IIR Impulse Response

- Calculate the impulse response for the system defined through the difference equation below.

$$y_n - y_{n-1} = \frac{1}{3}x_n - \frac{1}{3}x_{n-3}$$

- Put $x_n = \delta_n$ and turn the handle.

$$y_n = y_{n-1} + \frac{1}{3}x_n - \frac{1}{3}x_{n-3}$$

$$h_0 = 0 + \frac{1}{3} \times 1 - 0 = \frac{1}{3}$$

$$h_1 = \frac{1}{3} + 0 - 0 = \frac{1}{3}$$

$$h_2 = \frac{1}{3} + 0 - 0 = \frac{1}{3}$$

$$h_3 = \frac{1}{3} + 0 - \frac{1}{3} \times 1 = \frac{2}{3}$$

$$h_4 = \frac{2}{3} + 0 - 0 = \frac{2}{3}$$

$$h_4 = \frac{2}{3} + 0 - 0$$

$$h_5 = \frac{2}{3} \tag{2}$$

So the impulse response is $h_n = \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}, \frac{2}{3}, \frac{2}{3}, \dots$

- Very different from the FIR filter !
- The impulse response of an IIR filter obeys some recursive relationship
- This digital IIR filter is also LTI.

5 Step Response

- Given a difference equation, calculating the step response is done using the same idea as with the impulse response.
- Just input a step sequence and turn the handle!
- Remember discrete step function $u_n = 1, 1, 1, 1, 1, 1, 1, 1, \dots!$
- Also similar to the case with analogue signals, the step response is the running sum of the impulse response.
- The discrete step function u_n is the running sum of the discrete delta function i.e.

$$u_n = \sum_{k=0}^n \delta_k$$

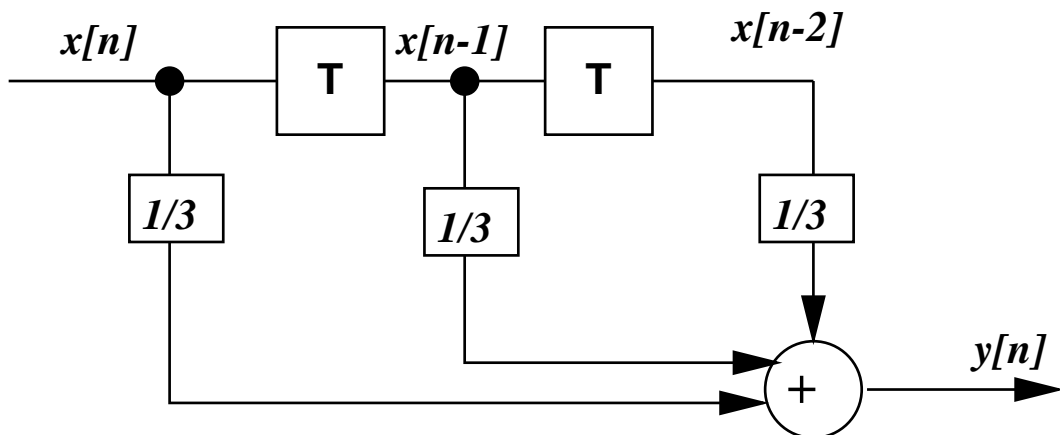
- Therefore if the running sum of a discrete delta is input into a discrete system, by the principles of LTI superposition, the output should be the running sum of the system response to a discrete delta function. (i.e. the running sum of the impulse response)

$$\text{Step response } g_n = \sum_{k=0}^n h_k$$

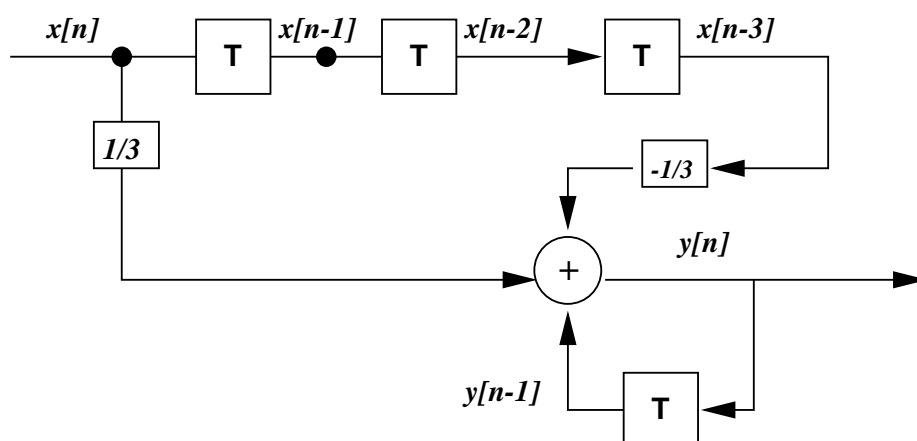
5.1 Block Diagrams for difference equations

- In the same way that block diagrams are useful for summarising the operation of analogue systems, block diagrams are useful for digital systems.
- The main, new operator needed is a *shift in time* operator. This we will denote using a box with \boxed{T} in it. Thus if x_n is put into \boxed{T} out comes x_{n-1} . \boxed{T} is a delay operator and shifts the signal that is input by 1 sample.

$$y_n = \frac{1}{3} \left[x_n + x_{n-1} + x_{n-2} \right]$$

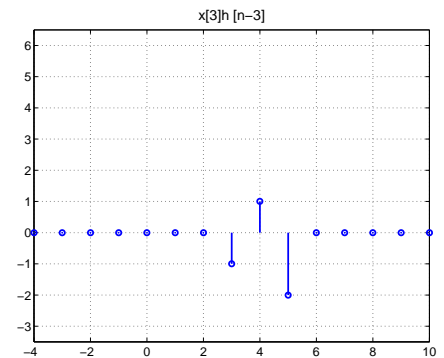
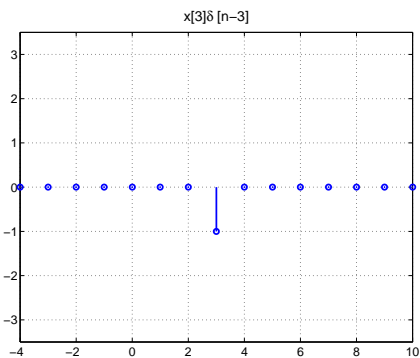
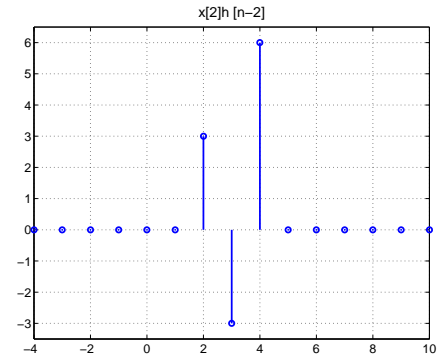
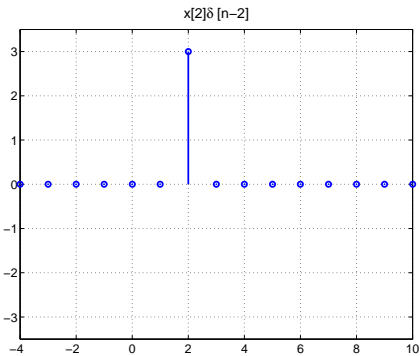
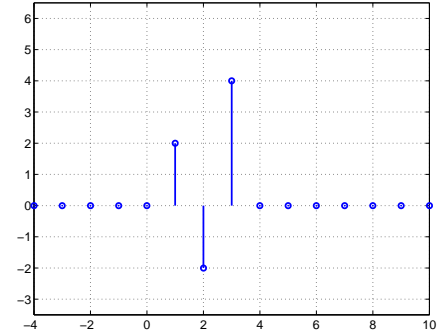
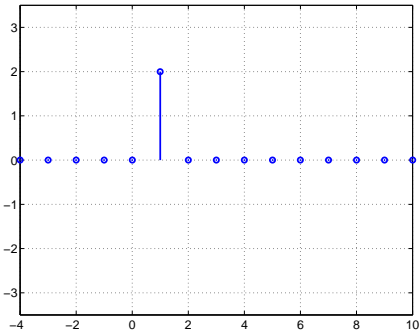
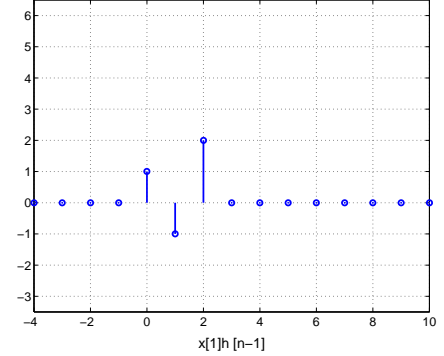
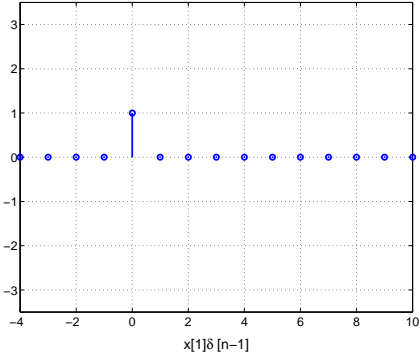
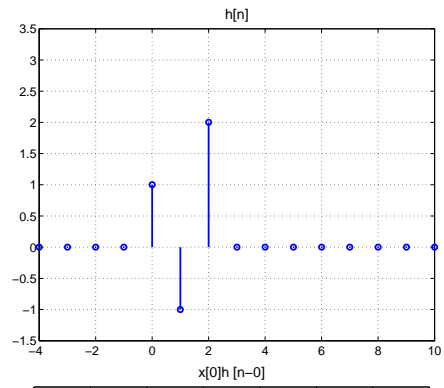
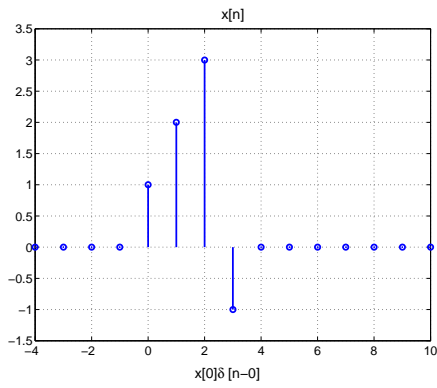


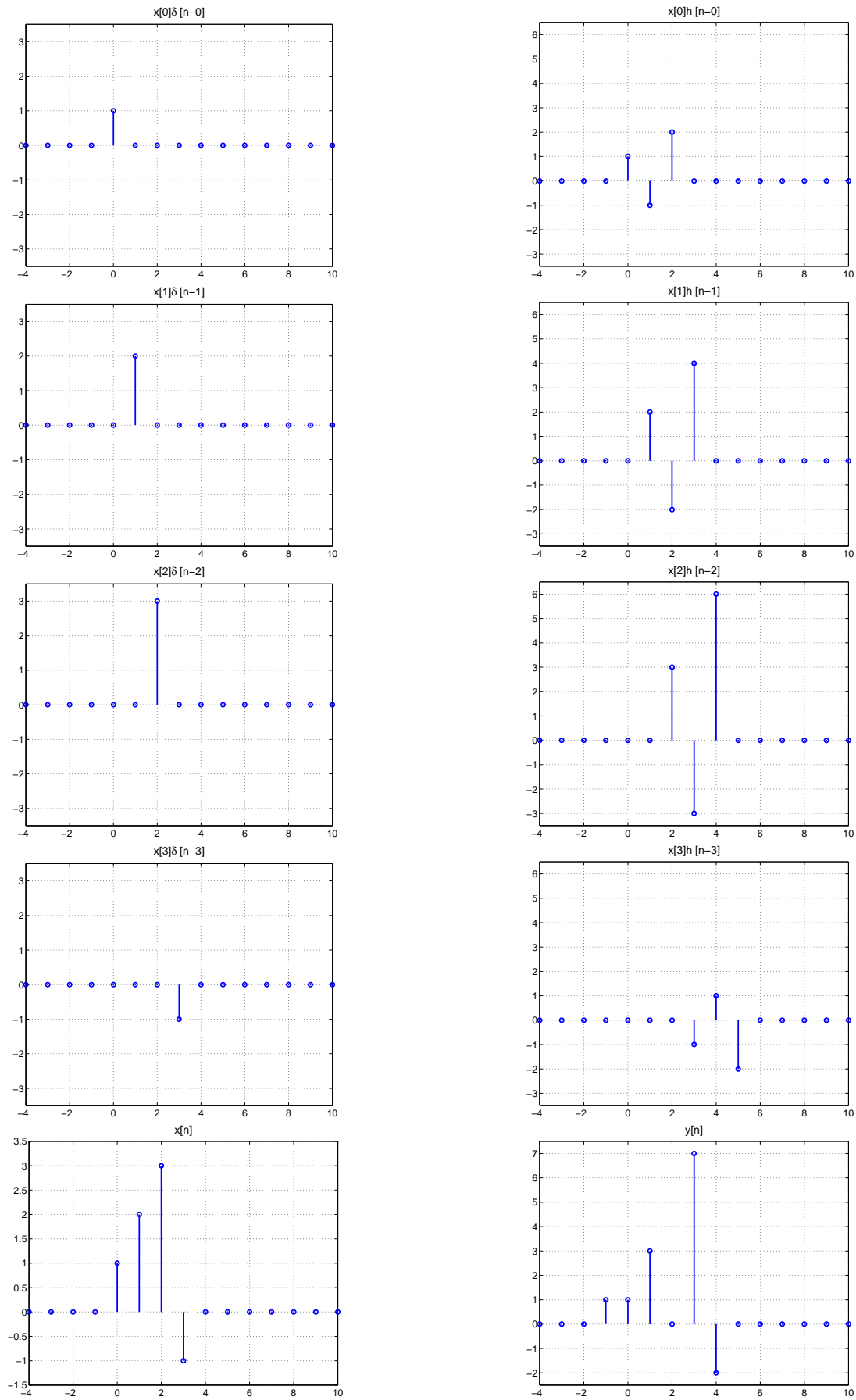
$$y_n - y_{n-1} = \frac{1}{3}x_n - \frac{1}{3}x_{n-3}$$



6 Digital Convolution

- If we know the impulse response of an LTI system, we can calculate the response to any input using convolution. We showed this from the principles of LTI systems way back when we were dealing with analogue signals and systems.
- The exact same thing happens for digital filters. But this time ... its alot more straightforward because everything is discrete anyway.
- The idea is the same. We can decompose x_n into a sum of δ_n functions. Each of these then pass through the LTI system to yield impulse responses delayed in time. The final output is the sum of these outputs.





6.1 Digital Convolution: The Expression

Given a system impulse response sequence h_n and an input sequence x_n .

- We can express any input sequence x_n as a sum of discrete delta functions

$$x_n = \sum_{k=-\infty}^{\infty} x_k \delta_{k-n}$$

Each delta function yields an impulse response as the output

INPUT \rightarrow OUTPUT

$$\delta_n \rightarrow h_n$$

$$x_0 \delta_n \rightarrow x_0 h_n$$

$$x_1 \delta_{n-1} \rightarrow x_1 h_{n-1}$$

$$x_0 \delta_n + x_1 \delta_{n-1} \rightarrow x_0 h_n + x_1 h_{n-1}$$

$$\sum_{k=-\infty}^{\infty} x_k \delta_{k-n} \rightarrow \sum_{k=-\infty}^{\infty} x_k h_{n-k}$$

HENCE THE CONVOLUTION SUM

$$y_n = \sum_{k=-\infty}^{\infty} x_k h_{n-k}$$

OR

$$y_n = \sum_{k=-\infty}^{\infty} h_k x_{n-k}$$

- Looks a lot like the statement for the FIR filter difference equation: with $b_k = h_k$. Hence the impulse response for an FIR filter is actually its coefficients.

$$y_n = \sum_{k=-\infty}^{\infty} h_k x_{n-k}$$

- Usual notation : $y_n = x_n * h_n$ Obeys all the same rules of linearity, superposition, associativity as convolution integral.

$$\begin{aligned} [x_n + g_n] * h_n &= x_n * h_n + g_n * h_n \\ x_n * h_n &= h_n * x_n \end{aligned}$$

- Usual simplifications for causality. System response h_n causal means $h_n = 0$ for $n < 0$. So we can change the lower limit to 0.

$$y_n = \sum_{k=0}^{\infty} h_k x_{n-k}$$

Signal causal means that $x_{n-k} = 0$ for $n - k < 0$ i.e. for $k > n$ so we can change the upper limit to n .

$$y_n = \sum_{k=0}^n h_k x_{n-k}$$

$$y_n = \sum_{k=0}^n h_k x_{n-k}$$

- FIR time domain filter implementation is the same as convolution. When people talk about FIR filter coefficients and FIR impulse response ... its in fact the same thing.