EEN1034 - Digital Signal Processing (Digital Filters and DFT)

Dr. Marissa Condon

Acknowledgment
The notes are adapted from those given by
Dr. Dushyantha Basnayaka

Learning Outcomes

Learning Outcomes from the entire course

- Digital signals and their fundamental properties
- Systematic design and analysis of digital systems and filters
- Performance evaluation of digital filters
- Real-time implementation of DSP systems

Assessments

Assessment format of EEN1034 is:

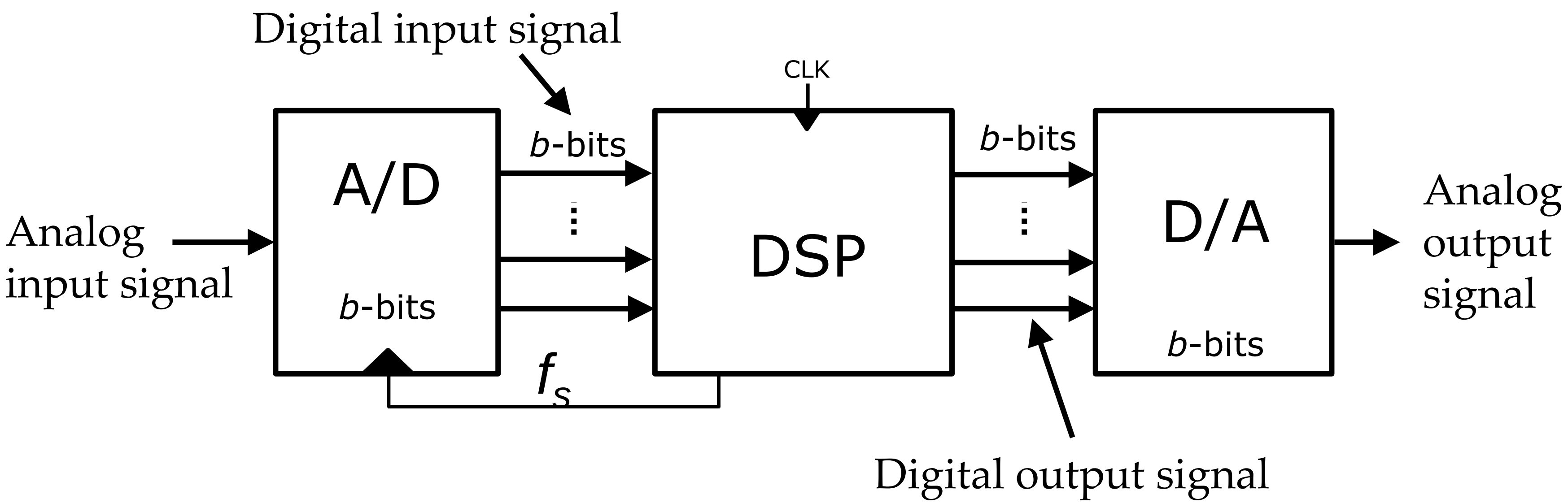
a) CA (20%)

- 2 Supervised Loop Quizzes

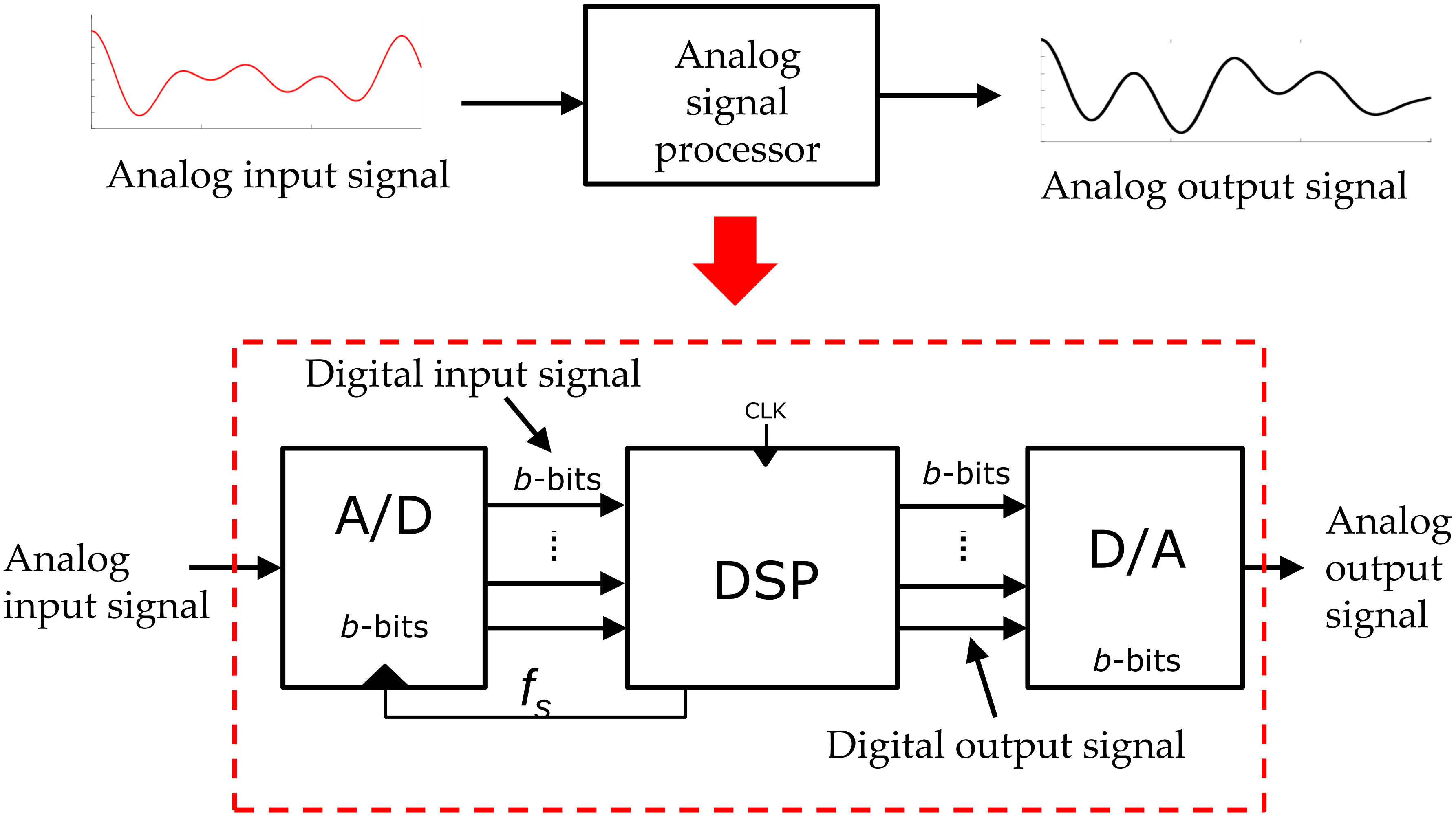
b) Exam (80%)

Digital Signal Processing

- Analog signals cannot be directly fed into digital signal processors.
- The signal conditioning operation employed prior to feeding into DSP is commonly known as A/D conversion, which includes many sub-operations.

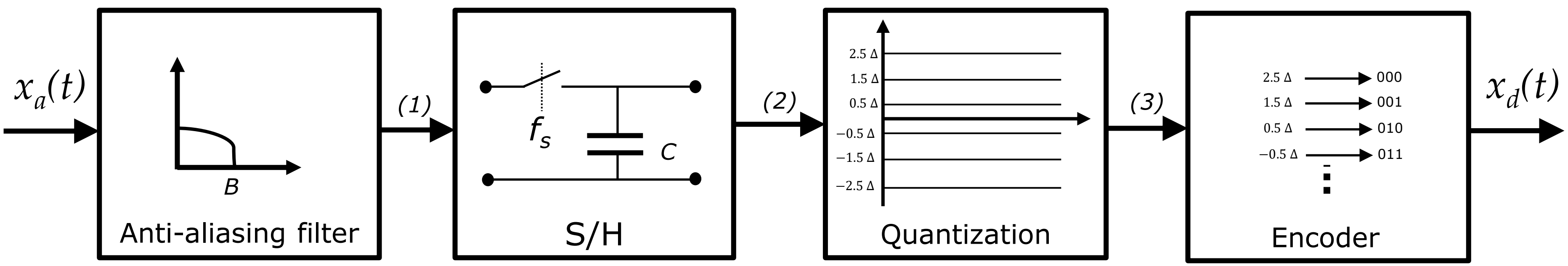


Introduction



A/D Conversion

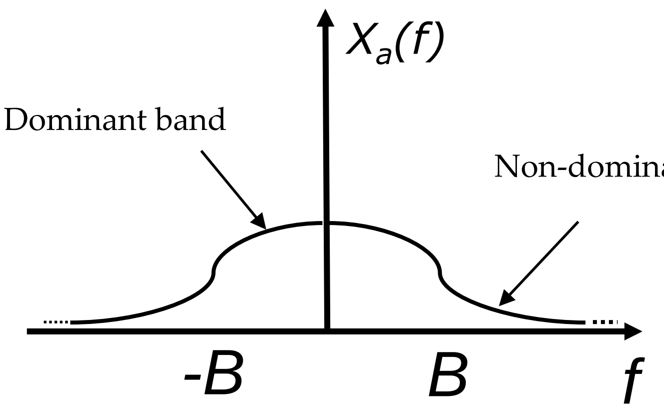
- The major sub-operations of A/D conversion includes low pass filtering (anti-aliasing filtering), sample-and-hold, quantization and encoding.

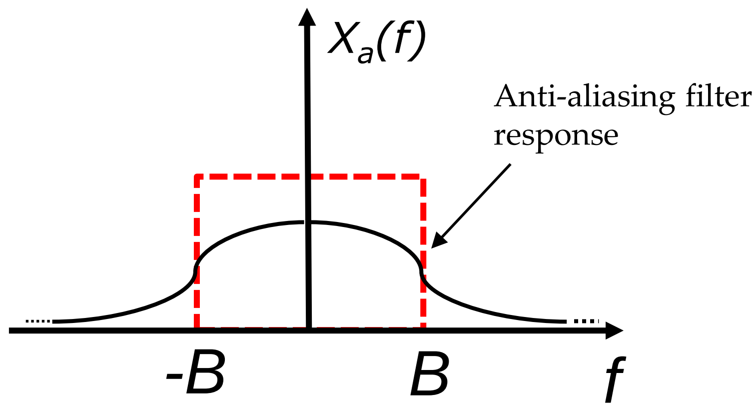


$x_a(t)$ - Analog signal, (1) - Band-limited analog signal, (2) - Stair-case analog signal, (3) – Quantized Discrete-Time signal, and $x_d(t)$ - Digital signal with 1s and 0s.

A/D Conversion – Anti-Aliasing

- Real analog signals often have a dominant and non-dominant frequency band. The non-dominant band could occupy several times larger a band than the dominant band.

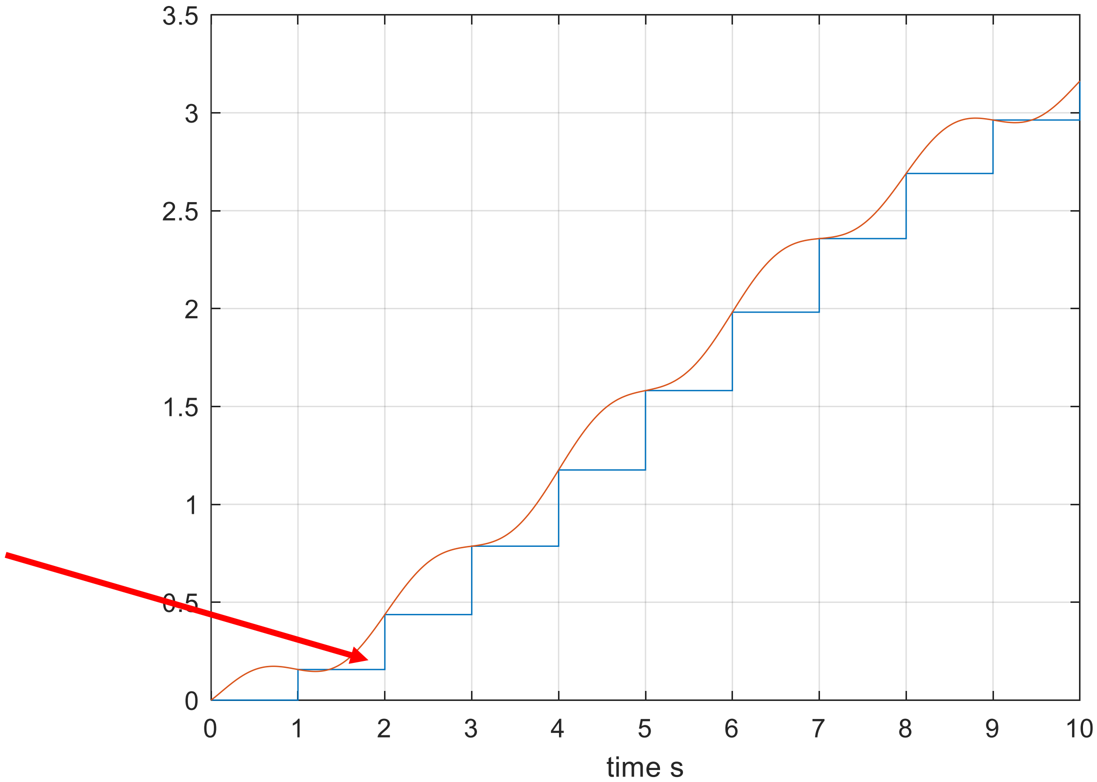




A low-pass filter known as an anti-aliasing filter must be used prior to sampling so that the power in the non-dominant band is significantly reduced.

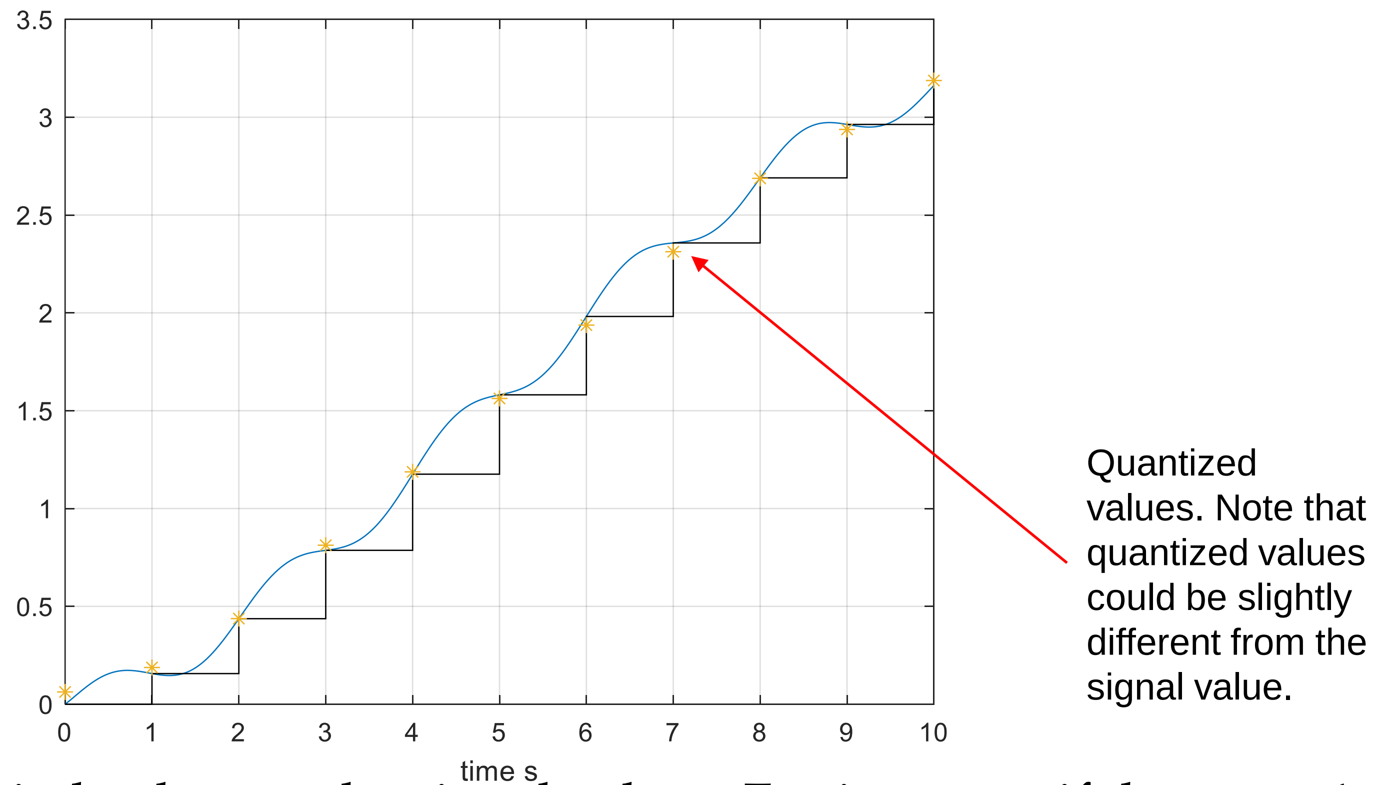
A/D Conversion – Sample and Hold

Stair-case
analog
signal



Graphical representation of A/D conversion in DSP.
The signal is sampled and held at that value until the next sampling instant

A/D Conversion - Encoding



One assigns digital values to the signal values. For instance, if there are 4 signal values, one can assign “00”, “01”, “10”, “11” to them. This is called encoding.

Notational Convention

- Analog signal: $x_a(t)$

- Discrete-time (DT) signal: DT signals are always associated with a sampling interval, T_s. One can use [] or () brackets

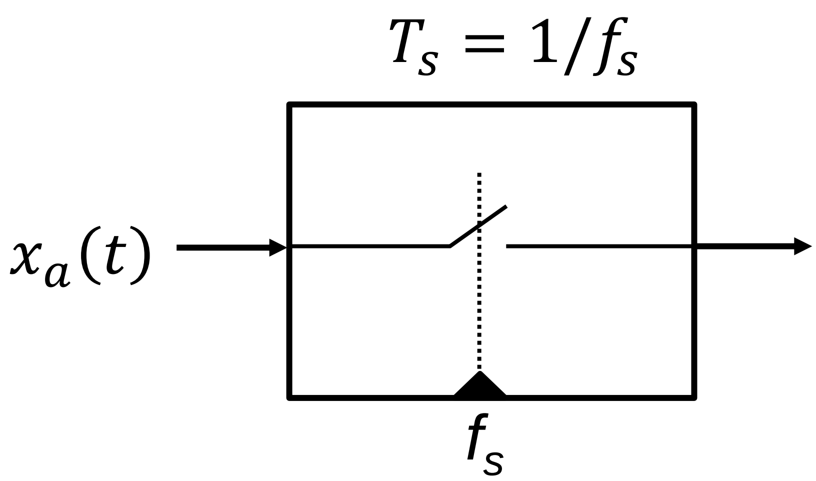
$$x[n] = x(n) = x_a(nT_s)$$

- Quantized discrete-time signal: $x_q[n] = Q(x[n])$

- Digital signal: This is a signal with "1" and "0".

$$x_d(t)$$

Mathematical Basics for DSP



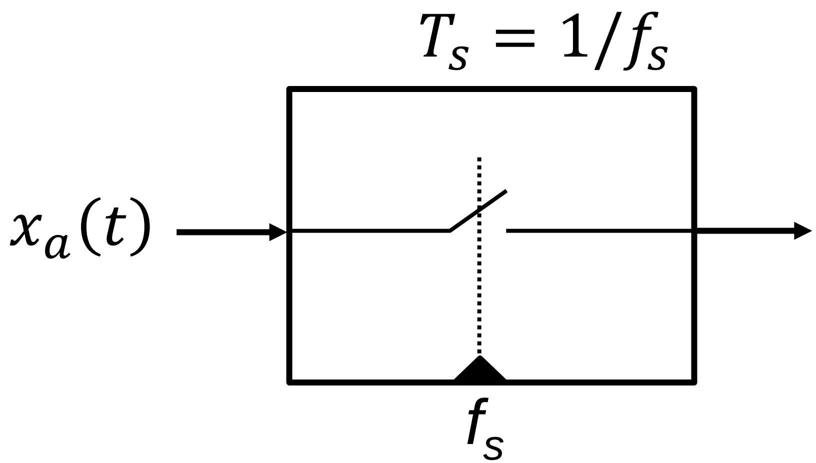
$$x[n] = x_a(t)\left[\sum_{n=-\infty}^{\infty} \delta(t - nT_s)\right]$$

Fig: A model for ideal sampling with an ideal switch.

Time shifted Dirac delta function

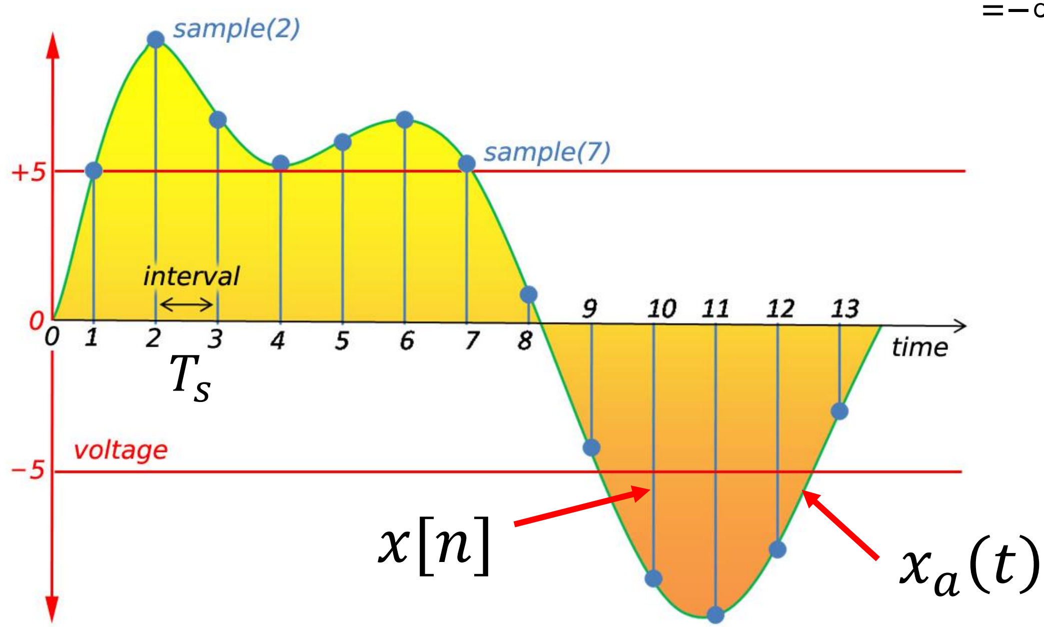
It is assumed that the ideal switch can switch on and off in zero time giving rise to the notion of a train of impulses. The output signal, x[n], is discrete in time but continuous in amplitude.

Mathematical Basics for DSP



$$x[n] = x_a(t)\left[\sum_{n=-\infty}^{\infty} \delta(t - nT_s)\right]$$

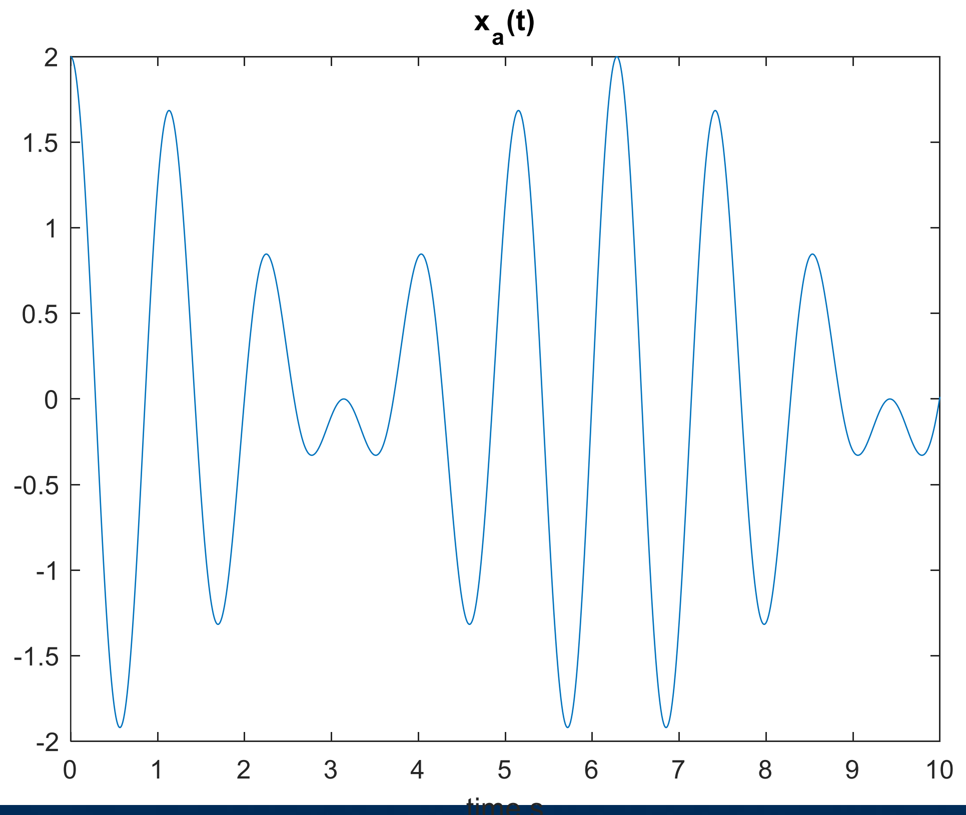
$$= \sum_{=-\infty}^{\infty} x_a(nT_s)\,\delta(t - nT_s)$$



Basic MATLAB for DSP

- How to plot an analog signal in MATLAB?

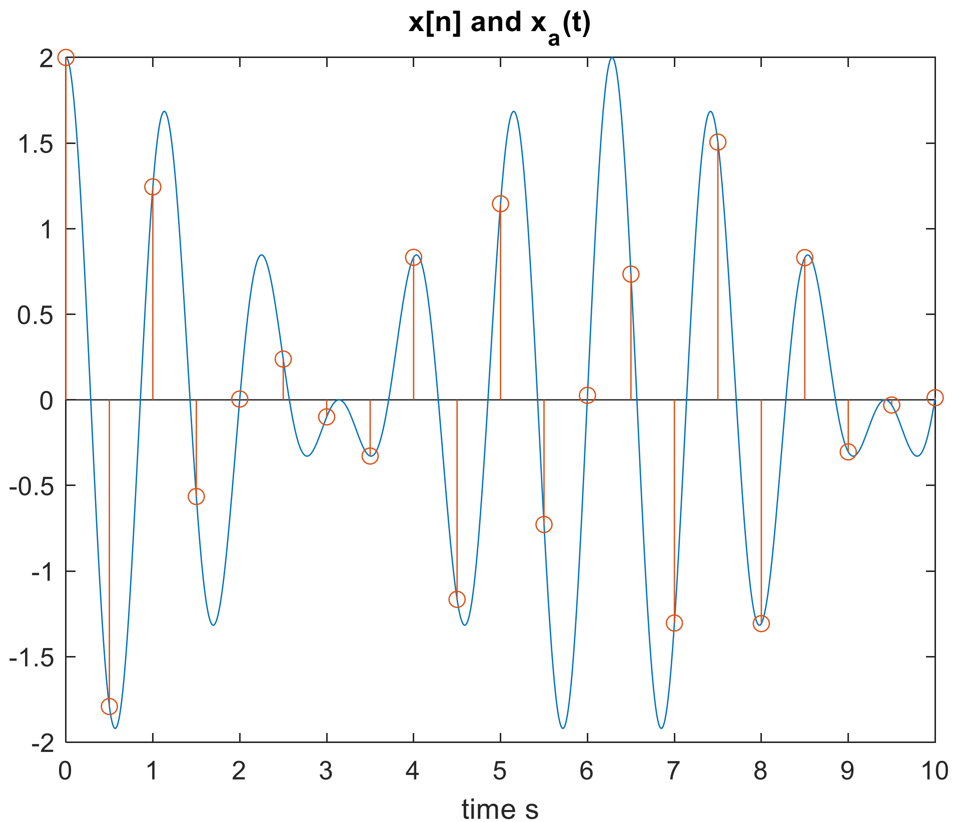
Consider $x_a(t) = \cos(5t) + \cos(6t)$.




```
clear
t=0:1/100:10;
xa=cos(5*t)+cos(6*t);
plot(t,xa)
xlabel('time s')
title('x_{a}(t)')
```

Basic MATLAB for DSP

- Consider $x_a(t) = \cos(5t) + \cos(6t)$.
- Suppose one samples this signal with a sampling interval of 0.5s



Example code in MATLAB

```
clear
t=0:1/100:10;
xa=cos(5*t)+cos(6*t);
figure(1)
plot(t,xa,'b')
xlabel('time s')
title('x_{a}(t)')


sample_time=0.5;
n=0:20;
xn=cos(5*n*sample_time)+cos(6*n*sample_time);
hold on
stem(sample_time,xn,'r')
xlabel('time s')
title('x[n] and x_{a}(t)')
```

Signal Quantization

$$x_q[n] = Q(x[n])$$

There are many types of quantization, but here consider "Rounding".
Let the quantization levels be:

$$\{\dots, -2.5\Delta, -1.5\Delta, -0.5\Delta, 0.5\Delta, 1.5\Delta, 2.5\Delta, \dots\}$$

where Δ is the quantization step size. In rounding, $x_q[n]$ is the closest quantization level to $x[n]$. The quantized sequence can be given by:

$$x_q[n] = x[n] + e_q[n]$$

where $e_q[n]$ is defined as the quantization error sequence, and $-0.5\Delta \leq e_q[n] \leq 0.5\Delta$.

Example

$$\Delta = 0.1$$

The levels are

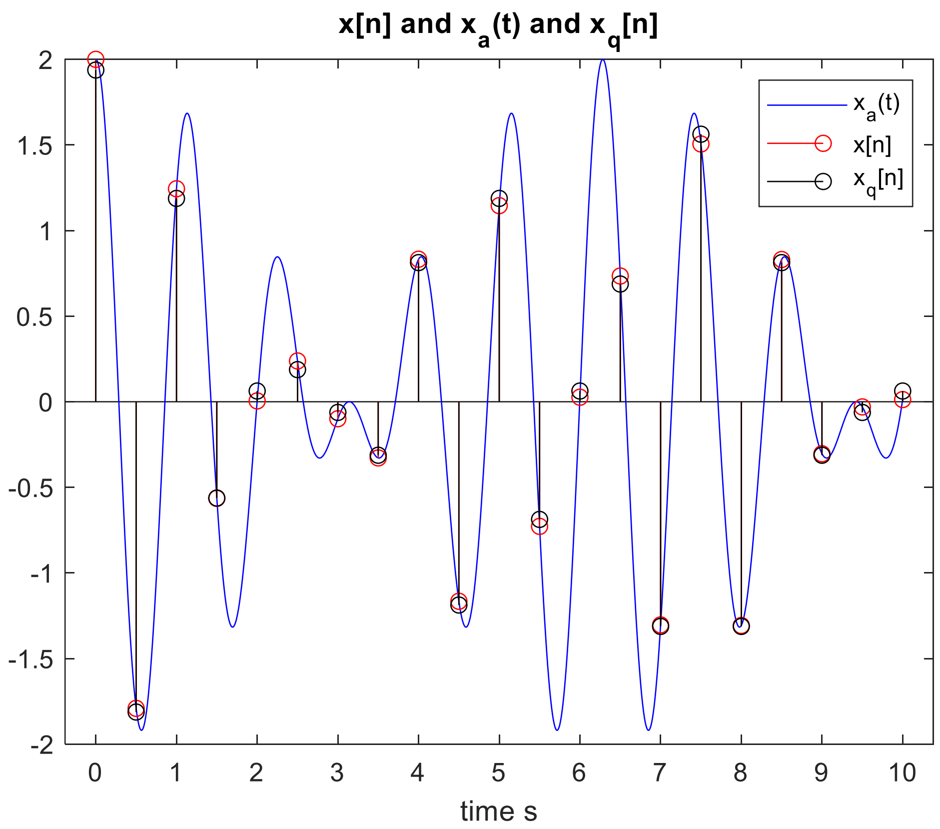
$$0.05, 0.15, 0.25, 0.35\ 0.45\ \ldots$$

So if x[n]=0.763, the quantization levels nearest to x[n] are 0.75 and 0.85.

$$x_q[n] = 0.75$$

Quantization in MATLAB

- How to quantize $x_a(t) = \cos(5t) + \cos(6t)$ in MATLAB.



Note the difference between $x[n]$ and $x_q[n]$. It reduces as the number of quantization levels increases.

```
xmax=ceil(max(xn)); %round the max to the nearest integer above
xmin=floor(min(xn)); %round the min to the nearest integer below


Sigma=(xmax-xmin)/32; %set the quantization step size
q1=xmin+Sigma/2:Sigma:xmax;%set the quantization level vector
for k=1:length(xn)
xt=xn(k)*ones(1,length(q1));
error=abs(xt-q1);
[aa,eindex]=min(error);
xq(k)=q1(eindex); %quantized value for xn
end


hold on
stem(n*sample_time,xq,'k')
legend('x_{a}(t)','x[n]','xq[n]')
```