Theoretical background

1. Depth of storage:

Sampling rate f_s – sample per second

Time base T_B – how many second per division. Shorter time base settings (faster display of the signal on the screen) typically lead to a higher sampling rate because the oscilloscope must sample more frequently to capture the waveform accurately within the shorter time frame.

N_x: number of divisions on the screen in the x axis

N_y: number of divisions on the screen in the y axis

The depth of the storage can be calculated as:

$$DST = f_s \times TB \times n_x$$

Nyquist Criterion: To avoid aliasing, the sampling rate must be at least twice the maximum frequency present in the signal. This ensures that the waveform is accurately represented without "overlapping" frequencies, which can cause distortion.

2. Resolution if the analogue to digital converter

ADC : $2^n = 256\ quantization\ levelsor\ incrment$

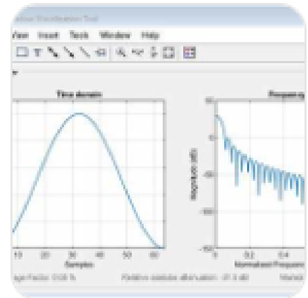
Hence, if the voltage sensitivity is set too low and the signal is represented by only a few divisions, only a part of the 255 increments is used to represent the signal. In this way we will loose resolution. To choose an appropriate setting of the voltage sensitivity we have to make sure that the signal uses as much of the screen as possible without being cut off by the limits of the screen.

3. Leakage effect

The frequency components in a signal is infinitive however, the windows of the machine DSO is limted so this lead to the loss of some ordinary frequency -> leakage effect

A rectangular time window leads to a significant leakage.

Other window functions with a slow rise and decay of the signal amplitude can reduce the leakage effect. An example for such a window function is the "von Hann" window which will be used in our experiment.



The **von Hann window**, also known as the **Hanning window**, is a function used in digital signal processing to reduce spectral leakage during Fourier analysis. It smoothly tapers the edges of a signal segment to zero, minimizing discontinuities at the boundaries.

Mathematically, the Hanning window is defined as:

$$w(n) = 0.5\left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right)$$

where:

- n is the sample index (ranging from 0 to $N-1$),
- N is the total number of samples in the window.

The leakage effect can be avoided when the window length is an integer multiple of the period of the signal even when a rectangular time window is used, compare fig. 1.2.

Prepare for the experiment

1. Experiment 1
    a. Purpose

To understand how depth of storage (DST) works on a DSO and

calculate it based on the DSO settings.

To observe aliasing

    b. Steps

Disable the Anti-Aliasing Option on the DSO:

Set Up a Sine Wave Signal:

- Use a function generator to generate a sine wave signal (suggested parameters: $f = 250\,\mathrm{kHz}$, $U_0 = 1\,\mathrm{VPP}$).
- Apply this signal to **Channel 1** of the DSO.

Adjust Time Base Settings:

- Choose two different time base settings for the DSO, such as:
    - $\mathrm{TB1} = 200\,\mathrm{ms/DIV}$ (a slow time base setting)
    - $\mathrm{TB2} = 1\,\mathrm{s/DIV}$ (another slow time base setting)
- These settings affect the sampling rate, as the DSO adjusts the sampling rate based on the time base.

Check the Sampling Rate on the DSO:

- On the DSO model (Agilent DSO5014), press the “Main/Delayed” button to view the sampling rate,
- Observe how the DSO’s sampling rate changes with different time base settings.

Calculate Depth of Storage (DST): $\mathrm{DST} = f_s \times (\mathrm{TB} \times n_x)$

Consider Aliasing:

2. Experiment 2
    a. Purpose of Experiment 2

The main goals of Experiment 2 are:

To determine the resolution of the DSO's ADC by observing how it quantizes the input signal.

To understand how voltage settings affect ADC resolution, particularly how different settings can either improve or reduce the ability to accurately represent the signal.

    b. Steps Involved in Experiment 2

Set Up a Low-Amplitude Sine Wave Signal:

- Use a function generator to generate a sine wave with a **small amplitude** (e.g., $U_0 = 0.2\,\mathrm{VPP}$).
- Apply this signal to **Channel 1** of the DSO.

Choose Appropriate Time Base Setting: - no aliasing

Use Two Different Voltage Settings:

- Select two different voltage sensitivity settings on the DSO, for example:
  - **Voltage Setting 1**: $100\,\mathrm{mV/DIV}$
  - **Voltage Setting 2**: $5\,\mathrm{V/DIV}$
- These settings determine the range of the vertical scale and impact the ADC's effective resolution for capturing the signal.

Calculate the ADC Resolution:

- The DSO's ADC has **8-bit resolution**, giving it $2^8 = 256$ quantization levels.
- To calculate the effective **voltage resolution** (smallest detectable change in voltage) for each voltage setting:

$$\text{Voltage Resolution} = \frac{\text{Full-Scale Voltage Range}}{\text{Number of Quantization Levels}}$$

- For each voltage setting, use the number of vertical divisions $n_y = 8$ and the chosen voltage sensitivity (e.g., $100\,\mathrm{mV/DIV}$ and $5\,\mathrm{V/DIV}$) to find the voltage resolution for each case.

3. Experiment 3
   a. Purpose of Experiment 3

The main goals of Experiment 3 are:

To determine the frequency resolution of the FFT and observe how different time base settings affect it.

To study the leakage effect when analyzing a sine wave and learn how using a von Hann window can reduce this effect for clearer frequency analysis.

   b. Steps Involved in Experiment 3

Part A: Determining Frequency Resolution Without Leakage

Set Up a Sine Wave Signal:

- Use a function generator to generate a sine wave signal with a frequency that will **not cause leakage** or **aliasing**. This is done by selecting a frequency such that the displayed time window on the DSO screen is an integer multiple of the signal period.
- Suggested settings: $f = 100\,\mathrm{kHz}$.

Choose Two Different Time Base Settings:

- Use the DSO to capture the sine wave with two different time base settings:
  - **Time Base 1 (TB1)**: A short time base setting, such as $5\ \mu\mathrm{s}/\mathrm{DIV}$.
  - **Time Base 2 (TB2)**: A long time base setting, such as $100\ \mu\mathrm{s}/\mathrm{DIV}$.
- These settings will provide different numbers of samples, affecting the frequency resolution of the FFT.

Analyze the Frequency Resolution Using MATLAB:

- Read the signal into MATLAB using the **mess7** function, and analyze it with the **spec_1** function to perform an FFT.
- Determine the **frequency of the applied sine wave** and **frequency resolution** of the FFT spectrum for each time base setting.
- The **frequency resolution** improves (i.e., gets finer) with a longer time base setting because more samples are captured, allowing for a clearer view of the frequency components.

Insert FFT Spectrum into the Lab Report:

- Save the FFT spectrum generated in MATLAB for each time base setting, as this will show how different settings affect the frequency resolution.

Part B: Studying the Influence of the von Hann Window

Apply the von Hann Window:

- Using the sine wave signal captured with the **high time base setting** from Part A (e.g., $\mathrm{TB} = 100\ \mu\mathrm{s}/\mathrm{DIV}$), apply a **von Hann window** in MATLAB using mess7.
- The von Hann window tapers the edges of the signal, reducing spectral leakage in the frequency domain.

Observe and Describe Changes in the FFT Spectrum:

- Compare the FFT spectrum obtained with the von Hann window to the spectrum without the window function from Part A.
- Document how the von Hann window affects the **sharpness of the frequency peaks** and reduces **side lobes** (unwanted frequencies), providing a clearer view of the main frequency component.

4. Experiment 4

The main goals of Experiment 4 are:

1. **To observe the leakage effect** in the frequency spectrum when the time window is not an integer multiple of the signal period.
2. **To study how the von Hann window can minimize leakage**, improving the clarity of the frequency spectrum.

Set Up a Sine Wave with Slightly Off Frequency:

- Use a function generator to create a sine wave with a frequency close to, but not exactly the same as, the frequency used in **Experiment 3**.
- This small difference in frequency will ensure that the displayed time window on the DSO screen is no longer an integer multiple of the sine wave's period, which will cause leakage in the frequency spectrum.
- Suggested settings: frequency period $T = 10.6\,\mu\mathrm{s}$ and time base setting $\mathrm{TB} = 100\,\mu\mathrm{s}/\mathrm{DIV}$.

Capture the Signal on the DSO:

- Set the DSO time base to provide a good frequency resolution, as done in Experiment 3 (e.g., $\mathrm{TB} = 100\,\mu\mathrm{s}/\mathrm{DIV}$).
- Ensure the sine wave appears stable on the screen, and capture it using the DSO.

Read the Signal into MATLAB and Perform FFT:

- Read the signal into MATLAB with the **mess7** function and perform an FFT on the signal with a **rectangular window** (no windowing function applied).
- Observe and record the frequency spectrum, focusing on the **leakage effect**, which will show up as additional unwanted frequencies or side lobes around the main frequency component.

Analyze the Effect of the von Hann Window on the Spectrum:

- Apply a **von Hann window** to the captured signal in MATLAB.
- Perform the FFT again on the windowed signal and observe the frequency spectrum.
- The von Hann window should reduce the leakage effect by tapering the edges of the signal, resulting in a clearer and more defined main frequency peak with reduced side lobes.

5. Experiment 5

The main goals of Experiment 5 are:

1. **To measure the phase shift introduced by a low-pass filter** at its cut-off frequency.
2. **To compare the filter's effect on a sine wave and a rectangular signal**, observing the phase response for different frequency components of the rectangular signal.

1. **Design and Set Up the Low-Pass Filter**:
   - Create a simple **RC low-pass filter** using a resistor R and capacitor C.
   - The cut-off frequency f_{co} for an RC low-pass filter is calculated as:

$$f_{\text{co}} = \frac{1}{2\pi RC}$$

   - Use a resistor value of $R = 1\,\text{k}\Omega$ and calculate the required capacitance C to achieve a cut-off frequency of 1 kHz.
2. **Set Up a Sine Wave at the Cut-Off Frequency**:
   - Generate a sine wave from the function generator with a frequency equal to the cut-off frequency, $f = 1\,\text{kHz}$, and apply it as the input to the low-pass filter.
   - Use the DSO to monitor both the input and output signals of the filter on **two separate channels**.
3. **Measure the Phase Shift**:
   - Using the **FFT** function on the DSO or in MATLAB (via the "Spec + phasediff" function in **mess7**), measure the **phase difference** between the input and output signals.
   - The phase shift at the cut-off frequency should be close to -45°, which is typical for an RC low-pass filter at its cut-off.

Part B: Phase Shift and Frequency Response for a Rectangular Signal

1. **Apply a Rectangular Signal**:
    - Generate a rectangular signal from the function generator, applying it to the low-pass filter.
    - A rectangular wave contains a fundamental frequency and several harmonics (odd multiples of the fundamental frequency), so it provides a more complex input for observing the frequency response of the filter.
2. **Measure Frequency Components in the FFT Spectrum**:
    - Use the FFT spectrum to identify the **three largest frequency components** of the rectangular signal.
    - Observe how the low-pass filter attenuates these components based on their frequency relative to the cut-off frequency.
3. **Measure the Phase Shift of Each Component**:
    - For each of the three largest frequency components, measure the **phase shift** using the FFT phase spectrum.
    - Record how the phase shift varies for the fundamental frequency and its harmonics, as higher frequencies should experience more phase shift and attenuation due to the low-pass filtering effect.