

# DIGITAL SIGNAL GENERATOR

**Team Members :-**

**Prakash Palsaniya (2023BITE055)**

**Pranav Abani (2023BITE078)**

**Kanhaiya Yadav(2023BITE076)**

**Subject:-** Data Communication

**GitHub:-** [Repo](#)

**Date :-** 31/10/2025

## 1. Programming Language

**C++ (ISO C++11 Standard)** – Ensures portability, reliability, and efficient handling of numerical and graphical operations.

## 2. Libraries Used

**OpenGL:** Core graphics library for rendering and real-time signal visualization.

**GLUT/FreeGLUT:** Window management, event handling, and interactive visualization.

**Standard C++ Libraries:** `<iostream>` (console I/O), `<cstring>` (string handling and binary data manipulation), `<cmath>` (mathematical functions for sine, cosine, and quantization).

## 3. Key Assumptions

- Input binary data is valid and contains only '0' and '1' characters.
- **PCM:** Analog samples are real-valued; quantization levels =  $2^{\text{bits}}$ .
- **Delta Modulation:** Step size ( $\Delta$ ) = 0.5; initial predicted value = 0.
- **Bipolar Schemes:** B8ZS operates on AMI-encoded signals with 8 consecutive zeros; HDB3 with 4 consecutive zeros.
- Display window assumption: Visualization assumes a minimum resolution of 800x600 pixels for correct signal rendering.

## 4. Compilation and Execution (Windows)

▶ Download and install the FreeGLUT and MinGW (Windows GCC) development environment before compilation.

▶ Setup Freeglut Files - Check [Github README](#)

Compilation:

```
g++ signal_generator.cpp -o signal_generator.exe -L. -l. -lfreeglut -lopengl32 -lglu32
```

Execution:

```
./signal_generator.exe
```

## 5. References

1. "Data Communication Class Notes - **Iqra Altaf Gillani** (Line Coding and Scrambling Techniques)
2. GLUT Documentation: [freeglut.sourceforge.net](http://freeglut.sourceforge.net)