



Time Division Multiplexing, Signal Reconstruction using Samples and Delta Modulation

Lab Exercises on June 23, 2021

Department of Electronics and Computer Engineering

Pulchowk Campus, Lalitpur

Ashlesh Pandey

PUL074BEX007

Contents

List of Figures	ii
Listings	iii
1 Objectives	1
2 Background Theory	1
2.1 Time Division Multiplexing (TDM)	1
2.2 Sampling and signal reconstruction using samples	2
2.3 Delta modulation	3
3 Exercises	4
3.1 Time division multiplexing of two signals and recovery of signals	4
3.2 Reconstruction of a signal from its samples	5
3.3 Delta modulation and bit sequence for delta modulation transmission	7
4 Discussion and Conclusion	9

List of Figures

1	Block diagram of TDM	1
2	Block diagram of sampling and reconstruction using samples	3
3	Block diagram of delta modulation at transmitter	3
4	Time division multiplexing of two signals and recovered signals	5
5	Reconstruction of a signal from its samples	6
6	Delta modulation and bit sequence for delta modulation transmission	9

Listings

1	Matlab script for time division multiplexing and recovery of original signals	5
2	Matlab script for sampling and reconstruction of a signal from its samples .	6
3	Matlab script for delta modulation and bit sequence for delta modulation transmission	8

1 Objectives

- Perform time division multiplexing of two signals and recover the original signals.
- Perform reconstruction of a signal from its samples.
- Perform delta modulation and visualize the bit sequence for DM transmission.

2 Background Theory

2.1 Time Division Multiplexing (TDM)

TDM is the time interleaving of samples from different independent sources so that the information is transmitted serially over a single communication channel. This implementation of multiplexing enables the joint utilization of a common channel by a number of independent message signals without any interference.

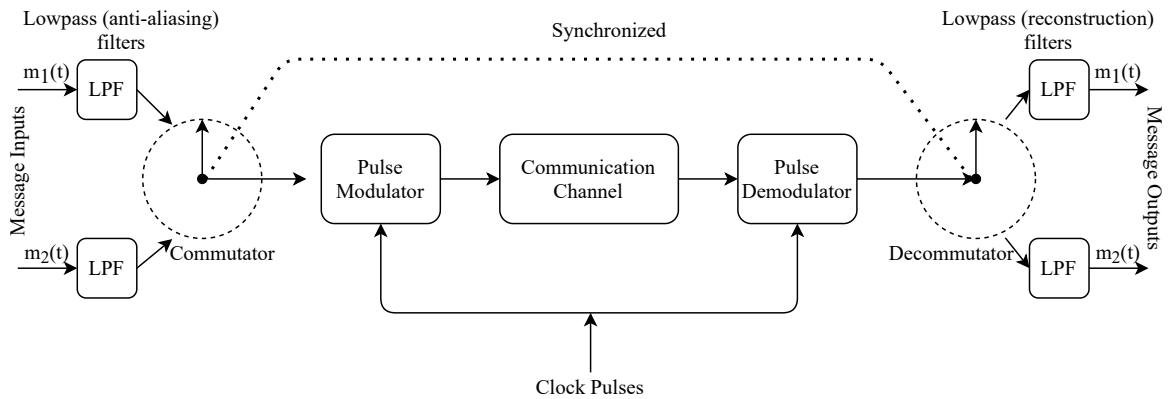


Figure 1: Block diagram of TDM

Figure 1 shows the block diagram for TDM, where each input message signal is band-limited with a lowpass pre-alias filter to remove the frequencies that are considered non-essential for transmission. Then the pre-alias filter outputs are applied to a commutator (implemented using electronic switching circuit) that takes a narrow sample of each of the N input message

at the rate of f_s that is higher than $2f_m$, where f_m is the cutoff frequency of pre-alias filter. It also performs the task of interleaving these N samples inside a sampling interval $T_s = \frac{1}{f_s}$. The multiplexed signal is then applied in a pulse-amplitude modulator to transform the signal to a form suitable for transmission over the channel. At the receiver side, the signal is applied to a pulse-amplitude demodulator that produces short pulses which are distributed to appropriate lowpass reconstruction filters by the decommutator working in sync with the commutator.

2.2 Sampling and signal reconstruction using samples

Sampling is a process by which continuous time signals are converted into discrete time signal called samples. For a continuous time signal $s(t)$, the sampled discrete time signal $s[n]$ and sampling period of T_s , the sampling equation is given as,

$$s[n] = s(nT_s) \quad (1)$$

Sampling frequency $f_s = \frac{1}{T_s}$ is used to denote sampling rate. For a continuous time signal $s(t)$ with f_m as the highest frequency component, the Nyquist-Shannon theorem confines the sampling rate as,

$$f_s > 2f_m \quad (2)$$

Equation 2 must be satisfied to avoid aliasing of signal during sampling and for perfect reconstruction of the signal from its samples. The reconstruction of a signal using its samples is performed using a reconstruction filter, that is an ideal lowpass filter with cutoff at sampling frequency f_s . The passage of signal through this reconstruction filter corresponds to a convolution with the *sinc* function in time domain. So a *sinc* signal appears centered at each integer multiple of T_s during reconstruction, which eventually settles to give the actual signal reconstructed using the samples, which is why this method of reconstruction is also known as ideal interpolation of samples.

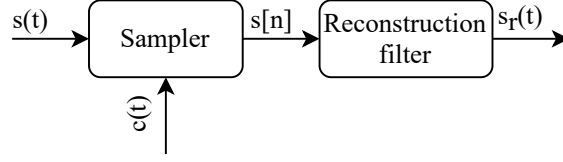


Figure 2: Block diagram of sampling and reconstruction using samples

2.3 Delta modulation

Delta modulation is a type of modulation where the sampling rate is much greater and the step-size after quantization is a smaller value Δ . It is a simplified version of the Differential Pulse Code Modulation (DPCM). It uses a 1-bit quantizer and a delay circuit with two summer circuits which is why it is also known as 1-bit DPCM scheme. Only one binary bit per sample is transmitted.

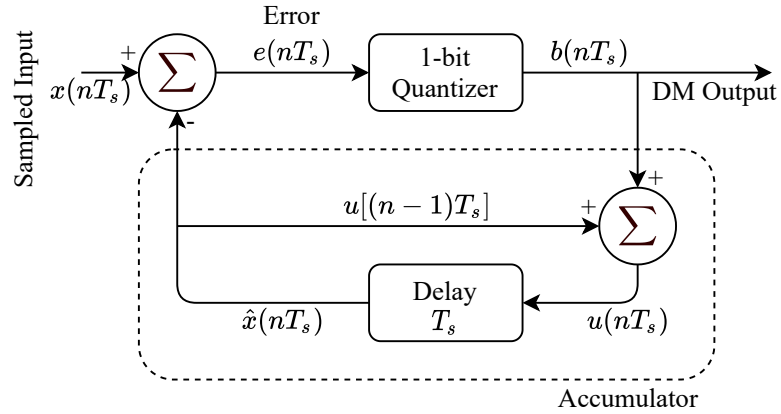


Figure 3: Block diagram of delta modulation at transmitter

Figure 3 shows the block diagram for the delta modulation at transmitter. The summer in the accumulator adds quantizer output ($\pm\Delta$) with the previous sample approximation which gives present sample approximation as,

$$u(nT_s) = u[(nT_s - T_s)] + [\pm\Delta]$$

$$u(nT_s) = u[(n-1)T_s] + b(nT_s)$$

The previous sample approximation $u[(n-1)T_s]$ is restored by introducing delay T_s . The samples input signal $x(nT_s)$ and staircase approximated signal $\hat{x}(nT_s)$ are subtracted to get error signal $e(nT_s)$, depending on which the 1-bit quantizer gives either $+\Delta$ or $-\Delta$ as the step-size. For $+\Delta$ binary bit '1' is transmitted and for $-\Delta$ binary bit '0' is transmitted.

3 Exercises

Problem 1

Perform time division multiplexing of two signals and recover the original signals.

```
1 % Signal generation
2 x = 0:0.1:4 * pi;
3 signal_a = 8 * sin(x);
4 l = length(signal_a);
5 signal_b = 8 * triang(1);
6 g = tiledlayout(4, 2);
7 title(g, 'Time Division Multiplexing (
    PUL074BEX007)')
8
9 % Displaying og signals
10 nexttile
11 plot(signal_a);
12 title('Signal a');
13 grid on
14 nexttile
15 plot(signal_b);
16 title('Signal b');
17 grid on
18
19 % Displaying stem-version of og signals
20 nexttile
21 stem(signal_a);
22 title('Signal a (Quantized)');
23 nexttile
24 stem(signal_b);
25 title('Signal b (Quantized)');
26
27 % Conversion of row vector to a matrix
28 l1 = length(signal_a);
29 l2 = length(signal_b);
30 for i = 1:l1
31     signal(1, i) = signal_a(i);
32     signal(2, i) = signal_b(i);
33 end
34
35 % Multiplexing (TDM) of quantized signals
36 tdmsig = reshape(signal, 1, 2 * l1);
37 nexttile([1, 2])
38 stem(tdmsig);
39 title('TDM signal');
40
41 % Demultiplexing of TDM signal
42 demux = reshape(tdmsig, 2, l1);
43 % Conversion of a matrix to row vectors
44 for i = 1:l1
45     signal_a_recovered(i) = demux(1, i);
46     signal_b_recovered(i) = demux(2, i);
47 end
48
49 % Displaying recovered signals
50 nexttile
51 plot(signal_a_recovered);
52 grid on
53 title('Recovered signal a');
54 nexttile
55 plot(signal_b_recovered);
```



```
56 title('Recovered signal b');
57 grid on
```

Listing 1: Matlab script for time division multiplexing and recovery of original signals

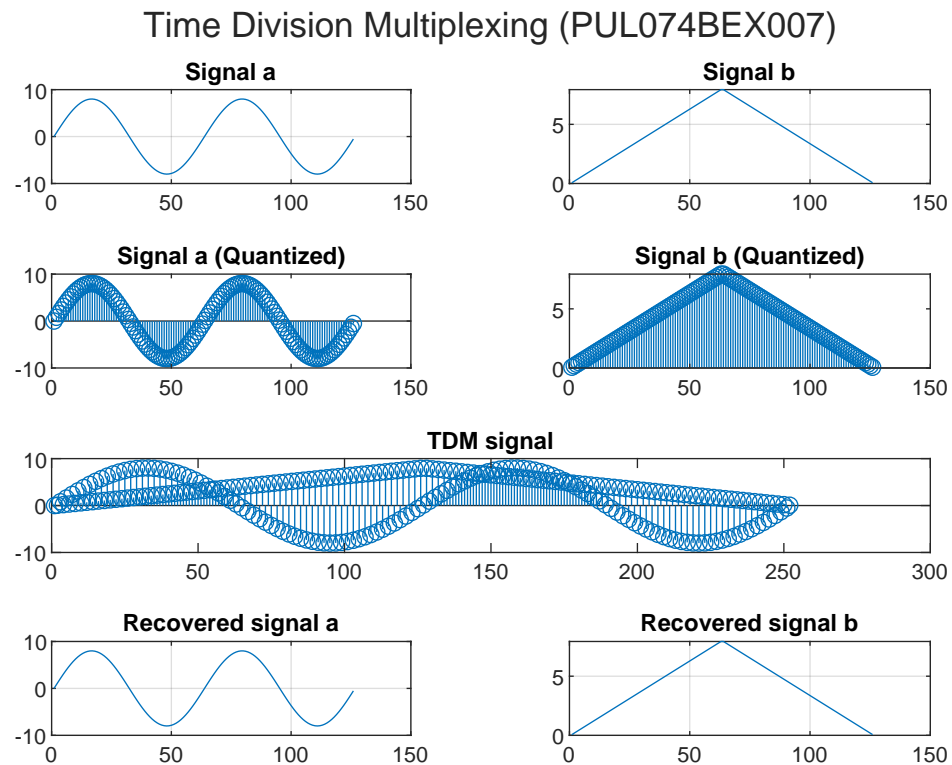


Figure 4: Time division multiplexing of two signals and recovered signals

Problem 2

Perform reconstruction of a signal from its samples.

```
1 % Signal generation
2 t = 0:0.1:20;
3 f1 = 0.1;
4 f2 = 0.2;
5 x = sin(2 * pi * f1 * t) + sin(2 * pi * f2
    * t);
6 g = tiledlayout(3, 1);

7 title(g, 'Sampling and Signal
    Reconstruction (PUL074BEX007)')
8
9 % Displaying og signal
10 nexttile
11 plot(t, x);
12 hold on;
```

```

13 plot([0 20], [0 0], 'black');
14 hold off;
15 title('Original signal');
16 xlabel('Time');
17 ylabel('Amplitude');
18
19 % Sampling of og signal
20 nexttile
21 x_samples = x(1:10:201);
22 stem(x_samples, 'filled');
23 title('Sampled signal');
24 xlabel('n');
25 ylabel('x_s(n)');
26 axis([1 21 -2 2]);
27
28 % Reconstruction using samples
29 x_recon = 0;
30 nexttile
31 for k = 0:length(x_samples) - 1
32     stem(0:length(x_samples) - 1, x_samples
        , 'filled');
33
34     if k == length(x_samples) - 1
35         title('Reconstructed signal');
36     else
37         title(sprintf('Sample by sample
38             reconstruction at n=%d', k));
39     end
40     grid on;
41     l = k:-0.1:-20 + k;
42     x_recon = x_recon + (x_samples(k + 1) *
43         sinc(l));
44     axis([0 20 -2 2]);
45     xlabel('n');
46     ylabel('x_s(n)');
47     hold on;
48     plot(t, x_samples(k + 1) * sinc(l), 'r'
49 );
50     plot(t, x_recon);
51     hold off;
52     waitforbuttonpress;
53 end

```

Listing 2: Matlab script for sampling and reconstruction of a signal from its samples

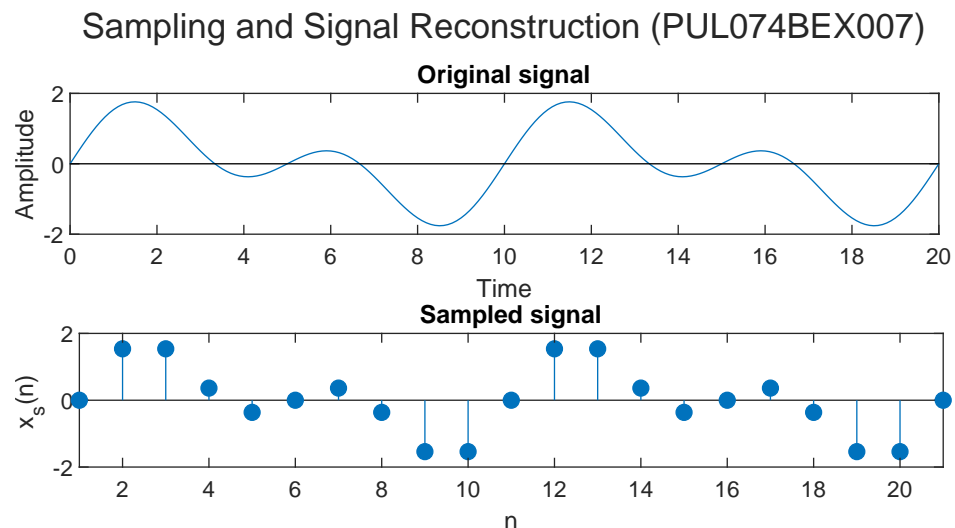


Figure 5: Reconstruction of a signal from its samples

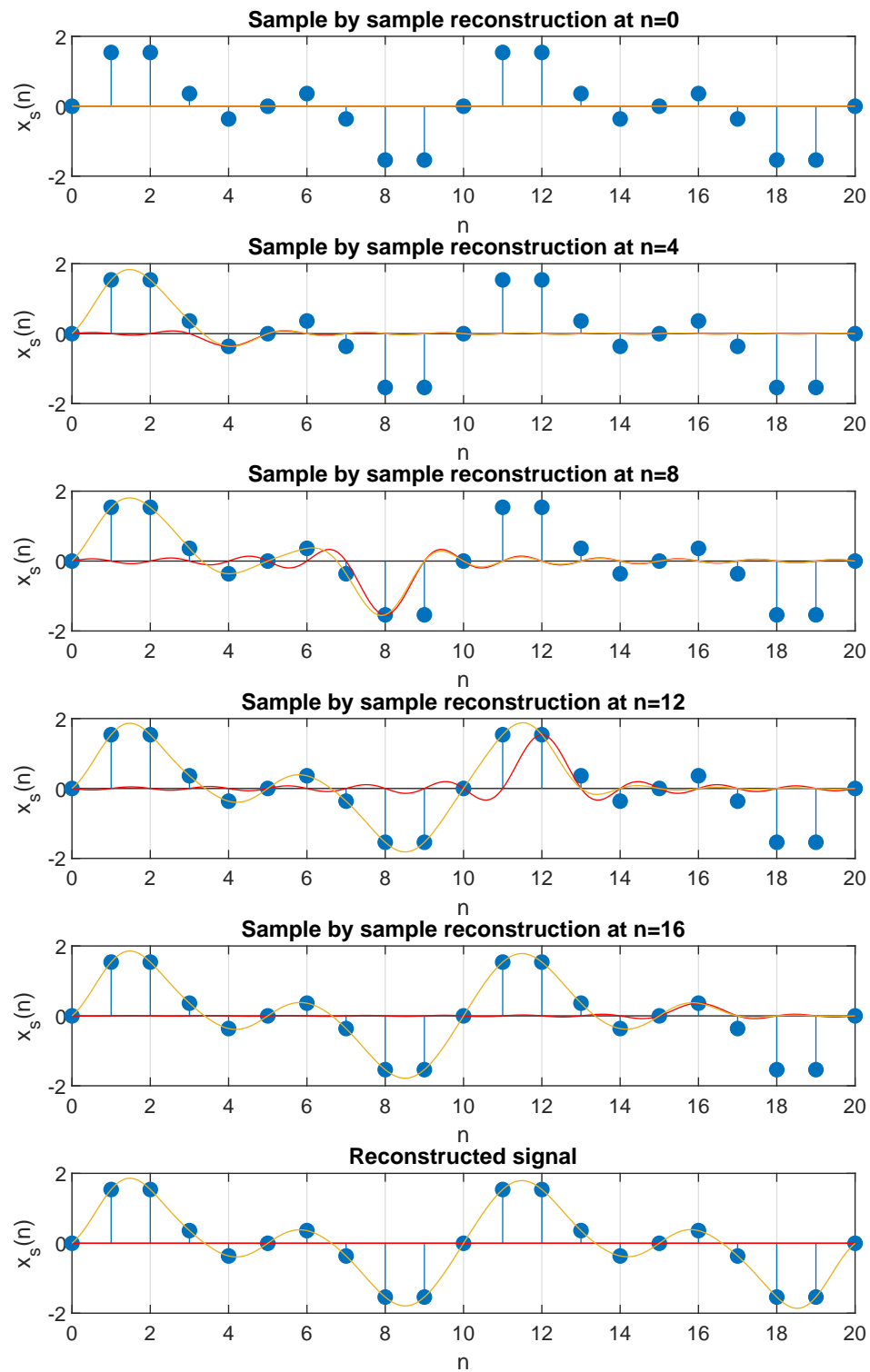


Figure 5: Reconstruction of a signal from its samples (continued)

Problem 3

Perform delta modulation and visualize the bit sequence for delta modulation transmission.

```

1 % Signal generation
2 am = 5;
3 fm = 1;
4 fs = 50 * fm;
5 t = 0:1 / fs:((1 / fm) - (1 / fs));
6 x = am * sin(2 * pi * fm * t);
7 g = tiledlayout(2, 1);
8 title(g, 'Delta Modulation (PUL074BEX007)')
9
10 % Displaying og signal
11 nexttile
12 plot(x);
13 axis([0 50 -6 6]);
14 hold on;
15
16 % Delta modulation
17 del = (2 * pi * am * fm) / fs;
18 for i = 1:length(x)
19     if i == 1
20         e(i) = x(i);
21         eq(i) = del * sign(e(i));
22         xq(i) = eq(i);
23     else
24         e(i) = x(i) - xq(i - 1);
25         eq(i) = del * sign(e(i));
26         xq(i) = eq(i) + xq(i - 1);
27     end
28 end
29
30 % Bit sequence for DM transmission
31 for i = 1:length(x)
32     if e(i) > 0
33         dm(i) = 1;
34     else
35         dm(i) = 0;
36     end
37 end
38
39 % Displaying DM stairs over og signal
40 stairs(xq, 'LineWidth', 0.8, 'Color', '#
    D95319');
41 title('Delta modulation of original signal'
    );
42
43 % Displaying bit sequence
44 nexttile
45 stairs(dm, 'LineWidth', 1, 'Color', '#
    D95319');
46 axis([0 50 -0.2 1.2]);
47 title('Bit sequence for DM transmission');
```

Listing 3: Matlab script for delta modulation and bit sequence for delta modulation transmission

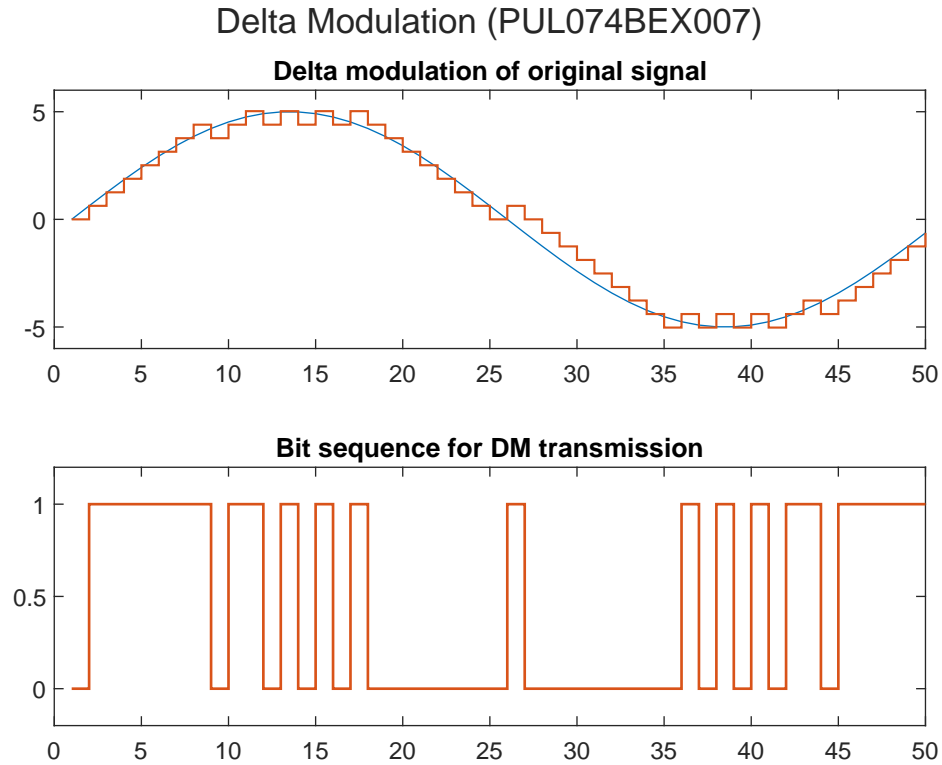


Figure 6: Delta modulation and bit sequence for delta modulation transmission

4 Discussion and Conclusion

The lab experiment dealt with the realization of time division multiplexing, signal reconstruction using samples and delta modulation.

TDM is a concept where multiple quantized signals are multiplexed in time and transmitted over the same channel as the multiplexed signal. On the receiving end a demultiplexer is used to recover the original signals. TDM allows low capacity communication channels to be used with little to no crosstalk while utilizing the full channel bandwidth.

Signal reconstruction using samples is a necessary topic of discussion since sampled signals can be used to recover the original signal with great accuracy. This is realized using a reconstruction filter which we coded in MATLAB. One thing to be noted is that sampling must suffice the Nyquist criteria so that signal integrity isn't lost on reconstruction.

Delta modulation is an AD/DA conversion technique mostly used for voice transmission where timely data delivery is prioritized over data quality. It requires less channel bandwidth than PCM since it generates only one bit per sample.

Hence the objectives of the lab experiment were fulfilled and the visualization of TDM, reconstruction of signals using samples and delta modulation were performed.