**Experiment 1**

**Name – Shivam Malviya**

**Roll No. - 18EC01044**

**Theory**

* Sine: y = Asin(2πft)
* Square: y(t) = { A , t ∈ (−T/2, T/2)

0, Otherwise}

* Saw tooth: y(t) = { At, t ∈ (0, T)

0, Otherwise}

* Triangular: y(t) = { At, t ∈ (0, T/2)

A(T − t), t ∈ (T/2, T) }

* Impulse: y(t) = { 1, t = 0

0, Otherwise}

* Step: y(t) = { A, t ≥ 0

0, Otherwise }

* Pulse: y(t) = { At, t ∈ (0, βT)

0, t ∈ (βT, T) } β:Duty Cycle

**Even and Odd Component**

The even and odd component of x(t) can given by equations shown below:

* xe(t) = (x(t) + x(−t)) / 2
* xo(t) = (x(t) − x(−t)) / 2

Signal Transformation:

**Amplitude Scaling:**

If the signal x(t) is scaled by factor A then it means

x(t) = A\*x(t)

**Time scaling:**

If a signal x(t) is scaled by a factor τ, then its means

x(t) = x(t/ τ)

**Time shifting:**

If a signal x(t) is shifted by τ, then it means

x(t) = x(t – τ)

**Folding:**

After folding a signal x(t), becomes x(−t).

**APPENDEX**

**Function of Signals**

* **Sin**

**function y = fn\_sin(t);**

**y = sin(2\* pi \* t);**

* **Square**

**function y = fn\_square(T, A, t);**

**T = T/2;**

**y = 0;**

**if (-T < t && t < T)**

**y = A;**

**end**

* **Sawtooth**

**function y = fn\_sawtooth(T, A, t);**

**y = 0;**

**t = rem(t, T);**

**if (t < 0)**

**t = T + t;**

**end**

**if (t < T)**

**y = A\*t;**

**end**

* **Triangular**

**function y = fn\_triangular(T, A, t)**

**t = rem(t, T);**

**if (t < 0)**

**t = T + t;**

**end**

**if (t < T/2)**

**y = A\*t;**

**else**

**y = A\*(T-t);**

**end**

* **Impulse**

function y = fn\_impulse(t)

y = 0;

if (t == 0)

y = 1;

end

* **Step**

function y = fn\_step(A, t)

if (0 <= t)

y = A;

else

y = 0;

end

* **Pulse**

function y = fn\_pulse(T, A, beta, t);

y = 0;

t = rem(t, T);

if (t < 0)

t = T + t;

end

if (t < T\*beta)

y = A ;

end

* **Even Component**

**function even\_plot(fun)**

**fun = @(t) ((fun(t) + fun(-t))/2);**

**Plot(fun);**

* **Odd Component**

**function odd\_plot(fun)**

**fun = @(t) ((fun(t) - fun(-t))/2);**

**Plot(fun);**

* **Plot**

**function Plot(fun, tmin = -10, tmax = 10);**

**xran = [-10 10];**

**t = tmin : 0.01 : tmax;**

**n = length(t);**

**y = zeros(1, n);**

**for x = 1:n;**

**y(x) = fun(t(x));**

**end**

**plot(t, y, '-r', 'LineWidth', 3);**

**xlabel('Time');**

**ylabel('Signal');**

**xlim(xran);**

**ylim([2\*min(y) 2\*max(y)]);**

* **Plot all**

**function plot\_all(f)**

**figure('name', 'Function');**

**Plot(f);**

**figure('name', 'Even Component');**

**even\_plot(f);**

**figure('name', 'Odd Component');**

**odd\_plot(f);**

**amplitude\_scale(f);**

**time\_scale(f);**

**time\_shift(f);**

**folding(f);**

* **Amplitude Scale**

**function fun= amplitude\_scale(fun, A = 2)**

**fun = @(t) (A \* fun(t));**

**figure('name', 'Amplitude Scaling');**

**Plot(fun);**

* **Time Scale**

**function fun = time\_scale(fun, tau = 2);**

**fun = @(t) fun(t/tau);**

**figure('name', 'Time Scaling');**

**Plot(fun);**

* **Time Shift**

**function fun = time\_shift(fun, tau = 2)**

**fun = @(t) fun(t-tau);**

**figure('name', 'Time Shifting');**

**Plot(fun);**

* **Folding**

**function fun = folding(fun)**

**fun = @(t) fun(-t);**

**figure('name', 'Folding');**

**Plot(fun);**