

# **Digital Design for DSP & Communication**

## **EE-278**

### **Mini Project- IV**

#### **Parameterizable LMS Adaptive filter – DSP IP core**

**Under supervision of**

**Dr. Charles Chang Choo**

**By**

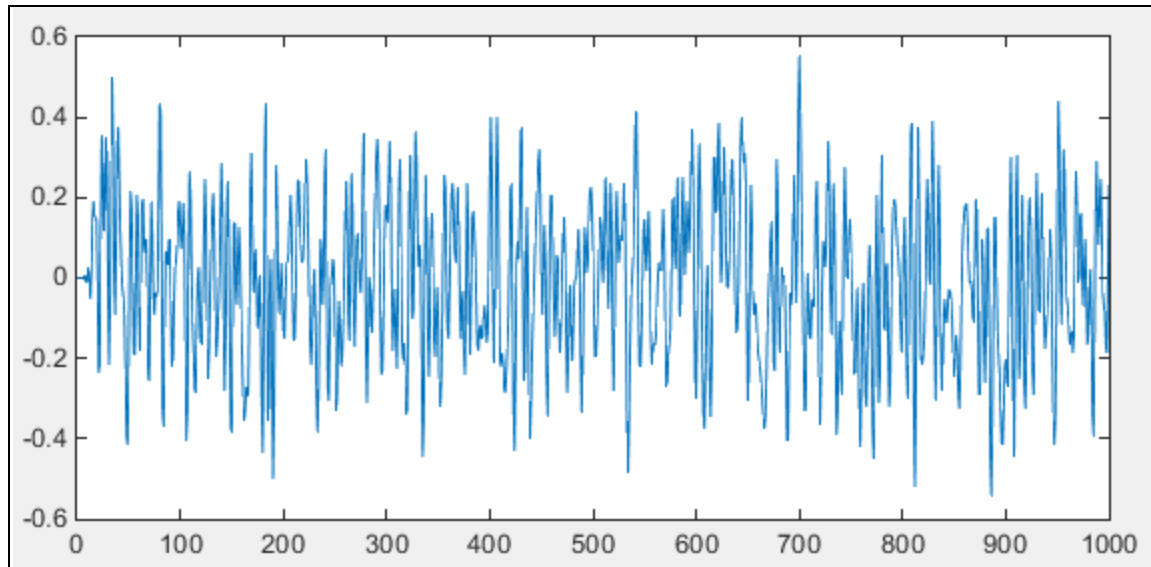
**Dhatri Patel**

**SJSU ID: 010692526**

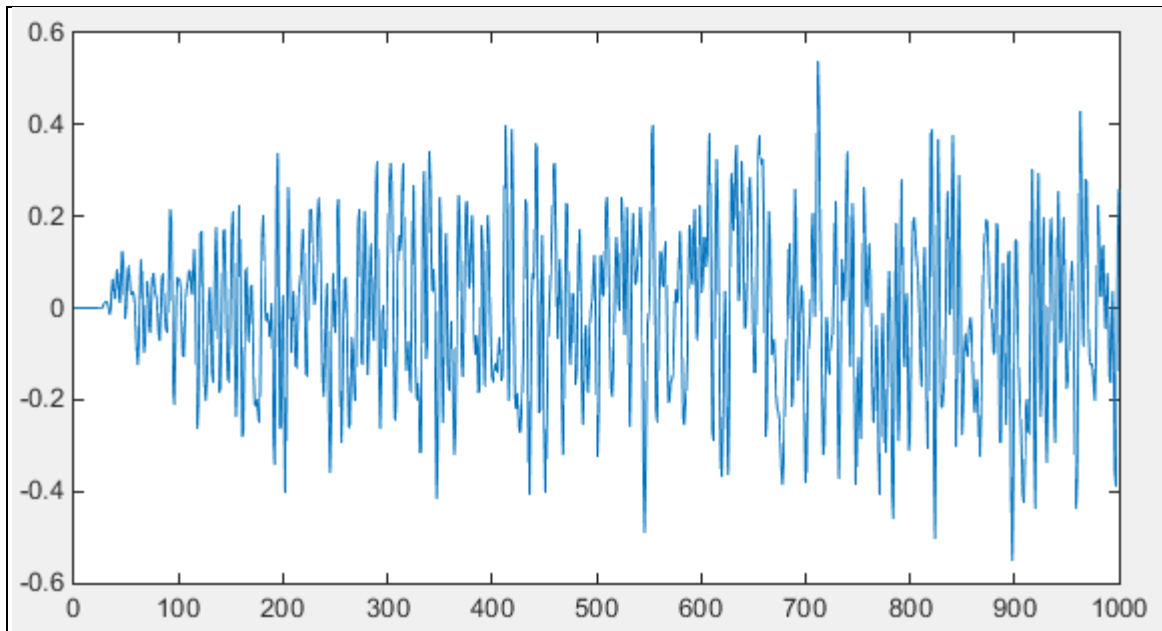
## MATLAB Code

```
ntaps = 24 ;
nsamp = 1000;
ibeta = 8; %2^8; %  $\mu=1/2^8$ 
amp = 1;
refsig = amp*rand(1,nsamp) - (amp/2);
% generate input signal x(t)
Wn = 0.47;
B = fir1(30,Wn);
x = filter(B,1,refsig);
x = x';
% generate desired signal d(t)
Coeff = fir1(ntaps,0.5);
d = filter(Coeff,1,x);
%%%% by now, we have X and D.
y = zeros(1,nsamp);
e = zeros(1,nsamp);
h(1:ntaps) = 0;
for n=1:nsamp
    if n < ntaps
        x1 = [x(n:-1:1)' zeros(1,ntaps-n)];
    else
        x1 = x(n:-1:n-ntaps+1)';
    end
    y(n) = h * x1';
    e(n) = d(n) - y(n);
    hh = e(n)*x1/ibeta;
    h = h + hh;
end
```

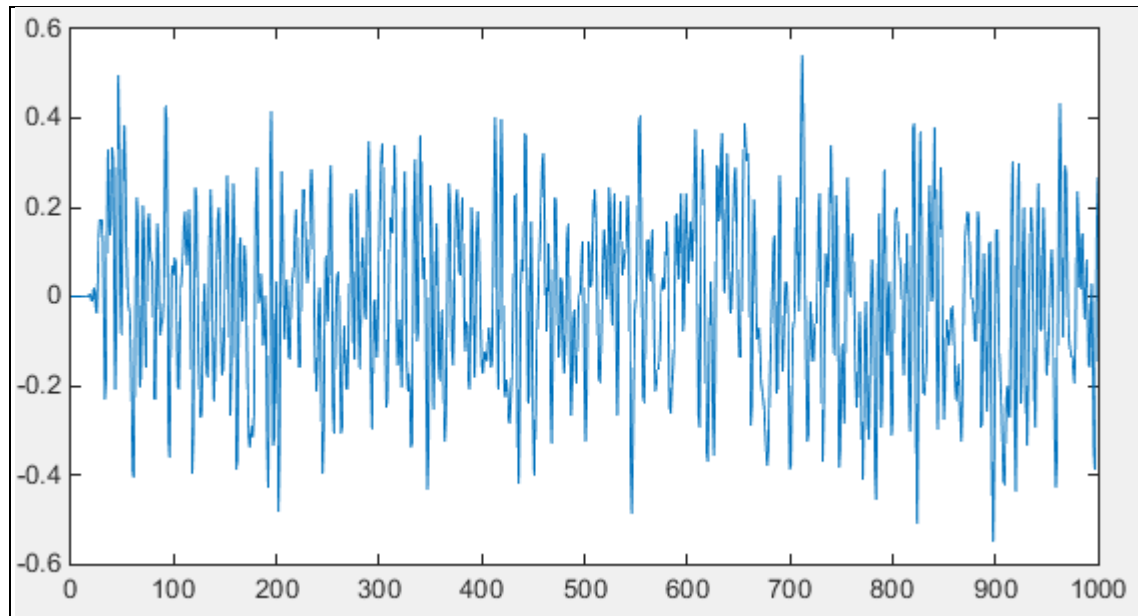
## **Output Waveforms in MATLAB**



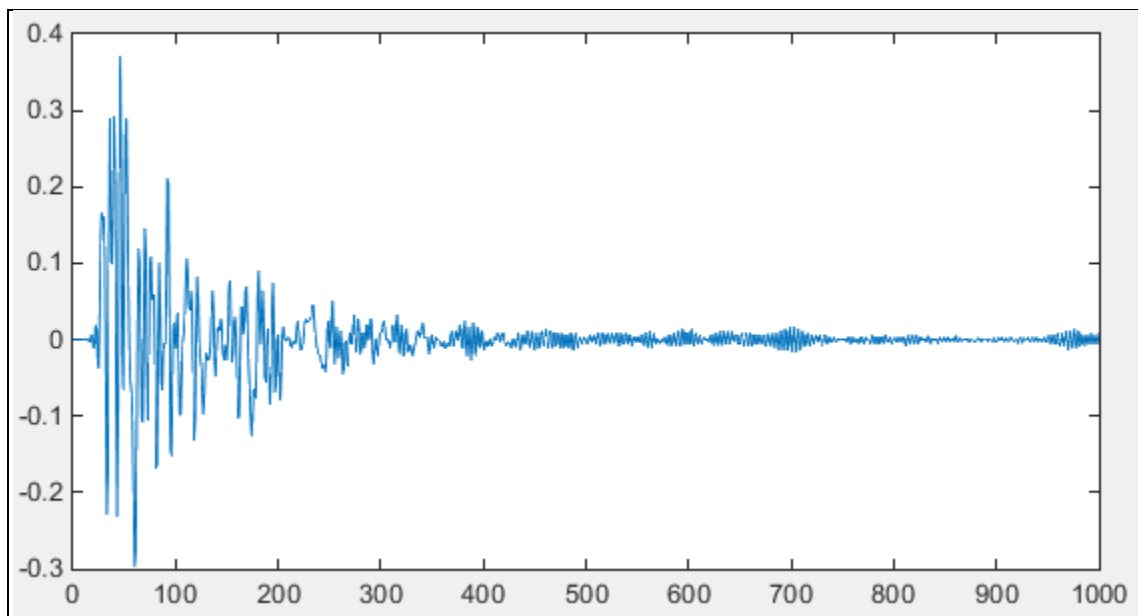
(Fig. 1: 'X' signal plot)



(Fig. 2: 'Y' signal plot)



(Fig. 3: 'd' signal plot)



(Fig. 4: 'e' signal plot)

## Verilog Code for Sequential LMS

```

`timescale 1ns/1ps
module lms_26 (clock,reset);

input clock,reset;
parameter fix_size =32;

reg [10:0]addr =0;
wire signed[fix_size-1:0] x_input_26;

reg signed[fix_size-1:0]
h1,h2,h3,h4,h5,h6,h7,h8,h9,h10,h11,h12,h13,h14,h15,h16,h17,h18,h19,h20
,h21,h22,h23,h24,h_1,h_2,h_3,h_4,h_5,h_6,h_7,h_8,h_9,h_10,h_11,h_12,h_
13,h_14,h_15,h_16,h_17,h_18,h_19,h_20,h_21,h_22,h_23,h_24;

reg signed[(2*fix_size)-1:0]
h_d1,h_d2,h_d3,h_d4,h_d5,h_d6,h_d7,h_d8,h_d9,h_d10,h_d11,h_d12
,h_d13,h_d14,h_d15,h_d16,h_d17,h_d18,h_d19,h_d20,h_d21,h_d22,h_d2
3,h_d24;

reg signed[(2*fix_size)-1:0]
h_f1,h_f2,h_f3,h_f4,h_f5,h_f6,h_f7,h_f8,h_f9,h_f10,h_f11,h_f12
,h_f13,h_f14,h_f15,h_f16,h_f17,h_f18,h_f19,h_f20,h_f21,h_f22,h_f2
3,h_f24;

reg signed[fix_size-1:0]
h_r1,h_r2,h_r3,h_r4,h_r5,h_r6,h_r7,h_r8,h_r9,h_r10,h_r11,h_r12
,h_r13,h_r14,h_r15,h_r16,h_r17,h_r18,h_r19,h_r20,h_r21,h_r22,h_r2
3,h_r24;

reg signed[fix_size-1:0]
h_r1_26,h_r2_26,h_r3_26,h_r4_26,h_r5_26,h_r6_26,h_r7_26,h_r8_26,h_r9_2
6,h_r10_26,h_r11_26,h_r12_26
,h_r13_26,h_r14_26,h_r15_26,h_r16_26,h_r17_26,h_r18_26,h_r19_26,h
_r20_26,h_r21_26,h_r22_26,h_r23_26,h_r24_26;

reg signed[fix_size-1:0] x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x12
,x13,x14,x15,x16,x17,x18,x19,x20,x21,x22,x23,x24;

reg signed[(2*fix_size)-1:0] y1,y2,y3,y4,y5,y6,y7,y8,y9,y10,y11,y12

```

```

        ,y13,y14,y15,y16,y17,y18,y19,y20,y21,y22,y23,y24;

reg signed[fix_size-1:0]
y_d1,y_d2,y_d3,y_d4,y_d5,y_d6,y_d7,y_d8,y_d9,y_d10,y_d11,y_d12
        ,y_d13,y_d14,y_d15,y_d16,y_d17,y_d18,y_d19,y_d20,y_d21,y_d22,y_d2
3,y_d24;

reg signed[(2*fix_size)-1:0]
y_r1,y_r2,y_r3,y_r4,y_r5,y_r6,y_r7,y_r8,y_r9,y_r10,y_r11,y_r12
        ,y_r13,y_r14,y_r15,y_r16,y_r17,y_r18,y_r19,y_r20,y_r21,y_r22,y_r2
3,y_r24;
reg signed[fix_size-1:0] y_f;

reg signed[fix_size-1:0] a1_input_26, a2_input_26, a3_input_26,
a4_input_26
        , a5_input_26, a6_input_26, a7_input_26,
a8_input_26
        , a9_input_26, a10_input_26, a11_input_26,
a12_input_26
        , a13_input_26, a14_input_26, a15_input_26,
a16_input_26
        , a17_input_26, a18_input_26, a19_input_26,
a20_input_26
        , a21_input_26, a22_input_26, a23_input_26,
a24_input_26;

reg signed[(2*fix_size)-1:0] y_26;

reg signed[fix_size-1:0] desired_26;
reg signed[fix_size-1:0] error_26,error1_26;
wire signed[fix_size-1:0] d;

lms_ram_26
ram_x(.address(addr),.clock(clock),.data(32'b0),.wren(1'b0),.q(x_input
_26));
lms_ramd_26
ram_d(.address(addr),.clock(clock),.data(32'b0),.wren(1'b0),.q(d));

always@(*)
begin
    y1 = h1*x_input_26;
    y2 = h2*x1;
    y3 = h3*x2;
    y4 = h4*x3;
    y5 = h5*x4;
    y6 = h6*x5;
    y7 = h7*x6;
    y8 = h8*x7;

```

```

y9 = h9*x8;
y10 = h10*x9;
y11 = h11*x10;
y12 = h12*x11;
y13 = h13*x12;
y14 = h14*x13;
y15 = h15*x14;
y16 = h16*x15;
y17 = h17*x16;
y18 = h18*x17;
y19 = h19*x18;
y20 = h20*x19;
y21 = h21*x20;
y22 = h22*x21;
y23 = h23*x22;
y24 = h24*x23;

```

```

y_d1 = y1[(2*fix_size)-2:fix_size-1];
y_d2 = y2[(2*fix_size)-2:fix_size-1];
y_d3 = y3[(2*fix_size)-2:fix_size-1];
y_d4 = y4[(2*fix_size)-2:fix_size-1];
y_d5 = y5[(2*fix_size)-2:fix_size-1];
y_d6 = y6[(2*fix_size)-2:fix_size-1];
y_d7 = y7[(2*fix_size)-2:fix_size-1];
y_d8 = y8[(2*fix_size)-2:fix_size-1];
y_d9 = y9[(2*fix_size)-2:fix_size-1];
y_d10 = y10[(2*fix_size)-2:fix_size-1];
y_d11 = y11[(2*fix_size)-2:fix_size-1];
y_d12 = y12[(2*fix_size)-2:fix_size-1];
y_d13 = y13[(2*fix_size)-2:fix_size-1];
y_d14 = y14[(2*fix_size)-2:fix_size-1];
y_d15 = y15[(2*fix_size)-2:fix_size-1];
y_d16 = y16[(2*fix_size)-2:fix_size-1];
y_d17 = y17[(2*fix_size)-2:fix_size-1];
y_d18 = y18[(2*fix_size)-2:fix_size-1];
y_d19 = y19[(2*fix_size)-2:fix_size-1];
y_d20 = y20[(2*fix_size)-2:fix_size-1];
y_d21 = y21[(2*fix_size)-2:fix_size-1];
y_d22 = y22[(2*fix_size)-2:fix_size-1];
y_d23 = y23[(2*fix_size)-2:fix_size-1];
y_d24 = y24[(2*fix_size)-2:fix_size-1];

```

```

y_r1 = (y1[fix_size-2])?(y_d1+1'b1):y_d1;
y_r2 = (y2[fix_size-2])?(y_d2+1'b1):y_d2;
y_r3 = (y3[fix_size-2])?(y_d3+1'b1):y_d3;
y_r4 = (y4[fix_size-2])?(y_d4+1'b1):y_d4;
y_r5 = (y5[fix_size-2])?(y_d5+1'b1):y_d5;
y_r6 = (y6[fix_size-2])?(y_d6+1'b1):y_d6;
y_r7 = (y7[fix_size-2])?(y_d7+1'b1):y_d7;

```

```

y_r8 = (y8[fix_size-2]))?(y_d8+1'b1):y_d8;
y_r9 = (y9[fix_size-2]))?(y_d9+1'b1):y_d9;
y_r10 = (y10[fix_size-2]))?(y_d10+1'b1):y_d10;
y_r11 = (y11[fix_size-2]))?(y_d11+1'b1):y_d11;
y_r12 = (y12[fix_size-2]))?(y_d12+1'b1):y_d12;
y_r13 = (y13[fix_size-2]))?(y_d13+1'b1):y_d13;
y_r14 = (y14[fix_size-2]))?(y_d14+1'b1):y_d14;
y_r15 = (y15[fix_size-2]))?(y_d15+1'b1):y_d15;
y_r16 = (y16[fix_size-2]))?(y_d16+1'b1):y_d16;
y_r17 = (y17[fix_size-2]))?(y_d17+1'b1):y_d17;
y_r18 = (y18[fix_size-2]))?(y_d18+1'b1):y_d18;
y_r19 = (y19[fix_size-2]))?(y_d19+1'b1):y_d19;
y_r20 = (y20[fix_size-2]))?(y_d20+1'b1):y_d20;
y_r21 = (y21[fix_size-2]))?(y_d21+1'b1):y_d21;
y_r22 = (y22[fix_size-2]))?(y_d22+1'b1):y_d22;
y_r23 = (y23[fix_size-2]))?(y_d23+1'b1):y_d23;
y_r24 = (y24[fix_size-2]))?(y_d24+1'b1):y_d24;

y_f= y_r1 + y_r2 + y_r3 + y_r4 + y_r5 + y_r6 + y_r7 + y_r8
      + y_r9 + y_r10 + y_r11 + y_r12 + y_r13 + y_r14 + y_r15
+ y_r16
      + y_r17 + y_r18 + y_r19 + y_r20 + y_r21 + y_r22 +
y_r23 + y_r24;

error_26 = desired_26 - y_f;

h_d1 = error_26*x1;
h_d2 = error_26*x2;
h_d3 = error_26*x3;
h_d4 = error_26*x4;
h_d5 = error_26*x5;
h_d6 = error_26*x6;
h_d7 = error_26*x7;
h_d8 = error_26*x8;
h_d9 = error_26*x9;
h_d10 = error_26*x10;
h_d11 = error_26*x11;
h_d12 = error_26*x12;
h_d13 = error_26*x13;
h_d14 = error_26*x14;
h_d15 = error_26*x15;
h_d16 = error_26*x16;
h_d17 = error_26*x17;
h_d18 = error_26*x18;
h_d19 = error_26*x19;
h_d20 = error_26*x20;
h_d21 = error_26*x21;
h_d22 = error_26*x22;
h_d23 = error_26*x23;
h_d24 = error_26*x24;

h_f1 = (h_d1[1]))?((h_d1>>>2) + 1):(h_d1>>>2);

```



```

h_f2 = (h_d2[1])?((h_d2>>>2) + 1):(h_d2>>>2);
h_f3 = (h_d3[1])?((h_d3>>>2) + 1):(h_d3>>>2);
h_f4 = (h_d4[1])?((h_d4>>>2) + 1):(h_d4>>>2);
h_f5 = (h_d5[1])?((h_d5>>>2) + 1):(h_d5>>>2);
h_f6 = (h_d6[1])?((h_d6>>>2) + 1):(h_d6>>>2);
h_f7 = (h_d7[1])?((h_d7>>>2) + 1):(h_d7>>>2);
h_f8 = (h_d8[1])?((h_d8>>>2) + 1):(h_d8>>>2);
h_f9 = (h_d9[1])?((h_d9>>>2) + 1):(h_d9>>>2);
h_f10 = (h_d10[1])?((h_d10>>>2) + 1):(h_d10>>>2);
h_f11 = (h_d11[1])?((h_d11>>>2) + 1):(h_d11>>>2);
h_f12 = (h_d12[1])?((h_d12>>>2) + 1):(h_d12>>>2);
h_f13 = (h_d13[1])?((h_d13>>>2) + 1):(h_d13>>>2);
h_f14 = (h_d14[1])?((h_d14>>>2) + 1):(h_d14>>>2);
h_f15 = (h_d15[1])?((h_d15>>>2) + 1):(h_d15>>>2);
h_f16 = (h_d16[1])?((h_d16>>>2) + 1):(h_d16>>>2);
h_f17 = (h_d17[1])?((h_d17>>>2) + 1):(h_d17>>>2);
h_f18 = (h_d18[1])?((h_d18>>>2) + 1):(h_d18>>>2);
h_f19 = (h_d19[1])?((h_d19>>>2) + 1):(h_d19>>>2);
h_f20 = (h_d20[1])?((h_d20>>>2) + 1):(h_d20>>>2);
h_f21 = (h_d21[1])?((h_d21>>>2) + 1):(h_d21>>>2);
h_f22 = (h_d22[1])?((h_d22>>>2) + 1):(h_d22>>>2);
h_f23 = (h_d23[1])?((h_d23>>>2) + 1):(h_d23>>>2);
h_f24 = (h_d24[1])?((h_d24>>>2) + 1):(h_d24>>>2);

```

```

h_r1 = h_f1[(2*fix_size)-2:fix_size-1];
h_r2 = h_f2[(2*fix_size)-2:fix_size-1];
h_r3 = h_f3[(2*fix_size)-2:fix_size-1];
h_r4 = h_f4[(2*fix_size)-2:fix_size-1];
h_r5 = h_f5[(2*fix_size)-2:fix_size-1];
h_r6 = h_f6[(2*fix_size)-2:fix_size-1];
h_r7 = h_f7[(2*fix_size)-2:fix_size-1];
h_r8 = h_f8[(2*fix_size)-2:fix_size-1];
h_r9 = h_f9[(2*fix_size)-2:fix_size-1];
h_r10 = h_f10[(2*fix_size)-2:fix_size-1];
h_r11 = h_f11[(2*fix_size)-2:fix_size-1];
h_r12 = h_f12[(2*fix_size)-2:fix_size-1];
h_r13 = h_f13[(2*fix_size)-2:fix_size-1];
h_r14 = h_f14[(2*fix_size)-2:fix_size-1];
h_r15 = h_f15[(2*fix_size)-2:fix_size-1];
h_r16 = h_f16[(2*fix_size)-2:fix_size-1];
h_r17 = h_f17[(2*fix_size)-2:fix_size-1];
h_r18 = h_f18[(2*fix_size)-2:fix_size-1];
h_r19 = h_f19[(2*fix_size)-2:fix_size-1];
h_r20 = h_f20[(2*fix_size)-2:fix_size-1];
h_r21 = h_f21[(2*fix_size)-2:fix_size-1];
h_r22 = h_f22[(2*fix_size)-2:fix_size-1];
h_r23 = h_f23[(2*fix_size)-2:fix_size-1];
h_r24 = h_f24[(2*fix_size)-2:fix_size-1];

```

```

h_r1_26 = (h_f1[fix_size-2]) ? (h_r1+1) : h_r1;
h_r2_26 = (h_f2[fix_size-2]) ? (h_r2+1) : h_r2;
h_r3_26 = (h_f3[fix_size-2]) ? (h_r3+1) : h_r3;
h_r4_26 = (h_f4[fix_size-2]) ? (h_r4+1) : h_r4;
h_r5_26 = (h_f5[fix_size-2]) ? (h_r5+1) : h_r5;
h_r6_26 = (h_f6[fix_size-2]) ? (h_r6+1) : h_r6;
h_r7_26 = (h_f7[fix_size-2]) ? (h_r7+1) : h_r7;
h_r8_26 = (h_f8[fix_size-2]) ? (h_r8+1) : h_r8;
h_r9_26 = (h_f9[fix_size-2]) ? (h_r9+1) : h_r9;
h_r10_26 = (h_f10[fix_size-2]) ? (h_r10+1) : h_r10;
h_r11_26 = (h_f11[fix_size-2]) ? (h_r11+1) : h_r11;
h_r12_26 = (h_f12[fix_size-2]) ? (h_r12+1) : h_r12;
h_r13_26 = (h_f13[fix_size-2]) ? (h_r13+1) : h_r13;
h_r14_26 = (h_f14[fix_size-2]) ? (h_r14+1) : h_r14;
h_r15_26 = (h_f15[fix_size-2]) ? (h_r15+1) : h_r15;
h_r16_26 = (h_f16[fix_size-2]) ? (h_r16+1) : h_r16;
h_r17_26 = (h_f17[fix_size-2]) ? (h_r17+1) : h_r17;
h_r18_26 = (h_f18[fix_size-2]) ? (h_r18+1) : h_r18;
h_r19_26 = (h_f19[fix_size-2]) ? (h_r19+1) : h_r19;
h_r20_26 = (h_f20[fix_size-2]) ? (h_r20+1) : h_r20;
h_r21_26 = (h_f21[fix_size-2]) ? (h_r21+1) : h_r21;
h_r22_26 = (h_f22[fix_size-2]) ? (h_r22+1) : h_r22;
h_r23_26 = (h_f23[fix_size-2]) ? (h_r23+1) : h_r23;
h_r24_26 = (h_f24[fix_size-2]) ? (h_r24+1) : h_r24;

```

```

a1_input_26 = h1 + h_r1_26;
a2_input_26 = h2 + h_r2_26;
a3_input_26 = h3 + h_r3_26;
a4_input_26 = h4 + h_r4_26;
a5_input_26 = h5 + h_r5_26;
a6_input_26 = h6 + h_r6_26;
a7_input_26 = h7 + h_r7_26;
a8_input_26 = h8 + h_r8_26;
a9_input_26 = h9 + h_r9_26;
a10_input_26 = h10 + h_r10_26;
a11_input_26 = h11 + h_r11_26;
a12_input_26 = h12 + h_r12_26;
a13_input_26 = h13 + h_r13_26;
a14_input_26 = h14 + h_r14_26;
a15_input_26 = h15 + h_r15_26;
a16_input_26 = h16 + h_r16_26;
a17_input_26 = h17 + h_r17_26;
a18_input_26 = h18 + h_r18_26;
a19_input_26 = h19 + h_r19_26;
a20_input_26 = h20 + h_r20_26;
a21_input_26 = h21 + h_r21_26;
a22_input_26 = h22 + h_r22_26;
a23_input_26 = h23 + h_r23_26;
a24_input_26 = h24 + h_r24_26;

```

end

always @(posedge clock) begin

```
if(reset)begin
    addr <= 1;

    x1  <= 0;
    x2  <= 0;
    x3  <= 0;
    x4  <= 0;
    x5  <= 0;
    x6  <= 0;
    x7  <= 0;
    x8  <= 0;
    x9  <= 0;
    x10 <= 0;
    x11 <= 0;
    x12 <= 0;
    x13 <= 0;
    x14 <= 0;
    x15 <= 0;
    x16 <= 0;
    x17 <= 0;
    x18 <= 0;
    x19 <= 0;
    x20 <= 0;
    x21 <= 0;
    x22 <= 0;
    x23 <= 0;
    x24 <= 0;

    h1  <= 0;
    h2  <= 0;
    h3  <= 0;
    h4  <= 0;
    h5  <= 0;
    h6  <= 0;
    h7  <= 0;
    h8  <= 0;
    h9  <= 0;
    h10 <= 0;
    h11 <= 0;
    h12 <= 0;
    h13 <= 0;
    h14 <= 0;
    h15 <= 0;
    h16 <= 0;
    h17 <= 0;
    h18 <= 0;
    h19 <= 0;
    h20 <= 0;
    h21 <= 0;
    h22 <= 0;
    h23 <= 0;
```

```

    h24 <= 0;

    h_1 <= 0;
    h_2 <= 0;
    h_3 <= 0;
    h_4 <= 0;
    h_5 <= 0;
    h_6 <= 0;
    h_7 <= 0;
    h_8 <= 0;
    h_9 <= 0;
    h_26<= 0;
    h_11<= 0;
    h_12<= 0;
    h_13<= 0;
    h_14<= 0;
    h_15<= 0;
    h_16<= 0;
    h_17<= 0;
    h_18<= 0;
    h_19<= 0;
    h_20<= 0;
    h_21<= 0;
    h_22<= 0;
    h_23<= 0;
    h_24<= 0;

    error_26 <= 0;
    desired_26 <= 0;
    y_26 <= 0;
end
else begin
    addr <= addr+1;
    x1  <= x_input_26;
    x2  <= x1 ;
    x3  <= x2 ;
    x4  <= x3 ;
    x5  <= x4 ;
    x6  <= x5 ;
    x7  <= x6 ;
    x8  <= x7 ;
    x9  <= x8 ;
    x10 <= x9 ;
    x11 <= x10;
    x12 <= x11;
    x13 <= x12;
    x14 <= x13;
    x15 <= x14;
    x16 <= x15;
    x17 <= x16;
    x18 <= x17;
    x19 <= x18;

```

```

x20 <= x19;
x21 <= x20;
x22 <= x21;
x23 <= x22;
x24 <= x23;

h_1 <= a1_input_26;
h_2 <= a2_input_26;
h_3 <= a3_input_26;
h_4 <= a4_input_26;
h_5 <= a5_input_26;
h_6 <= a6_input_26;
h_7 <= a7_input_26;
h_8 <= a8_input_26;
h_9 <= a9_input_26;
h_10 <= a10_input_26;
h_11 <= a11_input_26;
h_12 <= a12_input_26;
h_13 <= a13_input_26;
h_14 <= a14_input_26;
h_15 <= a15_input_26;
h_16 <= a16_input_26;
h_17 <= a17_input_26;
h_18 <= a18_input_26;
h_19 <= a19_input_26;
h_20 <= a20_input_26;
h_21 <= a21_input_26;
h_22 <= a22_input_26;
h_23 <= a23_input_26;
h_24 <= a24_input_26;

h1 <= h_1;
h2 <= h_2;
h3 <= h_3;
h4 <= h_4;
h5 <= h_5;
h6 <= h_6;
h7 <= h_7;
h8 <= h_8;
h9 <= h_9;
h10<= h_26;
h11<= h_11;
h12<= h_12;
h13<= h_13;
h14<= h_14;
h15<= h_15;
h16<= h_16;
h17<= h_17;
h18<= h_18;
h19<= h_19;
h20<= h_20;
h21<= h_21;

```

```

    h22<= h_22;
    h23<= h_23;
    h24<= h_24;

    error_26 <= error1_26;
    desired_26 <= d;
end
end
endmodule

```

## Verilog Code for Parallel LMS

```

timescale 1ns/1ps
module lms_26 (clock,reset);

input clock,reset;

reg [9:0]addr = 0;
wire signed[fix_size-1:0] x_input_26;

reg signed[fix_size-1:0] h1,h2,h3,h4,h5,h6,h7,h8,h9,h10,h11,h12
    ,h13,h14,h15,h16,h17,h18,h19,h20,h21,h22,h23,h24
    ,h_1,h_2,h_3,h_4,h_5,h_6,h_7,h_8,h_9,h_26,h_11,h_12
    ,h_13,h_14,h_15,h_16,h_17,h_18,h_19,h_20,h_21,h_22,h_23,h_24;

reg signed[(2*fix_size)-1:0] h_d1,h_d2,h_d3,h_d4,h_d5,h_d6,h_d7,h_d8,h_d9,h_d10,h_d11,h_d12
    ,h_d13,h_d14,h_d15,h_d16,h_d17,h_d18,h_d19,h_d20,h_d21,h_d22,h_d23,h_d24;

reg signed[(2*fix_size)-1:0] h_f1,h_f2,h_f3,h_f4,h_f5,h_f6,h_f7,h_f8,h_f9,h_f10,h_f11,h_f12
    ,h_f13,h_f14,h_f15,h_f16,h_f17,h_f18,h_f19,h_f20,h_f21,h_f22,h_f23,h_f24;

reg signed[fix_size-1:0] h_r1,h_r2,h_r3,h_r4,h_r5,h_r6,h_r7,h_r8,h_r9,h_r10,h_r11,h_r12
    ,h_r13,h_r14,h_r15,h_r16,h_r17,h_r18,h_r19,h_r20,h_r21,h_r22,h_r23,h_r24;

reg signed[fix_size-1:0]
h_r1_26,h_r2_26,h_r3_26,h_r4_26,h_r5_26,h_r6_26,h_r7_26,h_r8_26,h_r9_26,h_r10_26,h_r11_26,h_r
12_26
    ,h_r13_26,h_r14_26,h_r15_26,h_r16_26,h_r17_26,h_r18_26,h_r19_26,h_r20_26,h_r21_26,h_r
22_26,h_r23_26,h_r24_26;

reg signed[fix_size-1:0] x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x12,
    x13,x14,x15,x16,x17,x18,x19,x20,x21,x22,x23,x24;

reg signed[(2*fix_size)-1:0] y1,y2,y3,y4,y5,y6,y7,y8,y9,y10,y11,y12
    ,y13,y14,y15,y16,y17,y18,y19,y20,y21,y22,y23,y24;

```

```
reg signed[fix_size-1:0] y_d1,y_d2,y_d3,y_d4,y_d5,y_d6,y_d7,y_d8,y_d9,y_d10,y_d11,y_d12
,y_d13,y_d14,y_d15,y_d16,y_d17,y_d18,y_d19,y_d20,y_d21,y_d22,y_d23,y_d24;
```

```
reg signed[(2*fix_size)-1:0] y_r1,y_r2,y_r3,y_r4,y_r5,y_r6,y_r7,y_r8,y_r9,y_r10,y_r11,y_r12
,y_r13,y_r14,y_r15,y_r16,y_r17,y_r18,y_r19,y_r20,y_r21,y_r22,y_r23,y_r24;
reg signed[fix_size-1:0] y_f,y_f1;
```

```
reg signed[fix_size-1:0] a1_input_26, a2_input_26, a3_input_26, a4_input_26
, a5_input_26, a6_input_26, a7_input_26, a8_input_26
, a9_input_26, a10_input_26, a11_input_26,
a12input_26, a13_input_26, a14_input_26, a15_input_26,
a16_input_26, a17_input_26, a18_input_26, a19_input_26,
a20_input_26, a21_input_26, a22_input_26, a23_input_26, a24_input_26;
```

```
reg signed[(2*fix_size)-1:0] y_26;
```

```
reg signed[fix_size-1:0] desired_26;
reg signed[fix_size-1:0] error_26,error1_26;
wire signed[fix_size-1:0] d;
parameter fix_size=32;
```

```
lms_ram_26 ram_x(.address(addr),.clock(clock),.data(0),.wren(1'b0),.q(x_input_26));
lms_ramd_26 ram_d(.address(addr),.clock(clock),.data(0),.wren(1'b0),.q(d));
```

```
always@(*)begin
    y_f = 0;
    h_d1 = error_26*x_input_26;
    h_f1 = (h_d1[1])?((h_d1>>>2) + 1):(h_d1>>>2);
    h_r1 = h_f1[2*fix_size-2:fix_size-1];
    h_r1_26 = (h_f1[fix_size-2]) ? (h_r1+1) : h_r1;
    a1_input_26 = h1 + h_r1_26;
```

```
////////////////////////////////////
    y1 = a1_input_26*x_input_26;
    y_d1 = y1[2*fix_size-2:fix_size-1];
    y_r1 = (y1[fix_size-2])?(y_d1+1'b1):y_d1;
    y_f = y_f + y_r1;////////////////////////////////
    h_d2 = error_26*x1;
    h_f2 = (h_d2[1])?((h_d2>>>2) + 1):(h_d2>>>2);
    h_r2 = h_f2[2*fix_size-2:fix_size-1];
    h_r2_26 = (h_f2[fix_size-2]) ? (h_r2+1) : h_r2;
    a2_input_26 = h2 + h_r2_26;
```

```
////////////////////////////////////
    y2 = a2_input_26*x1;
    y_d2 = y2[2*fix_size-2:fix_size-1];
    y_r2 = (y2[fix_size-2])?(y_d2+1'b1):y_d2;
```

```

y_f = y_f + y_r2;
h_d3 = error_26*x2;
h_f3 = (h_d3[1])?((h_d3>>>2) + 1):(h_d3>>>2);
h_r3 = h_f3[2*fix_size-2:fix_size-1];
h_r3_26 = (h_f3[fix_size-2]) ? (h_r3+1) : h_r3;
a3_input_26 = h3 + h_r3_26;

```

```

////////////////////////////////////

```

```

y3 = a3_input_26*x2;
y_d3 = y3[2*fix_size-2:fix_size-1];
y_r3 = (y3[fix_size-2])?(y_d3+1'b1):y_d3;
y_f = y_f + y_r3;
h_d4 = error_26*x3;
h_f4 = (h_d4[1])?((h_d4>>>2) + 1):(h_d4>>>2);
h_r4 = h_f4[2*fix_size-2:fix_size-1];
h_r4_26 = (h_f4[fix_size-2]) ? (h_r4+1) : h_r4;
a4_input_26 = h4 + h_r4_26;

```

```

////////////////////////////////////

```

```

y4 = a4_input_26*x3;
y_d4 = y4[2*fix_size-2:fix_size-1];
y_r4 = (y4[fix_size-2])?(y_d4+1'b1):y_d4;
y_f = y_f + y_r4;
h_d5 = error_26*x4;
h_f5 = (h_d5[1])?((h_d5>>>2) + 1):(h_d5>>>2);
h_r5 = h_f5[2*fix_size-2:fix_size-1];
h_r5_26 = (h_f5[fix_size-2]) ? (h_r5+1) : h_r5;
a5_input_26 = h5 + h_r5_26;

```

```

////////////////////////////////////

```

```

y5 = a5_input_26*x4;
y_d5 = y5[2*fix_size-2:fix_size-1];
y_r5 = (y5[fix_size-2])?(y_d5+1'b1):y_d5;
y_f = y_f + y_r5;
h_d6 = error_26*x5;
h_f6 = (h_d6[1])?((h_d6>>>2) + 1):(h_d6>>>2);
h_r6 = h_f6[2*fix_size-2:fix_size-1];
h_r6_26 = (h_f6[fix_size-2]) ? (h_r6+1) : h_r6;
a6_input_26 = h6 + h_r6_26;

```

```

////////////////////////////////////

```

```

y6 = a6_input_26*x5;
y_d6 = y6[2*fix_size-2:fix_size-1];
y_r6 = (y6[fix_size-2])?(y_d6+1'b1):y_d6;
y_f = y_f + y_r6;
h_d7 = error_26*x6;
h_f7 = (h_d7[1])?((h_d7>>>2) + 1):(h_d7>>>2);
h_r7 = h_f7[2*fix_size-2:fix_size-1];
h_r7_26 = (h_f7[fix_size-2]) ? (h_r7+1) : h_r7;
a7_input_26 = h7 + h_r7_26;

```



```

////////////////////////////////////
    y7 = a7_input_26*x6;
    y_d7 = y7[2*fix_size-2:fix_size-1];
    y_r7 = (y7[fix_size-2])?(y_d7+1'b1):y_d7;
    y_f = y_f + y_r7;
    h_d8 = error_26*x7;
    h_f8 = (h_d8[1])?((h_d8>>>2) + 1):(h_d8>>>2);
    h_r8 = h_f8[2*fix_size-2:fix_size-1];
    h_r8_26 = (h_f8[fix_size-2]) ? (h_r8+1) : h_r8;
    a8_input_26 = h8 + h_r8_26;
////////////////////////////////////
    y8 = a8_input_26*x7;
    y_d8 = y8[2*fix_size-2:fix_size-1];
    y_r8 = (y8[fix_size-2])?(y_d8+1'b1):y_d8;
    y_f = y_f + y_r8;
    h_d9 = error_26*x8;
    h_f9 = (h_d8[1])?((h_d8>>>2) + 1):(h_d8>>>2);
    h_r9 = h_f8[2*fix_size-2:fix_size-1];
    h_r9_26 = (h_f8[fix_size-2]) ? (h_r8+1) : h_r8;
    a9_input_26 = h8 + h_r8_26;
////////////////////////////////////
    y9 = a9_input_26*x8;
    y_d9 = y9[2*fix_size-2:fix_size-1];
    y_r9 = (y9[fix_size-2])?(y_d9+1'b1):y_d9;
    y_f = y_f + y_r9;
    h_d10 = error_26*x9;
    h_f10 = (h_d10[1])?((h_d10>>>2) + 1):(h_d10>>>2);
    h_r10 = h_f10[2*fix_size-2:fix_size-1];
    h_r10_26 = (h_f10[fix_size-2]) ? (h_r10+1) : h_r10;
    a10_input_26 = h10 + h_r10_26;
////////////////////////////////////
    y10 = a10_input_26*x9;
    y_d10 = y10[2*fix_size-2:fix_size-1];
    y_r10 = (y10[fix_size-2])?(y_d10+1'b1):y_d10;
    y_f = y_f + y_r10;
    h_d11 = error_26*x10;
    h_f11 = (h_d11[1])?((h_d11>>>2) + 1):(h_d11>>>2);
    h_r11 = h_f11[2*fix_size-2:fix_size-1];
    h_r11_26 = (h_f11[fix_size-2]) ? (h_r11+1) : h_r11;
    a11_input_26 = h11 + h_r11_26;
////////////////////////////////////
    y11 = a11_input_26*x10;
    y_d11 = y11[2*fix_size-2:fix_size-1];
    y_r11 = (y11[fix_size-2])?(y_d11+1'b1):y_d11;
    y_f = y_f + y_r11;
    h_d12 = error_26*x11;
    h_f12 = (h_d12[1])?((h_d12>>>2) + 1):(h_d12>>>2);
    h_r12 = h_f12[2*fix_size-2:fix_size-1];

```

```

h_r12_26 = (h_f12[fix_size-2]) ? (h_r12+1) : h_r12;
a12_input_26 = h12 + h_r12_26;
////////////////////////////////////
y12 = a12_input_26*x11;
y_d12 = y12[2*fix_size-2:fix_size-1];
y_r12 = (y12[fix_size-2])?(y_d12+1'b1):y_d12;
y_f = y_f + y_r12;
h_d13 = error_26*x12;
h_f13 = (h_d13[1])?((h_d13>>>2) + 1):(h_d13>>>2);
h_r13 = h_f13[2*fix_size-2:fix_size-1];
h_r13_26 = (h_f13[fix_size-2]) ? (h_r13+1) : h_r13;
a13_input_26 = h13 + h_r13_26;
////////////////////////////////////
y13 = a13_input_26*x12;
y_d13= y13[2*fix_size-2:fix_size-1];
y_r13 = (y13[fix_size-2])?(y_d13+1'b1):y_d13;
y_f = y_f + y_r13;
h_d14 = error_26*x13;
h_f14 = (h_d14[1])?((h_d14>>>2) + 1):(h_d14>>>2);
h_r14 = h_f14[2*fix_size-2:fix_size-1];
h_r14_26 = (h_f14[fix_size-2]) ? (h_r14+1) : h_r14;
a14_input_26 = h14 + h_r14_26;
////////////////////////////////////
y14 = a14_input_26*x13;
y_d14 = y14[2*fix_size-2:fix_size-1];
y_r14 = (y14[fix_size-2])?(y_d14+1'b1):y_d14;
y_f = y_f + y_r14;
h_d15 = error_26*x14;
h_f15 = (h_d15[1])?((h_d15>>>2) + 1):(h_d15>>>2);
h_r15 = h_f15[2*fix_size-2:fix_size-1];
h_r15_26 = (h_f15[fix_size-2]) ? (h_r15+1) : h_r15;
a15_input_26 = h15 + h_r15_26;
////////////////////////////////////
y15 = a15_input_26*x14;
y_d15 = y15[2*fix_size-2:fix_size-1];
y_r15 = (y15[fix_size-2])?(y_d15+1'b1):y_d15;
y_f = y_f + y_r15;
h_d16 = error_26*x15;
h_f16 = (h_d16[1])?((h_d16>>>2) + 1):(h_d16>>>2);
h_r16 = h_f16[2*fix_size-2:fix_size-1];
h_r16_26 = (h_f16[fix_size-2]) ? (h_r16+1) : h_r16;
a16_input_26 = h16 + h_r16_26;
////////////////////////////////////
y16 = a16_input_26*x15;
y_d16 = y16[2*fix_size-2:fix_size-1];
y_r16 = (y16[fix_size-2])?(y_d16+1'b1):y_d16;
y_f = y_f + y_r16;
h_d17 = error_26*x16;

```

```

h_f17 = (h_d17[1])?((h_d17>>>2) + 1):(h_d17>>>2);
h_r17 = h_f17[2*fix_size-2:fix_size-1];
h_r17_26 = (h_f17[fix_size-2]) ? (h_r17+1) : h_r17;
a17_input_26 = h17 + h_r17_26;

////////////////////////////////////
y17 = a17_input_26*x16;
y_d17 = y17[2*fix_size-2:fix_size-1];
y_r17 = (y17[fix_size-2])?(y_d17+1'b1):y_d17;
y_f = y_f + y_r17;
h_d18 = error_26*x17;
h_f18 = (h_d18[1])?((h_d18>>>2) + 1):(h_d18>>>2);
h_r18 = h_f18[2*fix_size-2:fix_size-1];
h_r18_26 = (h_f18[fix_size-2]) ? (h_r18+1) : h_r18;
a18_input_26 = h18 + h_r18_26;

////////////////////////////////////
y18 = a18_input_26*x17;
y_d18 = y18[2*fix_size-2:fix_size-1];
y_r18 = (y18[fix_size-2])?(y_d18+1'b1):y_d18;
y_f = y_f + y_r18;
h_d19 = error_26*x18;
h_f19 = (h_d19[1])?((h_d19>>>2) + 1):(h_d19>>>2);
h_r19 = h_f19[2*fix_size-2:fix_size-1];
h_r19_26 = (h_f19[fix_size-2]) ? (h_r19+1) : h_r19;
a19_input_26 = h19 + h_r19_26;

////////////////////////////////////
y19 = a19_input_26*x18;
y_d19 = y19[2*fix_size-2:fix_size-1];
y_r19 = (y19[fix_size-2])?(y_d19+1'b1):y_d19;
y_f = y_f + y_r19;
h_d20 = error_26*x19;
h_f20 = (h_d20[1])?((h_d20>>>2) + 1):(h_d20>>>2);
h_r20 = h_f20[2*fix_size-2:fix_size-1];
h_r20_26 = (h_f20[fix_size-2]) ? (h_r20+1) : h_r20;
a20_input_26 = h20 + h_r20_26;

////////////////////////////////////
y20 = a20_input_26*x19;
y_d20 = y20[2*fix_size-2:fix_size-1];
y_r20 = (y20[fix_size-2])?(y_d20+1'b1):y_d20;
y_f = y_f + y_r20;
h_d21 = error_26*x20;
h_f21 = (h_d21[1])?((h_d21>>>2) + 1):(h_d21>>>2);
h_r21 = h_f21[2*fix_size-2:fix_size-1];
h_r21_26 = (h_f21[fix_size-2]) ? (h_r21+1) : h_r21;
a21_input_26 = h21 + h_r21_26;

////////////////////////////////////
y21 = a21_input_26*x20;
y_d21 = y21[2*fix_size-2:fix_size-1];

```

```

        y_r21 = (y21[fix_size-2])?(y_d21+1'b1):y_d21;
        y_f = y_f + y_r21;
        h_d22 = error_26*x21;
        h_f22 = (h_d22[1])?((h_d22>>>2) + 1):(h_d22>>>2);
        h_r22 = h_f22[2*fix_size-2:fix_size-1];
        h_r22_26 = (h_f11[fix_size-2]) ? (h_r22+1) : h_r22;
        a22_input_26 = h22 + h_r22_26;
//////////
        y22 = a22_input_26*x21;
        y_d22 = y22[2*fix_size-2:fix_size-1];
        y_r22 = (y22[fix_size-2])?(y_d22+1'b1):y_d22;
        y_f = y_f + y_r22;
        h_d23 = error_26*x22;
        h_f23 = (h_d23[1])?((h_d23>>>2) + 1):(h_d23>>>2);
        h_r23 = h_f23[2*fix_size-2:fix_size-1];
        h_r23_26 = (h_f23[fix_size-2]) ? (h_r23+1) : h_r23;
        a23_input_26 = h23 + h_r23_26;
//////////
        y23 = a23_input_26*x22;
        y_d23 = y23[2*fix_size-2:fix_size-1];
        y_r23 = (y23[fix_size-2])?(y_d23+1'b1):y_d23;
        y_f = y_f + y_r23;
        h_d24 = error_26*x23;
        h_f24 = (h_d24[1])?((h_d24>>>2) + 1):(h_d24>>>2);
        h_r24 = h_f24[2*fix_size-2:fix_size-1];
        h_r24_26 = (h_f24[fix_size-2]) ? (h_r24+1) : h_r24;
        a24_input_26 = h24 + h_r24_26;

//////////
        y24 = a24_input_26*x23;
        y_d24 = y24[2*fix_size-2:fix_size-1];
        y_r24 = (y24[fix_size-2])?(y_d24+1'b1):y_d24;
        y_f = y_f + y_r24;

        error1_26 = desired_26 - y_f;
//////////

end

always @(posedge clock) begin
if(reset)begin
    addr <= 1;
    y_f1 <= 0;
    x1 <= 0;
    x2 <= 0;
    x3 <= 0;
    x4 <= 0;

```

```
x5 <= 0;  
x6 <= 0;  
x7 <= 0;  
x8 <= 0;  
x9 <= 0;  
x10 <= 0;  
x11 <= 0;  
x12 <= 0;  
x13 <= 0;  
x14 <= 0;  
x15 <= 0;  
x16 <= 0;  
x17 <= 0;  
x18 <= 0;  
x19 <= 0;  
x20 <= 0;  
x21 <= 0;  
x22 <= 0;  
x23 <= 0;  
x24 <= 0;
```

```
h1 <= 0;  
h2 <= 0;  
h3 <= 0;  
h4 <= 0;  
h5 <= 0;  
h6 <= 0;  
h7 <= 0;  
h8 <= 0;  
h9 <= 0;  
h10 <= 0;  
h11 <= 0;  
h12 <= 0;  
h13 <= 0;  
h14 <= 0;  
h15 <= 0;  
h16 <= 0;  
h17 <= 0;  
h18 <= 0;  
h19 <= 0;  
h20 <= 0;  
h21 <= 0;  
h22 <= 0;  
h23 <= 0;  
h24 <= 0;
```

```
error_26 <= 0;
```

```
        desired_26 <= 0;
        y_26 <= 0;
    end
    else begin
        y_f1 <= y_f;
        addr <= addr+1;
        x1 <= x_input_26;
        x2 <= x1 ;
        x3 <= x2 ;
        x4 <= x3 ;
        x5 <= x4 ;
        x6 <= x5 ;
        x7 <= x6 ;
        x8 <= x7 ;
        x9 <= x8 ;
        x10 <= x9 ;
        x11 <= x10;
        x12 <= x11;
        x13 <= x12;
        x14 <= x13;
        x15 <= x14;
        x16 <= x15;
        x17 <= x16;
        x18 <= x17;
        x19 <= x18;
        x20 <= x19;
        x21 <= x20;
        x22 <= x21;
        x23 <= x22;
        x24 <= x23;

        h1 <= a1_input_26;
        h2 <= a2_input_26;
        h3 <= a3_input_26;
        h4 <= a4_input_26;
        h5 <= a5_input_26;
        h6 <= a6_input_26;
        h7 <= a7_input_26;
        h8 <= a8_input_26;
        h9 <= a9_input_26;
        h10 <= a10_input_26;
        h11 <= a11_input_26;
        h12 <= a12_input_26;
        h13 <= a13_input_26;
        h14 <= a14_input_26;
        h15 <= a15_input_26;
        h16 <= a16_input_26;
        h17 <= a17_input_26;
```

```
    h18 <= a18_input_26;
    h19 <= a19_input_26;
    h20 <= a20_input_26;
    h21 <= a21_input_26;
    h22 <= a22_input_26;
    h23 <= a23_input_26;
    h24 <= a24_input_26;

    error_26 <= error1_26;
    desired_26 <= d;
end
end
endmodule
```

### **Testbench**

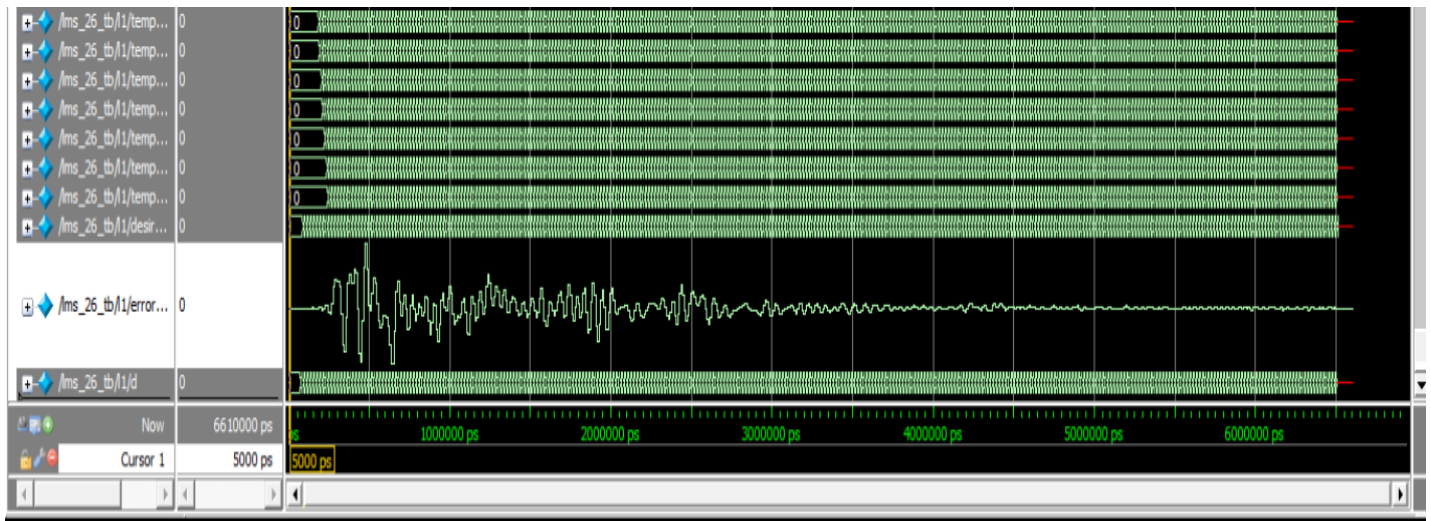
```
`timescale 1ns/1ps
module lms_26_tb();
reg clock, reset;

lms_26 l1(clock, reset);

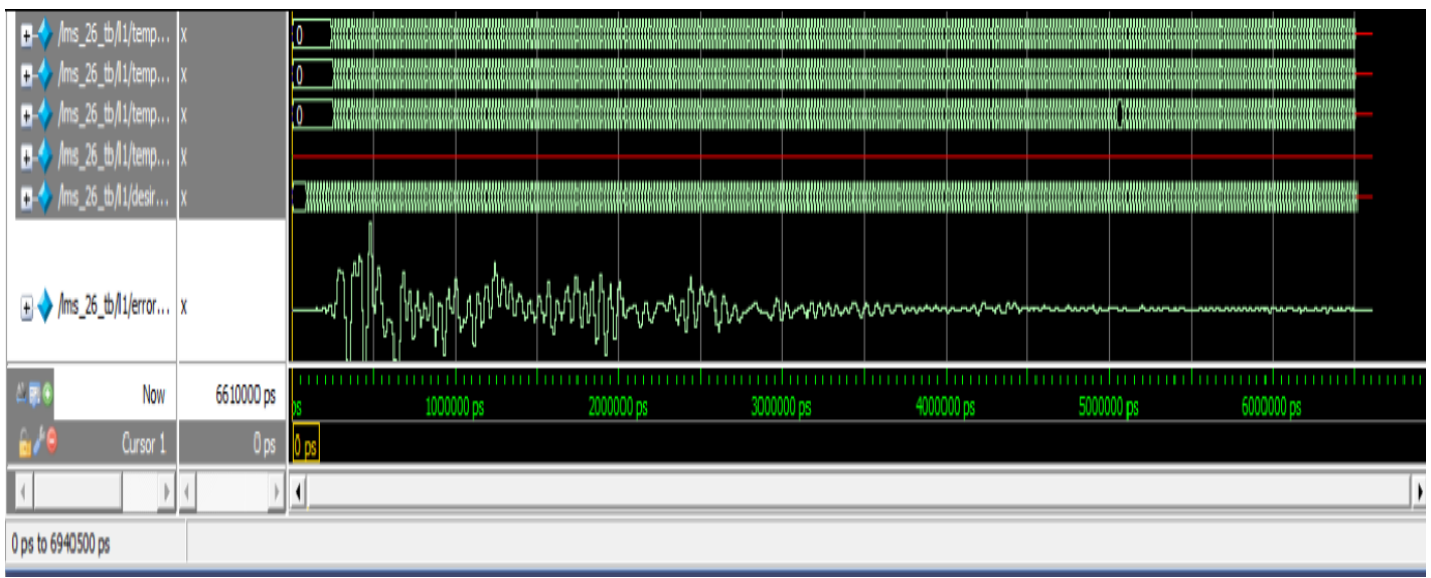
always #5 clock=~clock;
initial begin
reset=1'b0;
clock=1'b0;
#5
reset=1'b1;
#5
reset=1'b0;

#6500
$finish;
end
endmodule
```

## Output Waveforms

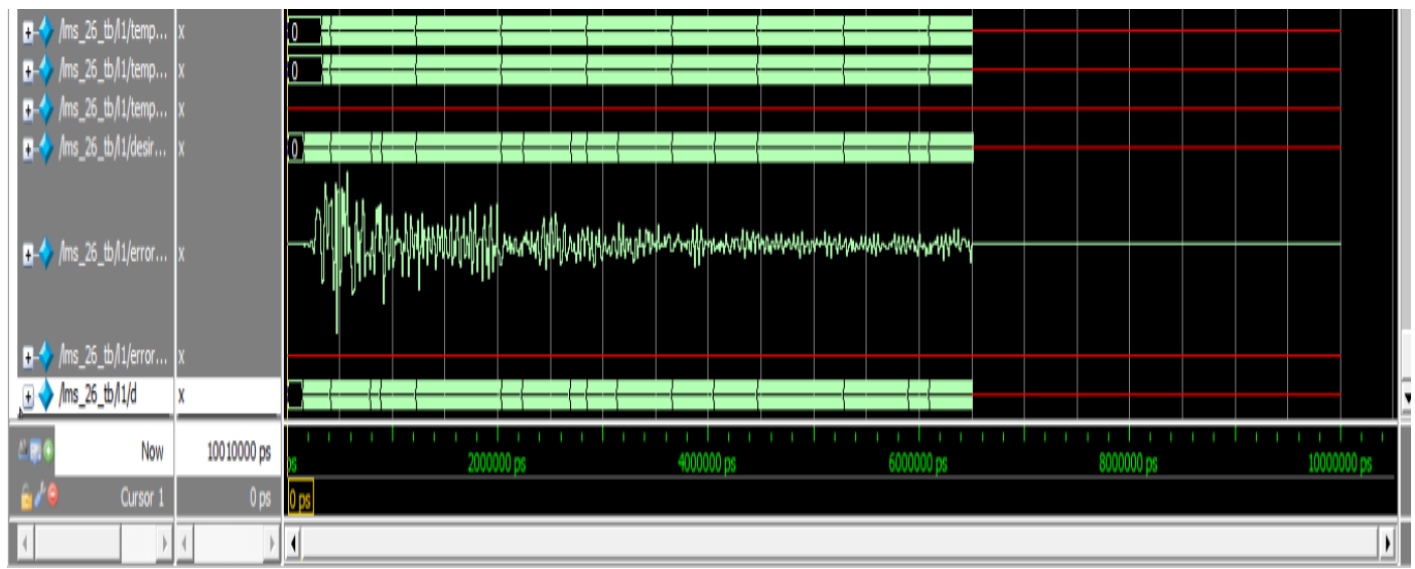


(Fig.4: output for 32 bit sequential LMS)

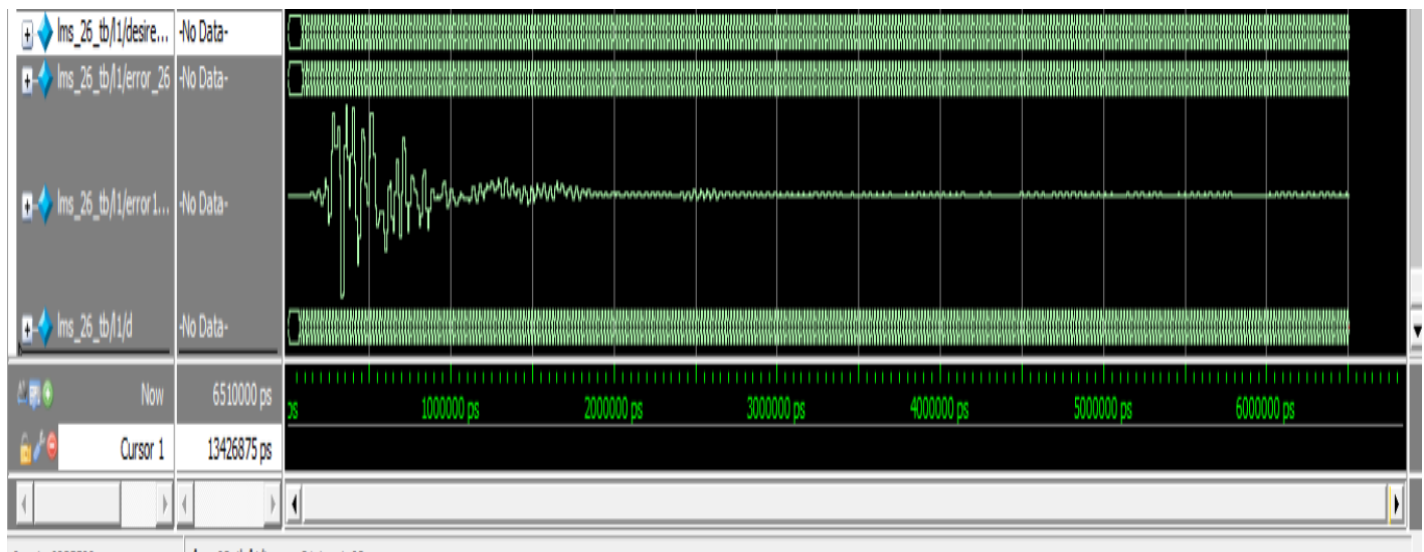


(Fig.5: Output for 16 bit sequential LMS)

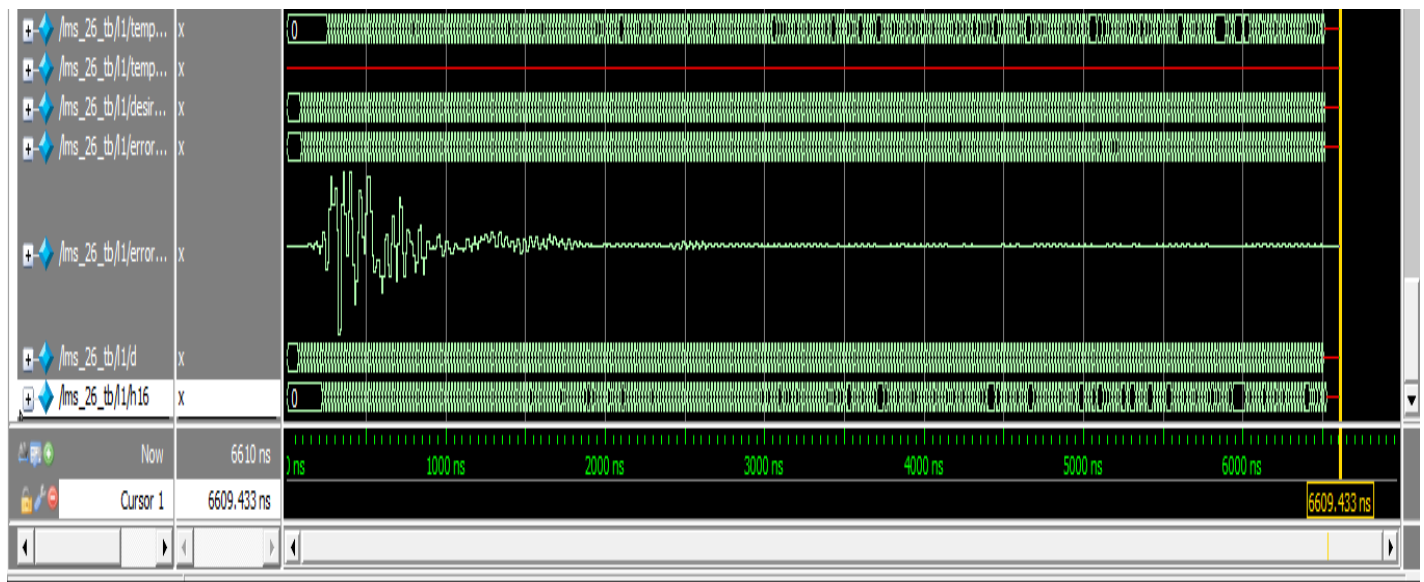




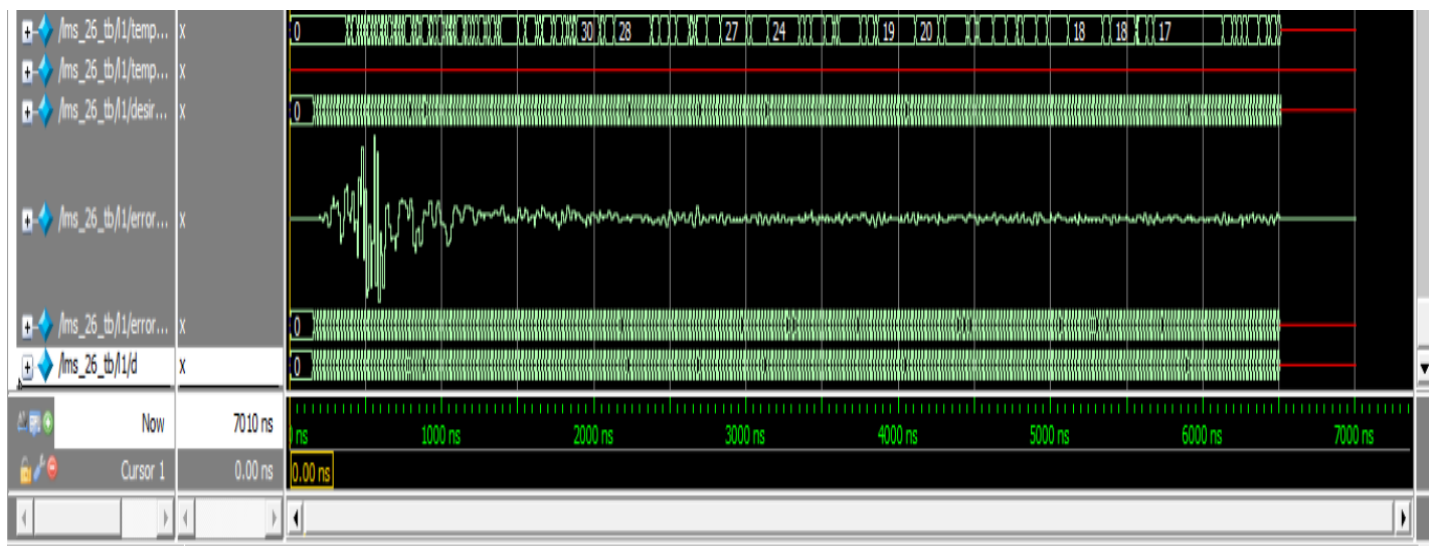
(Fig.6: Output for 8 bit sequential LMS)



(Fig.7: Output for 32 bit parallel LMS)



(Fig.8: Output for 16 bit parallel LMS)



(Fig.9: Output for 8 bit parallel LMS)