

```
function[state_out,y] = pll(state, a)
% Executes the phase locked loop .
% Inputs:
%   state      Current/Input state
%   a          input function
% Outputs:
%   state_out  Output state variables to be stored for next iteration
%% Get state
s = state;
z = zeros (size(a));
v = zeros (size(a));
y = zeros (size(a));
x = a;
amp = 0;

%% obtain the output of LPF
for n= 1:length(a)
    amp = amp + abs(x(n));
    z(n) = s.ym1*x(n)/s.amp_est;
    v(n) = s.vm1*s.a1 + z(n)*s.b0 + s.b1*s.zm1 ;
    s.zm1 = z(n) ;
    s.vm1 = v(n) ;
    % accumulator
    s.acc = s.acc + s.f - (s.k/(2*pi))*v(n) ;
    s.acc = s.acc - floor(s.acc);
    %sine table
    y(n) = s.sin_table(floor(1024*s.acc)+1);

    %
    s.ym1 = y(n);
    s.zm1 = z(n);
    s.xm1 = x(n);
end
s.amp_est = amp/length(x)/(2/pi);
%% Return updated state
state_out = s;
```