

```

bus_new = bus;
if ~isempty(svc_con)
    if flag == 0;
        svc_pot(:,1) = svc_con(:,4).*svc_con(:,3)/basma;
        % B_cv max on system base
        svc_pot(:,2) = svc_con(:,5).*svc_con(:,3)/basma;
        % B_cv min on system base
        jsvc = bus_int(svc_con(:,2)); % bus number
        B_cv(:,1) = bus(jsvc,5)/(bus(jsvc,2).*bus(jsvc,2)); % initial B_cv
        bus_new(jsvc,5) = zeros(n_svc,1);
        testmxlmt=max(B_cv(:,1) > svc_pot(:,1)); % B_cv > svc_pot(:,1)
        if testmxlmt
            error('SVC: BCV exceeds maximum at initialization')
        end
        testmnlmt=max(B_cv(:,1) < svc_pot(:,2)); % B_cv < svc_pot(:,2)
        if testmnlmt
            error('SVC: BCV below minimum at initialization')
        end
        svc_pot(:,3) = B_cv(:,1); % store initial value of B_cv
        svc_pot(:,4) = bus(jsvc,2) + B_cv(:,1)./svc_con(:,6); % reference voltage
        svc_pot(:,5) = ones(n_svc,1);
        if ~isempty(svc11_idx)
            svc_pot(svc11_idx,5) = svc_con(svc11_idx,8)./svc_con(svc11_idx,9);
        end
        B_con(:,1) = B_cv(:,1).*(ones(n_svc,1)-svc_pot(:,5))./svc_con(:,6);
    end
end
if flag == 1 % network interface computation
    % no interface calculation required - done in nc_load
end

if flag == 2 % exciter dynamics calculation
    % for linearization with operating condition at limits,
    % additional code will be needed
    lv_sbus=find(v_sbus<0.9 & svc_dsig(:,k)<0);
    d_sigin = svc_dsig(:,k);
    if ~isempty(lv_sbus)
        d_sigin(lv_sbus)=zeros(length(lv_sbus),1);
    end
    err = svc_sig(:,k) + svc_pot(:,4) + d_sigin - v_sbus;

    dB_con(:,k)= zeros(n_svc,1);
    if ~isempty(svc11_idx)
        n11 = length(svc11_idx);
        dB_con(svc11_idx,k) = (-B_con(svc11_idx,k)+(ones(n11,1)-svc_pot(svc11_idx,5)).*err)./svc_con(svc11_idx,9);
    end
    dB_cv(:,k) = (-B_cv(:,k)+svc_con(:,6).*(err.*svc_pot(:,5)+B_con(:,k)))./svc_con(:,7);
    % anti-windup reset
    indmx=find(B_cv(:,k) > svc_pot(:,1));
    if ~isempty(indmx)
        B_cv(indmx,k) = svc_pot(indmx,1);
        indrate = find(dB_cv(indmx,k)>0);
        if ~isempty(indrate)
            % set rate to zero
            dB_cv(indmx(indrate),k) = zeros(length(indrate),1);
        end
    end
    indmn = find(B_cv(:,k) < svc_pot(:,2));
    if ~isempty(indmn)
        B_cv(indmn,k) = svc_pot(indmn,2);
        indrate = find(dB_cv(indmn,k)<0);
        if ~isempty(indrate)
            % set rate to zero
            dB_cv(indmn(indrate),k) = zeros(length(indrate),1);
        end
    end
end
end
end
end

```