

Transient and Frequency Response Simulation of Second-order Notch Filter Switched-Capacitor Circuit (Biphase)

(Prepared by: Rajesh A. Thakker, Govt. Engg. College, Chandkheda, Gandhinagar,
Email id: rathakker2008@gmail.com)

Circuit file: sc_notch.in

Fig.1 shows the 2nd-order notch switched capacitor network with all parameters. This circuit is simulated here in both domains, Frequency and Time. It is viewed as a connection of many commonly found SC blocks, namely toggle switched inverter (sc_tsi_1), switched resistor (sc_res_1), floating capacitor (sc_float_cap) and switched-capacitor opamp (sc_opamp1).

One such block is shown in Fig. 2 with its Time-domain (a) and Frequency-domain (b) equivalent circuits. As shown in the Fig.2, time-domain circuit is two-port (don't consider the nodes connected with low-value resistance (Rdl)), where as its frequency-domain (z-domain) equivalent circuit consists of four ports. Unused ports or nodes are connected with low-value resistance (Rdl) in Time-domain and High-value resistance (Rdh) in frequency-domain to avoid matrix to become singular.

All commonly found blocks are described as compound elements in SEQUEL with separate time-domain and frequency-domain description. Frequency-domain description assumes ideal elements and equal phase intervals. Time-domain description permits study of non-ideal elements and unequal phase intervals.

The simulated transient and frequency domain response are shown in Fig. 3 and Fig. 4 respectively. The approach utilized here to generate frequency response for switched-capacitor networks is such that it provides frequency response (magnitude and phase responses) for even and odd phases separately. If both of them are nonzero for the frequency-range of interests then they should be combined together by adding real parts of both phase and the same for imaginary parts. Now both terms should added after squared and square root should be calculated. Now it could be converted to gain.

1. Circuit Block:

```
title: notch filter
```

```
greal tc=0.125m roff=1e18 ron=1m av=1e5 rinop=1e12 tcby2=62.5u
```

```
* Define a global parameter tc (clock period), roff and ron (switch off and on  
* resistances), rin (input resistance) and av (voltage gain) of op-amp and  
* tcby2.
```

```
begin_circuit
```

```
  eelement type=vsrccac p=a n=g a=1 f_hz=60 t0=0 phi=0
```

```
* This element produces sine-wave source with frequency 60 Hz.
```

```
  eelement name=srcl type=sc_source_2 vo_e=1e vo_o=1o g=g  
+ auxl=a tc=tcby2 dt=1u s_2=62.5u tau_1=62.5u tau_2=62.5u
```

```
* This element samples the sinewave generated above at one-half of clock  
* frequency for even and odd phase and it also constructs the frequency-domain
```

* representation for both even and odd phase sampled sinewave.

```
eelement type=sc_float_cap
+   vi_e=1e vi_o=1o vo_e=2e vo_o=2o g=g
+   c=1.0233p tc=tc ron=ron roff=roff
```

* This element implements floating capacitor.

```
eelement type=sc_float_cap
+   vi_e=2e vi_o=2o vo_e=3e vo_o=3o g=g
+   c=1p tc=tc ron=ron roff=roff
```

* This element implements floating capacitor.

```
eelement type=sc_res_1
+   vi_e=2e vi_o=2o vo_e=3e vo_o=3o g=g phi_e=phi_e phi_o=phi_o
+   ron=ron roff=roff c=0.04668p tc=tc
```

* This element implements switched resistor.

```
eelement type=sc_tsi_1
+   vi_e=3e vi_o=3o vo_e=4e vo_o=4o g=g phi_e=phi_e phi_o=phi_o
+   ron=ron roff=roff c=0.04688p tc=tc
```

* This element implements toggle switched inverter.

```
eelement type=sc_float_cap
+   vi_e=4e vi_o=4o vo_e=5e vo_o=5o g=g
+   c=1p tc=tc ron=ron roff=roff
```

* This element implements floating capacitor.

```
eelement type=sc_res_1
+   vi_e=5e vi_o=5o vo_e=2e vo_o=2o g=g phi_e=phi_e phi_o=phi_o
+   ron=ron roff=roff c=0.04688p tc=tc
```

* This element implements switched resistor.

```
eelement type=sc_tsi_1
+   vi_e=1e vi_o=1o vo_e=4e vo_o=4o g=g phi_e=phi_e phi_o=phi_o
+   ron=ron roff=roff c=0.04802p tc=tc
```

* This element implements toggle switched inverter.

```
eelement name=op1 type=sc_opamp1 vi_ep=g vi_em=2e vi_op=g
+   vi_om=2o vo_eo=3e vo_oo=3o gnd=g
+   rin=rinop rout=0 av=av
```

* Above statement implements switched capacitor op-amp.

* This op-amp consists of two op-amps one for each phase

```
eelement name=op2 type=sc_opamp1 vi_ep=g vi_em=4e vi_op=g
+   vi_om=4o vo_eo=5e vo_oo=5o gnd=g
+   rin=rinop rout=0 av=av
```

```

* Above statement implements switched capacitor op-amp.
* This op-amp consists of two op-amps one for each phase

delement type=clock a=phi_e i0=1 t1=62.49u t2=62.51u t0=-0.1u
delement type=clock a=phi_o i0=0 t1=62.51u t2=62.49u t0=-0.1u

* These two statements generate non-overlapping clock signal to make the
* switches on and off in even and odd phases.

refnode=g

* Node g is defined as the reference node for node voltages.

outvar:
+ va=nodev_of_a vle=nodev_of_1e vlo=nodev_of_1o
+ va_ac=nodev_ac_of_a vle_ac=nodev_ac_of_1e vlo_ac=nodev_ac_of_1o
+ v2e=nodev_of_2e v2o=nodev_of_2o
+ v2e_ac=nodev_ac_of_2e v2o_ac=nodev_ac_of_2o
+ v3e=nodev_of_3e v3o=nodev_of_3o
+ v3e_ac=nodev_ac_of_3e v3o_ac=nodev_ac_of_3o
+ v5e=nodev_of_5e v5o=nodev_of_5o
+ v5e_ac=nodev_ac_of_5e v5o_ac=nodev_ac_of_5o

* Definition of the output variables.

end_circuit

```

2. Solve Block 1:

```

begin_solve
  solve_type=startup
  initial_sol initialize
  method: t_startup=0

* Do a "start-up" solution. As the initial guess for the
* Newton-Raphson iterations, use the "initialize" option.
* Since we have not generated any solution prior to this,
* the "initialize" option has to be used.

end_solve

```

3. Solve Block 2:

```

begin_solve
  solve_type=trns
  initial_sol previous

* Perform a transient simulation, using the previous solution
* (obtained in the last solve block) as the initial guess for
* the Newton-Raphson iterations. Note that "initial_sol" is
* relevant only to get the transient simulation started (i.e.,
* the very first time step). For subsequent time steps, the
* solution obtained for the previous time step is always used
* as the initial guess for the N-R iterations.

```

```
begin_output
  filename=notch7.dat limit_lines=20000000
  variables: v3e
end_output
```

* Specify which output files are to be generated and which
* output variables are to be written to each of the files.
* Note that we write here only analog variables of interest (v3e).
* If we want to store digital variables, we should ask for one more file.
* The two types of variables cannot be written to a single file.

*
* Note that limit_lines has been specified in the above files
* since the default value (10,000) is too small. The number
* of lines which get written to the output files is larger
* than this default value, and this would cause the program
* to stop before the specified end time is reached.

```
method: norm_2=1.0e-5 itmax_trns=100000000 back_euler=yes
+   t_start=0 t_end=40m delt_const=10u delt_min=1n
```

* Specify the RHS norm ("tolerance"). The default value of
* norm_2 is 1.0e-10. However, that turns out to be too rigid
* for this problem and hence the need to change it.

* Use the backward Euler method with a constant time step of
* 10 usec.

* Specify the starting and ending times, and time step.
* Note that itmax_trns has been specified since the default
* value (10,000) is too small; the number of time points required
* for this simulation is larger than the default itmax_trns.

```
end_solve
```

4. Solve Block 3:

```
begin_solve
  solve_type=ac
  vary_freq from 1 to 200 type=log n_points=400
```

* Above two statements perform AC analysis. The complete frequency range of
* interest is divided in two parts. The first range is evaluated here with
* frequency points chosen in logarithmic fashion with number of points 400.

```
begin_output
  filename=notchm.dat
  variables:
+   mag_of_v3e_ac phase_of_v3e_ac
+   mag_of_v3o_ac phase_of_v3o_ac
end_output
```

* Specify the output file and variables to be written.

```
end_solve
```

5. Solve Block 4:

```
begin_solve
  solve_type=ac
  vary_freq from 200 to 500 type=log n_points=400

* Above two statements perform AC analysis for second range of frequency
* with number of points 400.

  begin_output
    filename=notchm.dat append=yes
    variables:
+     mag_of_v3e_ac phase_of_v3e_ac
+     mag_of_v3o_ac phase_of_v3o_ac
  end_output

* Specify the same output file and variables to be written, which are mentioned
* for first frequency range (Solve Block 3). Indicate "Append" to add the data.

end_solve

end_cf
```

Results:

In this example, simulated transient and frequency domain response are shown in Figs. 3 and 4, respectively.

References:

- [1]K. R. Laker, "Equivalent circuits for the analysis and synthesis of switched-capacitor networks," Bell Syst. Tech. J., vol. 58, pp. 727-767, Mar. 1979.
- [2]Opal and J. Vlach, "Analysis and sensitivity of periodically switched linear networks," IEEE Trans. Circuits and Syst., Vol. 36, No. 4, pp. 522-532, April -1989.

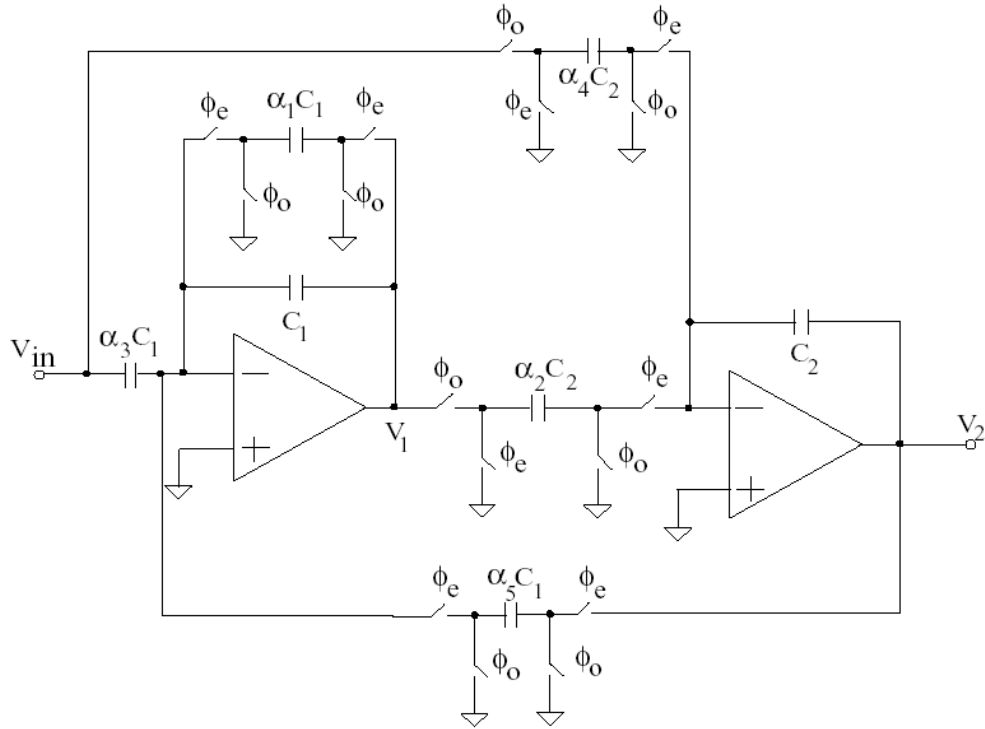
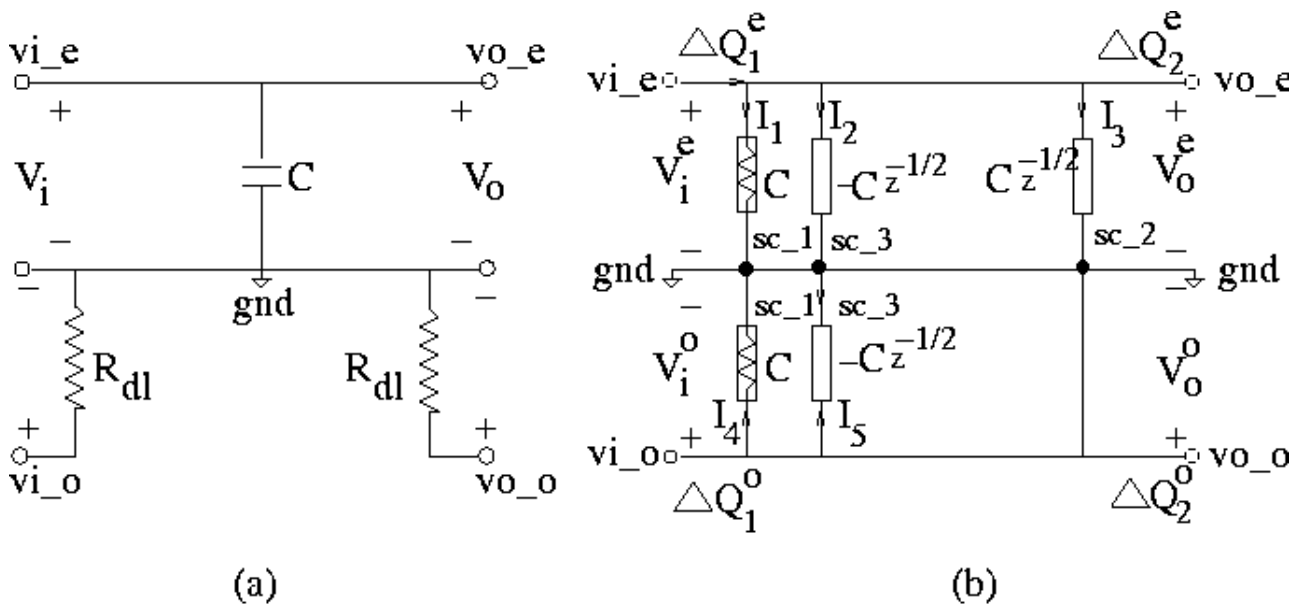


Fig. 1. A second-order notch filter; the parameters are $\alpha_1 = 0.04668$, $\alpha_2 = \alpha_5 = 0.04688$, $\alpha_3 = 1.0233$, $\alpha_4 = 0.04802$, $C_1 = C_2 = 1$ pF, $f_c = 8$ kHz, notch frequency = 60 Hz, 3-dB bandwidth = 58 Hz.



Grounded Capacitor

Fig. 2. Grounded Capacitor (sc_ground_cap) circuit, time-domain (left) and frequency-domain (right).

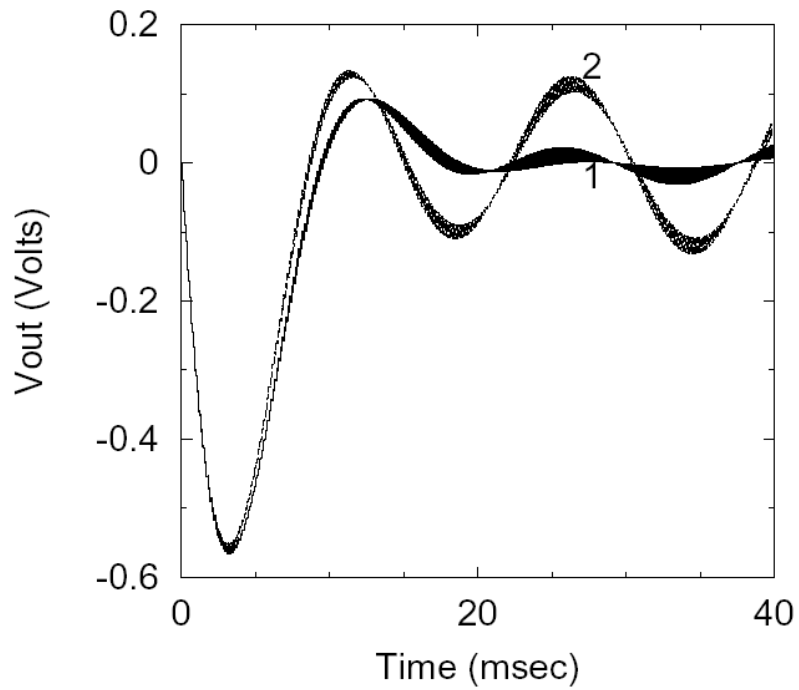


Fig. 3. Transient response showing output signal at node v3e to a sine input with a frequency 60 Hz (1) $R_{off} = 10^{12} \Omega$ and (2) $R_{off} = 10^6 \Omega$, where R_{off} is the off resistance of switches.

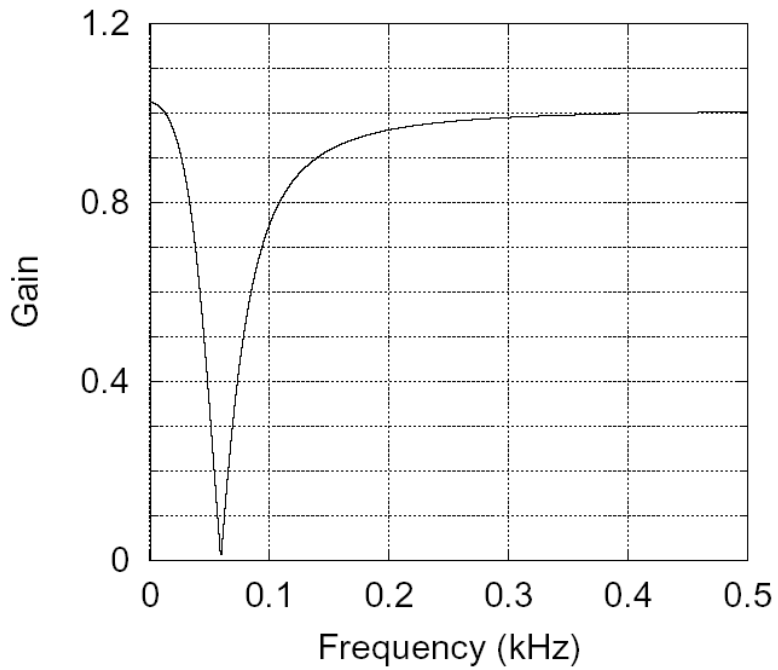


Fig. 4. Frequency-domain response of the notch filter.