Projet : asservissement numérique d'un oscillateur à quartz

É. Carry, J.-M Friedt

jmfriedt@femto-st.fr

transparents à `jmfriedt.free.fr`

19 février 2025

Amplificateur opérationnel suiveur

Problème de gestion des alimentations : comment alimenter l'amplificateur opérationnel suiveur quand ses entrées et sorties doivent aller jusqu'aux tensions d'alimentation ?

- amplificateur opérationnel classique : sortie $V_- + 0,7$ V à $V_+ - 0,7$ V
  Exemple : TL081 (TL084 pour quad), OP27
- amplificateur opérationnel "single supply" : sortie V_- V à $V_+ - 0,7$ V
  Exemple : "LM324 - Single Supply Quad Operational Amplifiers"
- amplificateur opérationnel "rail to rail" : sortie V_- V à V_+ V
  Exemple : ADA4891-4 (quad) $\longrightarrow$

ADA4891-4

Low Cost CMOS, High Speed Rail-to-Rail Amplifier (Quad)

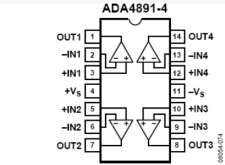


Figure 5. 14-Lead SOIC (R-14) and 14-Lead TSSOP (RU-14)

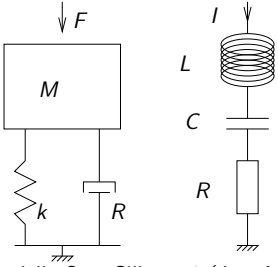
Simulation SPICE : équivalent mécanique-électrique-thermique [1]

Système résonant régi par une équation différentielle du second ordre :

Électrique (unité)	Mécanique (unité)	Thermique (unité)
R (Ω) résistance	R (N.s/m) amortissement	R (K/W) résistance
C (F) capacité	$1/k$ (m/N) raideur^{-1}	$m \times c_p$ (J/K) capacité calorifique
L (H) inductance	M (kg) masse	
V (V) potentiel	F (N) force	T (K) température
q (C) charge	x (m) position	
$i = \dot{q}$ (A=C/s) courant	$v = \dot{x}$ (m/s) vitesse	P (W=J/s) puissance

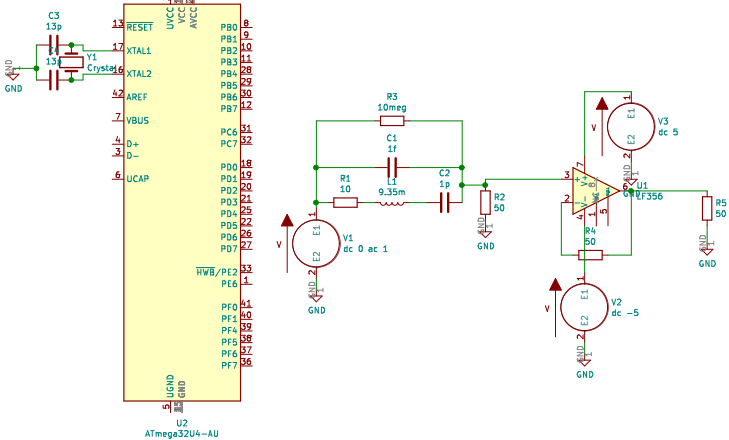
$$M\frac{d^2x}{dt^2} + R\frac{dx}{dt} + kx = F$$

$$L\frac{d^2q}{dt^2} + R\underbrace{\frac{dq}{dt}}_{i} + \frac{1}{C}q = U$$



1. J.-M Friedt, *Introduction à SPICE3 : simulation de circuits électroniques, et au-delà*, OpenSilicum 1 (Jan.-Mars 2011)

Simulation SPICE



- Composant spécifique SPICE (Add → pspice → VSOURCE)
- Définition des propriétés de la source dans sa valeuValid
- Désactiver les composants qui ne font pas partie de la simulation (Edit → Edit Spice Model → Disable symbol for simulation)
- Tools (KiCAD5)/Inspect (KiCAD≥6) → Simulator → Simulation → Settings
- Run/Stop Simulation → Probe pour sélectionner le point de mesure
- Simulation → Show SPICE Netlist pour récupérer le fichier ngspice

Modélisation de l'amplificateur opérationnel

Utilisation de SPICE (*Simulation Program with Integrated Circuit Emphasis* – Univ. Berkeley, CA) et sa version opensource `ngspice`[2]

- filtre passe-bas : modélisation d'un amplificateur opérationnel idéal
- ampli-op idéal : source de tension commandée en tension avec fort gain, suivi de passe bas pour une bande passante finie (éviter oscillations)

```
* +in (=1) -in (=2) out (=3) -- 100k=gain, rbw.cbw=LPF, eout=suiveur
.subckt opamp 1 2 3
e 3 0 1 2 100k
.ends opamp
```

2. J.-M Friedt, *Introduction à SPICE3 : simulation de circuits électroniques, et au-delà*, OpenSilicum 1 (Janvier-Mars 2011) à `jmfriedt.free.fr/lm_spice.pdf` et cours de Rémy Brendel à `http://jmfriedt.free.fr/Spice3.pdf`

Modélisation de l'amplificateur opérationnel

Utilisation de SPICE (*Simulation Program with Integrated Circuit Emphasis* – Univ. Berkeley, CA) et sa version opensource `ngspice`[3]

- filtre passe-bas : modélisation d'un amplificateur opérationnel idéal
- ampli-op idéal : source de tension commandée en tension avec fort gain, suivi de passe bas pour une bande passante finie (éviter oscillations)

```
* +in (=1) -in (=2) out (=3) -- 100k=gain, rbw.cbw=LPF, eout=suiveur
.subckt opamp 1 2 3
rin 1 2 2meg
e 3 0 1 2 100k
.ends opamp
```

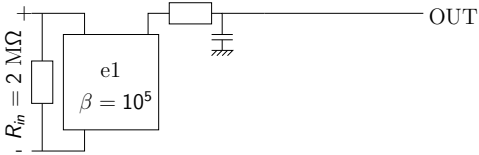
3. J.-M Friedt, *Introduction à SPICE3 : simulation de circuits électroniques, et au-delà*, OpenSilicum 1 (Janvier-Mars 2011) à `jmfriedt.free.fr/lm_spice.pdf` et cours de Rémy Brendel à `http://jmfriedt.free.fr/Spice3.pdf`

Modélisation de l'amplificateur opérationnel

Utilisation de SPICE (*Simulation Program with Integrated Circuit Emphasis* – Univ. Berkeley, CA) et sa version opensource `ngspice`[4]

- filtre passe-bas : modélisation d'un amplificateur opérationnel idéal
- ampli-op idéal : source de tension commandée en tension avec fort gain, suivi de passe bas pour une bande passante finie (éviter oscillations)

```
* +in (=1) -in (=2) out (=3) -- 100k=gain, rbw.cbw=LPF, eout=suiveur
.subckt opamp 1 2 3
rin 1 2 2meg
e 4 0 1 2 100k
rbw 4 3 0.5meg
cbw 3 0 31.85nf
.ends opamp
```

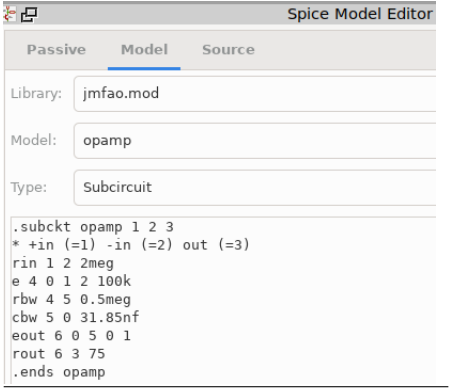


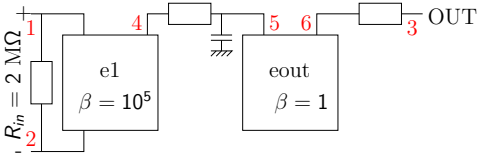
4. J.-M Friedt, *Introduction à SPICE3 : simulation de circuits électroniques, et au-delà*, OpenSilicum 1 (Janvier-Mars 2011) à `jmfriedt.free.fr/lm_spice.pdf` et cours de Rémy Brendel à `http://jmfriedt.free.fr/Spice3.pdf`

Modélisation de l'amplificateur opérationnel

Utilisation de SPICE (*Simulation Program with Integrated Circuit Emphasis* – Univ. Berkeley, CA) et sa version opensource `ngspice`[5]

- filtre passe-bas : modélisation d'un amplificateur opérationnel idéal
- ampli-op idéal : source de tension commandée en tension avec fort gain, suivi de passe bas pour une bande passante finie (éviter oscillations)





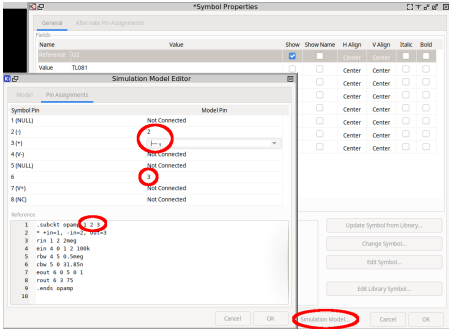
✓`Alternate node sequence: 3 2 6`
pour faire correspondre le modèle SPICE avec notre assignation de broches

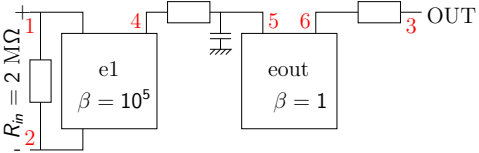
5. J.-M Friedt, *Introduction à SPICE3 : simulation de circuits électroniques, et au-delà*, OpenSilicum 1 (Janvier-Mars 2011) à `jmfriedt.free.fr/lm_spice.pdf` et cours de Rémy Brendel à `http://jmfriedt.free.fr/Spice3.pdf`

Modélisation de l'amplificateur opérationnel

Utilisation de SPICE (*Simulation Program with Integrated Circuit Emphasis* – Univ. Berkeley, CA) et sa version opensource `ngspice`[5]

- filtre passe-bas : modélisation d'un amplificateur opérationnel idéal
- ampli-op idéal : source de tension commandée en tension avec fort gain, suivi de passe bas pour une bande passante finie (éviter oscillations)

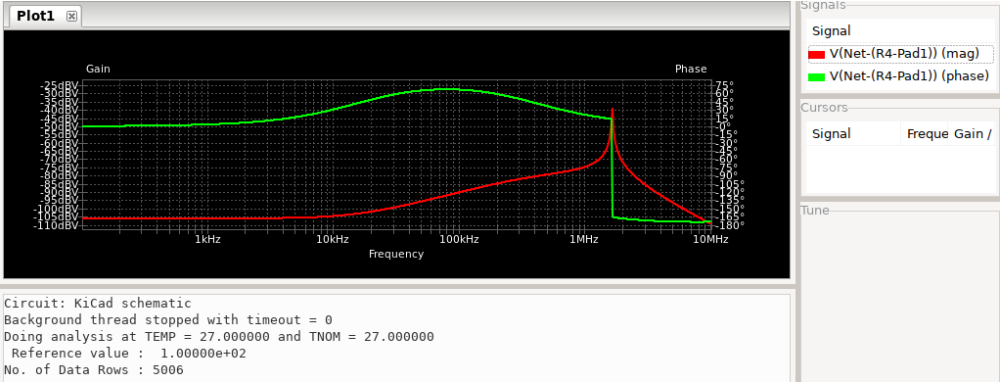




5. J.-M Friedt, *Introduction à SPICE3 : simulation de circuits électroniques, et au-delà*, OpenSilicum 1 (Janvier-Mars 2011) à `jmfriedt.free.fr/lm_spice.pdf` et cours de Rémy Brendel à `http://jmfriedt.free.fr/Spice3.pdf`

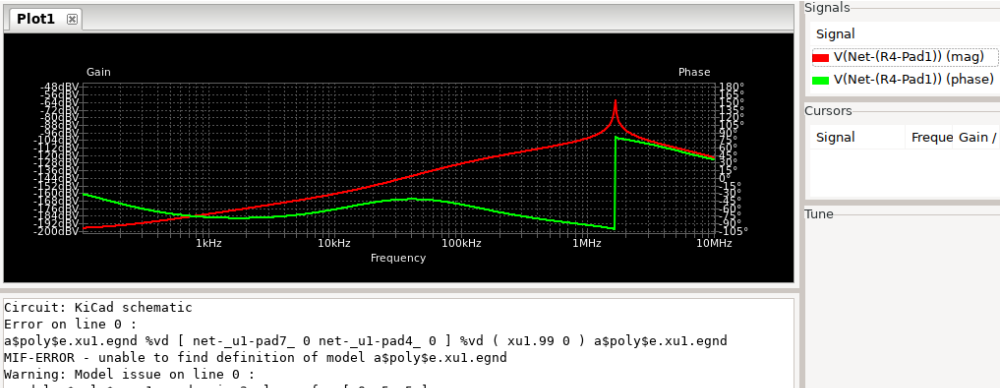
Simulation SPICE (ngspice)

- Charger un modèle : `Edit Spice Model` → `Model` → `Library` et charger fichier du sous-circuit (`.lib` rédigé avec son éditeur favori), sélectionner le `Model` et `Type=Subcircuit`
- Possibilité de charger des modèles "complexes", e.g. pour le TL081 (mais polynôme de LTSpice) `http://www.ti.com/product/TL081/toolssoftware`
- `Alternate Node Sequence` pour faire correspondre broches du modèle et du composant (3 2 7 4 6 avec modèle TL081, 3 2 6 avec modèle idéal sans alim.)
- Tune pour balayer la valeur d'un composant et voir son impact sur la fonction de transfert



Simulation SPICE (ngspice)

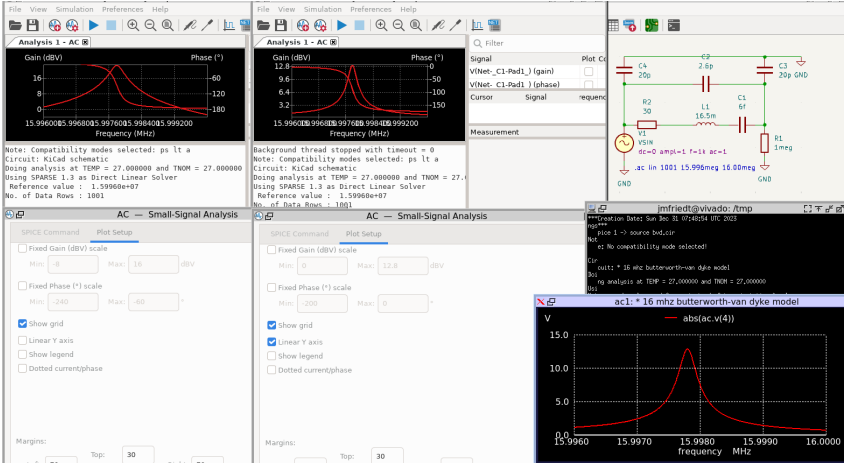
- Charger un modèle : `Edit Spice Model` → `Model` → `Library` et charger fichier du sous-circuit (`.lib` rédigé avec son éditeur favori), sélectionner le `Model` et `Type=Subcircuit`
- Possibilité de charger des modèles "complexes", e.g. pour le TL081 (mais polynôme de LTSpice) `http://www.ti.com/product/TL081/toolssoftware`
- `Alternate Node Sequence` pour faire correspondre broches du modèle et du composant (3 2 7 4 6 avec modèle TL081, 3 2 6 avec modèle idéal sans alim.)
- Tune pour balayer la valeur d'un composant et voir son impact sur la fonction de transfert



ngspice dans KiCAD 7 & 8

- Le facteur de qualité est la **largeur à mi-hauteur de l'admittance du résonateur**
- KiCAD 5 et 6 permettaient de définir la nature de la simulation comme un texte dans le schéma :

  `.ac lin nbre fstart fstop`
- Axes linéaires pour une simulation AC perdus dans KiCAD 7 et donc 8 qui imposent des échelles log-log (diagramme de Bode)
- "facile" à remettre en place en X, moins simple en Y
- `https://gitlab.com/kicad/code/kicad/-/issues/17227`

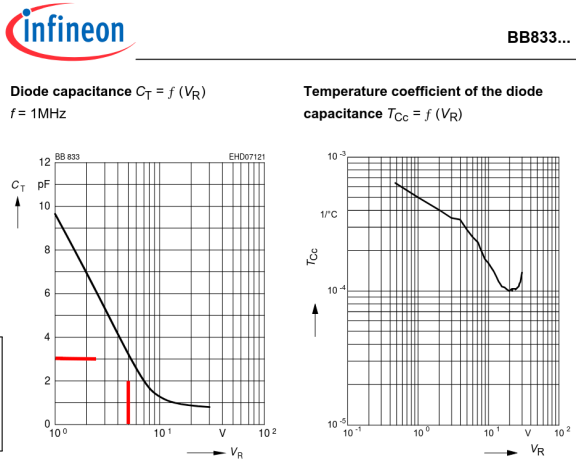


Choix de la varicap

- Gamme de polarisation limitée par la tension d'alimentation du microcontrôleur
- Gamme de tirage limitée par l'excursion de la capacité
- Fonctionnement dans la plage radiofréquence (16 MHz) de l'oscillateur [6]

⇒ Infineon BB833 (fonctionne jusqu'à 2,5 GHz), 9,5 à 3 pF pour 0 à 5 V, en parallèle d'une dizaine de pF pour maintenir l'oscillation.

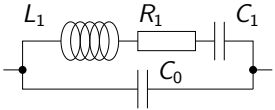
Voir aussi Skyworks SMV1235 (18–4 pF pour 0–5 V), SMV1236 (27–6 pF pour 0–5 V), SMV1247 (1–9 pF pour 0–3 V), SMV1272 (5–16 pF pour 1–4 V), Infineon BBY66 (66–12 pF pour 1–4,5 V)



6. une LED est une varicap ! `http://www.hanssummers.com/varicap/varicapled.html`

Modélisation du comportement de l'oscillateur

- Capacité de tirage d'un résonateur à quartz : on peut aller de la résonance à l'antirésonance (différence donnée par le coefficient de couplage électromécanique du substrat piézoélectrique)
- conditions de Barkhausen : $\varphi = 2n\pi$, $n \in \mathbb{N}$ & gain>pertes,
- résonateur oscille sur sa condition de phase nulle, compte tenu des condensateurs dans la rétroaction.



SPECIFICATIONS

Specifications are typical at 25°C unless otherwise noted.
Specifications are subject to change without notice.

	10 MHz	32MHz	155.52 MHz
Motional Resistance R_1 (Ω)	50	20	50
Motional Capacitance C_1 (fF)	5.5	7.8	0.5
Quality Factor Q (k)	80	36	41
Shunt Capacitance C_0 (pF)	2.2	2.6	3.2
Calibration Tolerance*	A ±0.01% (±100ppm)		
	B ±0.1%		
	C ±1.0%		
Load Capacitance	20 pF (Unless specified by customer)		
Drive Level	500 µW MAX.		
Frequency-Temperature Stability**	-10°C to +70°C from ±10ppm		
	-40°C to +85°C from ±20ppm		
	-55°C to +125°C from ±30ppm		
Aging, first year	5ppm MAX.		
Shock, survival***	3,000g, .3 msec., 1/2 sine		
Vibration, survival	20g rms, 10-2,000 Hz random		
Operating Temperature Range	-10°C to +70°C Commercial -40°C to +85°C Industrial -55°C to +125°C Military		
Storage Temperature	-55°C to +125°C		

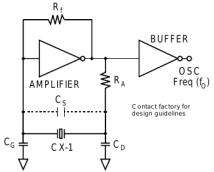
Note: The characteristics of the frequency temperature stability follow that of AT cut thickness shear mode.

CIRCUIT DESIGN

A conventional CMOS Pierce Oscillator is shown below. The crystal oscillates at a frequency f_0 above the crystal's series-resonant frequency f_S. The crystal is effectively inductive and, in combination with R_A, Cd and Cg in the feedback loop, provides approximately 180° phase shift necessary to obtain oscillation. Typical component values for a Pierce Oscillator using a 16 MHz crystal with a 74HC04 hex inverter are shown below:

Typical Application as Pierce Oscillator
Using 74HC04 CMOS Hex Inverter at 5 VDC

CONVENTIONAL CMOS PIERCE OSCILLATOR CIRCUIT

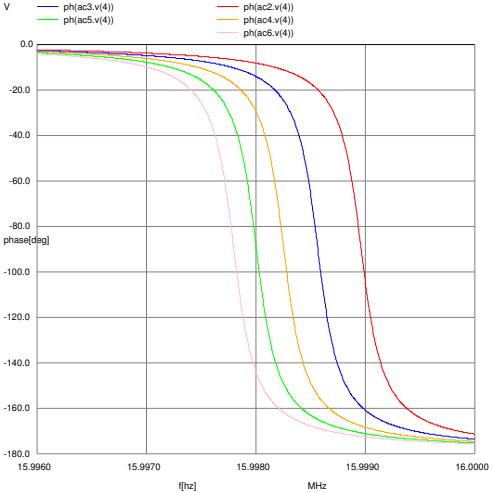


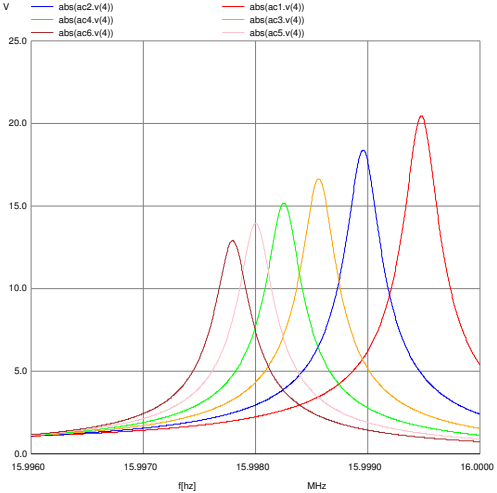
- $f_r = \frac{1}{2\pi}\sqrt{\frac{1}{L_1 \cdot C_1}}$
- $Q = \frac{1}{2\pi \cdot f_r \cdot R_1 \cdot C_1}$
- $\frac{d\varphi}{df} = \frac{360}{\pi} \cdot \frac{Q}{f_r}$
- $C_1 = C_0 \left(\frac{\omega_a^2 - \omega_r^2}{\omega_r^2}\right) = C_0 \cdot k^2$

1. Statek CX-1-03 CRYSTAL 8 MHz to 160 MHz Miniature AT-Cut Quartz Crystal
2. J.R. Vig, *Quartz Crystal Resonators and Oscillators*, tutorial UFFC, New Orleans (2000), `http://www.am1.us/Local_Papers/U11625%20VIG-TUTORIAL.pdf` slide 73

Modélisation du comportement de l'oscillateur

- Capacité de tirage d'un résonateur à quartz : on peut aller de la résonance à l'antirésonance (différence donnée par le coefficient de couplage électromécanique du substrat piézoélectrique)
- $\Rightarrow$ modéliser le signal oscillant, le filtre passe bas, la varicap et le tirage en tension





Exploitation de ngspice depuis octave

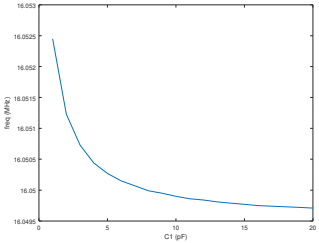
- ngspice pour la simulation de circuits électroniques, mais peu de fonctions de traitement
- octave permet de traiter les fichiers textes (print) générés par ngspice
- ⇒ générer la *netlist* (fichier texte) depuis octave, simuler, puis exploiter les résultats (e.g. optimisation par fminsearch())

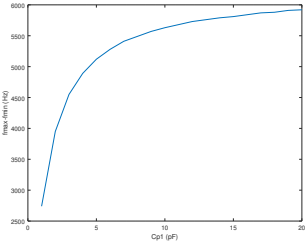
```
n=1;
for d=1:1:20
 m=1;
 for c=1:1:20
  f=fopen('rcoctave.cir','w');   % netlist
  fprintf(f,"* tirage resonateur\n");
  fprintf(f,"V1 99 0 1.0 ac 1.0\n"); % source
  fprintf(f,"rs 99 1 100k\n");       % src Z
  fprintf(f,"R1 1 2 22.8\n");        % BVD motional
  fprintf(f,"L1 2 3 70.3m\n");
  fprintf(f,"C1 3 4 1.4f\n");
  fprintf(f,"C0 1 4 0.65p\n");       % BVD elec
  fprintf(f,"Cp1 1 0 %fp\n",d);      % capa
  fprintf(f,"Cp2 4 0 %fp\n",c);      % capa
  fprintf(f,"RL 4 0 100k\n");        % load impedance
  fprintf(f,".control\n");
  fprintf(f,"ac lin 5000 16.02meg 16.08meg\n");
  fprintf(f,"print vdb(4) vp(4) > t\n");
  fprintf(f,"quit\n");   % 1=line, 2=freq, 3=dB, 4=ph
  fprintf(f,".endc\n");
  fclose(f);
  system('ngspice rcoctave.cir > /dev/null');
  system('cat t | grep ^[0-9] t > tt');
  system('rm t');
  load tt
  [a,b]=max(tt(:,3));    % recherche max d'admittance
  solution(m,n)=tt(b,2); % 2eme colonne = frequence
  m=m+1;
 end
 n=n+1
end
```

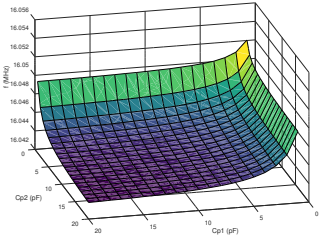
Exploitation de ngspice depuis octave

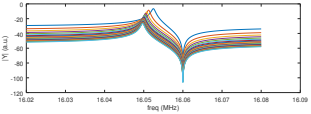
- ngspice pour la simulation de circuits électroniques, mais peu de fonctions de traitement

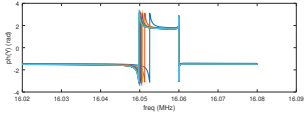
- octave permet de traiter les fichiers textes (print) générés par ngspice
- ⇒ générer la *netlist* (fichier texte) depuis octave, simuler, puis exploiter les résultats (e.g. optimisation par fminsearch())





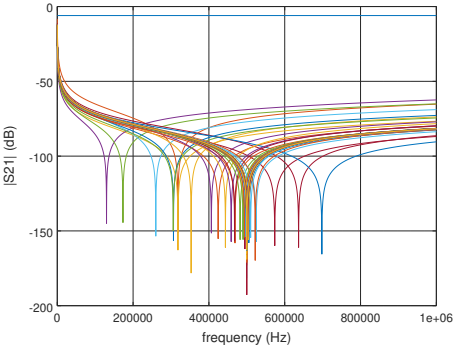






Optimisation de filtre

- fonction `fminsearch()` de Matlab/Octave pour la descente de gradient,
- fonction de coût exploite `ngspice` pour calculer la fonction de transfert d'un filtre LC coupe-bande dont on vise $f_c = 500$ kHz



- Solution : $L = 0.175\ \mu$H, $C = 0.579\ \mu$H
  $\Rightarrow f_c = 5.03e + 05$ Hz

```
L=.1e-6; % conditions initiales (L, C grossiers)
C=1e-6;
X0=[L C];
Xopt=fminsearch('cout_filtre', X0);
% optim_filtre([1e-6,1e-6])

printf("solution=%f %f\n",Xopt(1),Xopt(2));
1/2/pi/sqrt(L*C)
1/2/pi/sqrt(Xopt(1)*Xopt(2))
```

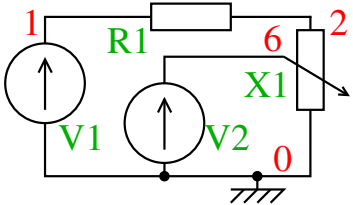
```
function erreur=cout_filtre(param)
L=param(1)
C=param(2)
if ((L<=0) || (C<=0))
  erreur=1e9;
else
  f=fopen('optim_fil.cir','w');
  fprintf(f,"* filtre passe-bande\n");
  fprintf(f,"V1 99 0 1.0 ac 1.0\n");
  fprintf(f,"rs 99 1 10k\n");
  fprintf(f,"L1 1 2 %fu\n",L);
  fprintf(f,"C1 2 0 %fu\n",C);
  fprintf(f,"RL 1 0 10k\n");
  fprintf(f,".control\n");
  fprintf(f,"ac lin 5000 100 1meg\n");
  fprintf(f,"print vdb(1) vp(1) > t\n");
  fprintf(f,"quit\n");
  fprintf(f,".endc\n");
  fclose(f);
  system('ngspice optim_fil.cir > /dev/null');
  system('cat t | grep ^[0-9] t > tt');
  system('rm t');
  load tt;
  plot(tt(:,2),tt(:,3));hold on
  [a,b]=min(tt(:,3));
  freq=tt(b,2)
  erreur=abs(freq-5e5);
 end
end
```

Conclusion (et questions)

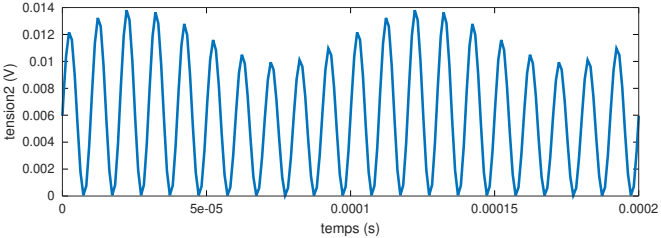
1. **Reproduire les simulations de la varicap,**
2. **Reproduire les simulation de tirage de fréquence du résonateur, compte tenu des informations fournies dans la documentation d'un résonateur, par exemple** `http://www.andhraelec.com/app_quartz_crystals.html`**.**
3. **Étendre la plage de fréquences de simulation : commenter sur l'évolution de la phase. Comment se compare-t-elle à la phase d'un circuit** RLC **série ?**
4. **Quel est le comportement asymptotique (en basse et haute fréquence) d'un circuit** RLC **? d'un circuit Butterworth-van Dyke ? Est-ce cohérent avec les simulations ?**

Passif commandé en tension

```
* resistance

.subckt varres 1 2 CTRL
r1 1 2 1e6
b1 1 2 I=V(1,2)/(V(CTRL)+1)
.ends

V1 1 0 ac 1 dc 1 sin(1 1 100k)
V2 6 0 sin(5 1 10k)
R1 1 2 1000
X1 2 0 6 varres

.control
tran 1u 200u
plot v(2)
.endc
```



- une résistance est déterminée par un courant variant linéairement avec la tension ($I = U/R$)
- facteur de proportionnalité déterminé par une tension (source `B n+ n-` : *Nonlinear dependent source*)

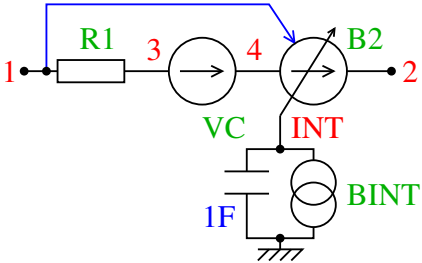


Modélisation du comportement de l'oscillateur

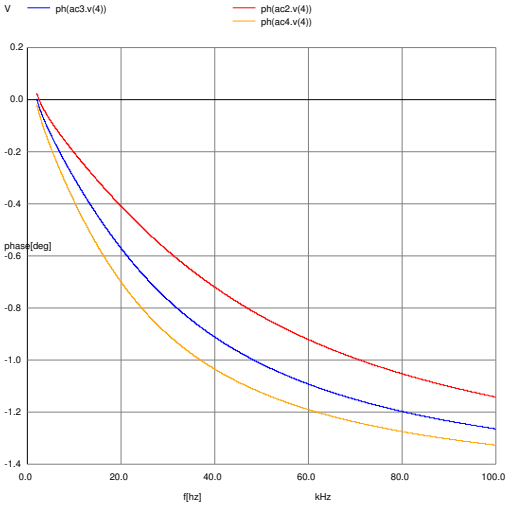
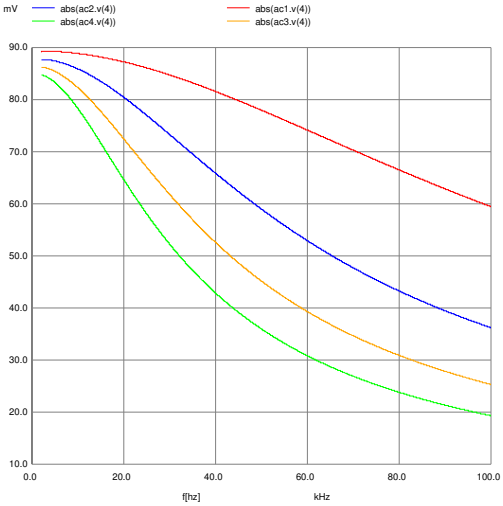
- Modéliser le tirage en tension en faisant varier la **varicap** (*varactor*)
- varactor[7] : $V_C(t) = \frac{1}{C}\int i(t)\cdot dt \Leftrightarrow C = \frac{1}{V_C}\int i(t)\cdot dt$ et $\int i$ est la charge d'un condensanteur de 1 F

```
* varactor demonstration
* (RC circuit with C: varactor)

* V=1/C \int i(t)dt
* => C=1/V \int i(t)dt
.SUBCKT VARICAP 1 2
R1 1 3 1
VC 3 4
BC 4 2 V=(1/(v(1)*10e-9))*v(int)
BINT 0 INT I=I(VC)
CINT INT 0 1
.ENDS

Ve 1 0 AC 1 DC 2 sin(1.5 1 1k)
R1 1 2 1k
C1 2 0 20n
X1 2 0 VARICAP
Coff 2 4 1u
Ro 4 0 100

.control
destroy all
set units = degrees
```

```
alter @ve[dc]=0
ac lin 10001 2k 100k
alter @ve[dc]=1
ac lin 10001 2k 100k
alter @ve[dc]=2
ac lin 10001 2k 100k
alter @ve[dc]=3
ac lin 10001 2k 100k

set hcopydevtype=postscript
set hcopypscolor=1
set color0=rgb:0/0/0
set color1=rgb:f/f/f

plot db(ac1.v(4)) db(ac2.v(4)) db(ac3.v(4)) db(ac4.v(4)) xlabel f[Hz] ylabel |mag(V)|[dB]
plot ph(ac1.v(4)) ph(ac2.v(4)) ph(ac3.v(4)) ph(ac4.v(4)) xlabel f[Hz] ylabel phase[deg]
hardcopy varicap_abs.ps abs(ac1.v(4)) abs(ac2.v(4)) abs(ac3.v(4)) abs(ac4.v(4)) xlabel f[Hz]
hardcopy varicap_pha.ps ph(ac2.v(4)) ph(ac3.v(4)) ph(ac4.v(4)) xlabel f[Hz] ylabel phase[deg]
unset units
.endc
```



7. C. Basso, *SPICE Analog Behavioral Modeling of Variable Passives*, Power Electronics Technology (April 2005)

Modélisation du comportement de l'oscillateur

- Validation du comportement de la varicap en l'insérant dans un filtre RC ($\tau = 1/(RC)$)
- variation de la valeur de la varicap en modifiant la polarisation DC