

TITRE :

Modélisation de composants

Exemple d'un VCO

OBJECTIF :

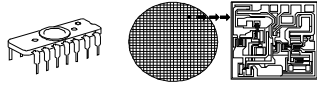
Expliquer le mode opératoire pour créer un modèle de composants à partir d'un schéma équivalent à base d'éléments *discrets* (*résistance, condensateur, bobine, diode, transistor...*) ainsi que de blocs fonctionnels (*Evalue, Gvalue, Etable, Sum, Diff, Integ ...*) puis définir un symbole équivalent associé à un macro modèle utilisable dans les simulations.

THEME :

Réalisation d'un *VCO sinusoidal* idéal à l'aide de blocs fonctionnels définis à partir des équations physiques du composant.

Création de la « Netlist », conversion en un « macro modèle » avec sauvegarde dans une librairie personnelle puis attribution d'un symbole utilisable pour toutes simulations futures.

Exemple d'utilisation : modulation de fréquence.



Modélisation d'un VCO « idéal »

1- Mise en équation

$s(t) = A \cdot \sin q(t)$ tension de sortie du VCO avec $q(t)$ la phase instantanée.

$w(t) = \frac{dq}{dt}$ pulsation instantanée

$f(t) = \frac{w(t)}{2\pi}$ fréquence instantanée. Celle ci varie autour d'une composante fixe f_0 .

$$q(t) = w_0 t + j(t)$$

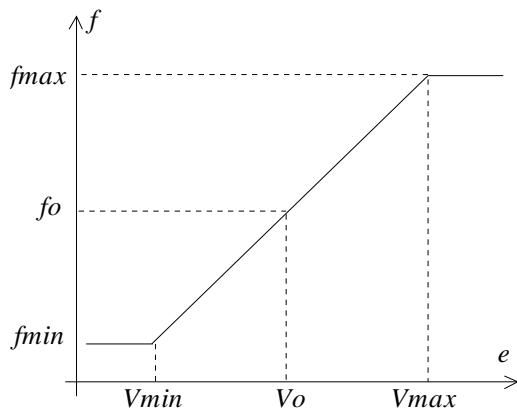
On peut écrire : $w(t) = w_0 + \frac{dj}{dt}$

$$f(t) = f_0 + \frac{1}{2\pi} \frac{dj}{dt}$$

Avec $\frac{1}{2\pi} \frac{dj}{dt}$ variation de fréquence autour de f_0 .

Dans un VCO, cette variation est commandée par la tension d'entrée $e(t)$.

On obtient la fonction de transfert suivante :



On en déduit les équations :

$$f(t) = f_0 + K_0(e(t) - V_0)$$

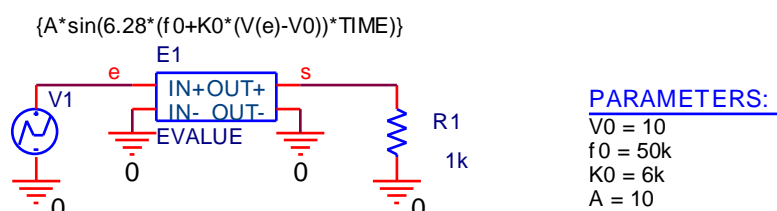
$$K_0 = \frac{f_{\max} - f_{\min}}{V_{\max} - V_{\min}} \text{ la « sensibilité de modulation »}$$

2- Réalisation du schéma fonctionnel

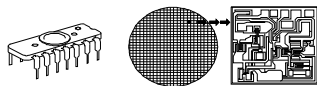
→ Lancer le programme <Capture CIS> dans le menu d' Orcad 9 puis travailler sous <Schématic> après avoir défini un nom de projet

2.1- 1^{ère} solution :

→ réaliser le schéma suivant :

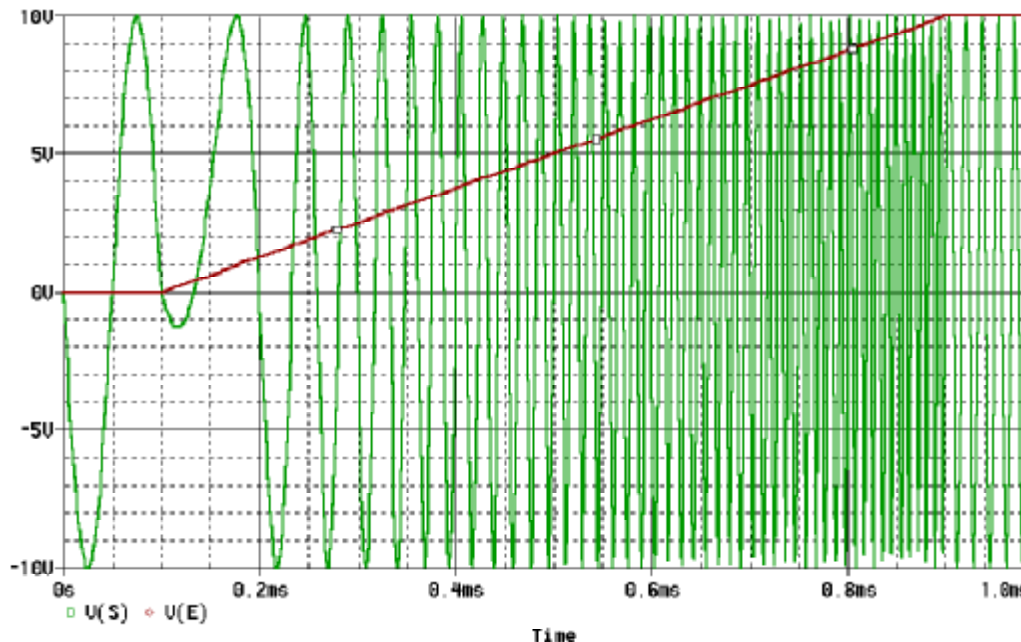


Les blocs fonctionnels utilisés se trouvent dans la librairie ABM.



→ Paramètres de la source V1 ($t1=0$ $V1=0$ $t2=0.1m$ $V2=0$ $t3=0.9ms$ $V3=10$ $t4=1ms$ $V4=10$)

→ Retrouver la réponse temporelle suivante : ($t=1ms$ et $max\ step\ size=1\mu s$ « pas de calcul »)



Conclusions: Il y a deux inconvénients avec ce schéma équivalent:

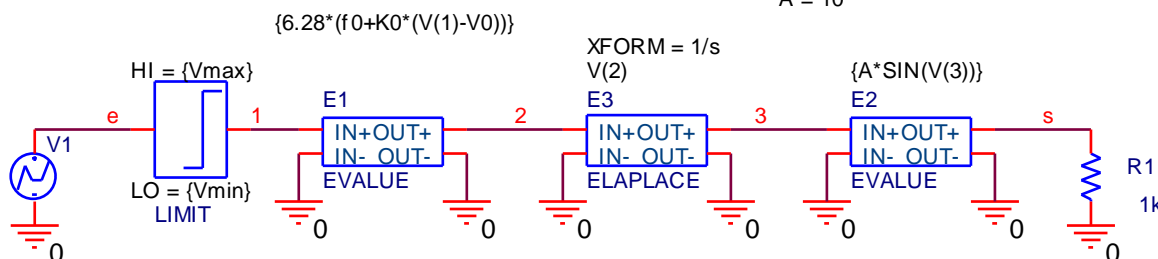
- Aucune limite en fréquence
- Aux variations brusques de l'entrée on distingue des discontinuités dans l'évolution de la fréquence.

2.2- 2^{ème} solution:

→ réaliser le schéma suivant

PARAMETERS:

$f_0 = 50k$
 $V_0 = 10$
 $K_0 = 6k$
 $V_{max} = 15$
 $V_{min} = 5$
 $A = 10$



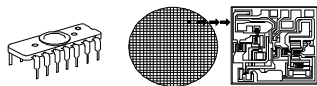
LIMIT : permet de limiter l'évolution de la fréquence entre 2 valeurs $f_0+K_0.(V_{min}-V_0)$ et $f_0+K_0.(V_{max}-V_0)$

E1 : fournit une image de la pulsation proportionnelle à l'entrée $e(t)$: $w(t)=2p(f_0+K_0.(e(t)-V_0))$

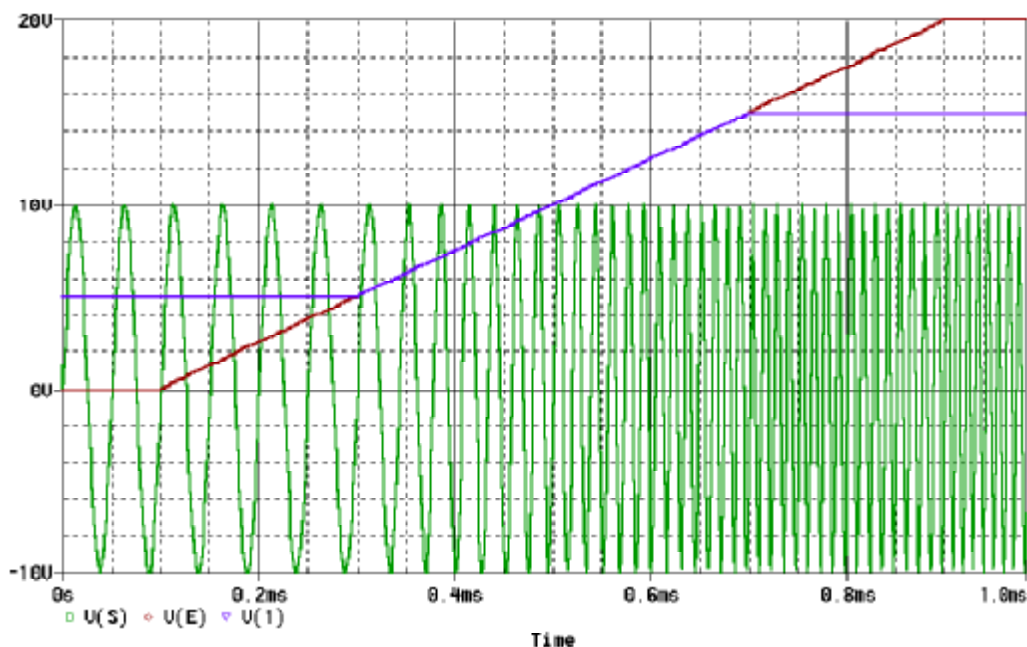
E3 : fournit la phase instantanée $q(t)$ par intégration de la pulsation $w(t)$.

E2 : fournit la tension de sortie sinusoidal d'amplitude A et de valeur angulaire $q(t)$.

→ Paramètres de la source V1 ($t1=0$ $V1=0$ $t2=0.1m$ $V2=0$ $t3=0.9ms$ $V3=20$ $t4=1ms$ $V4=20$)



→ Retrouver la réponse temporelle suivante : ($t=1ms$ et $max\ step\ size=1\mu s$ « pas de calcul »)

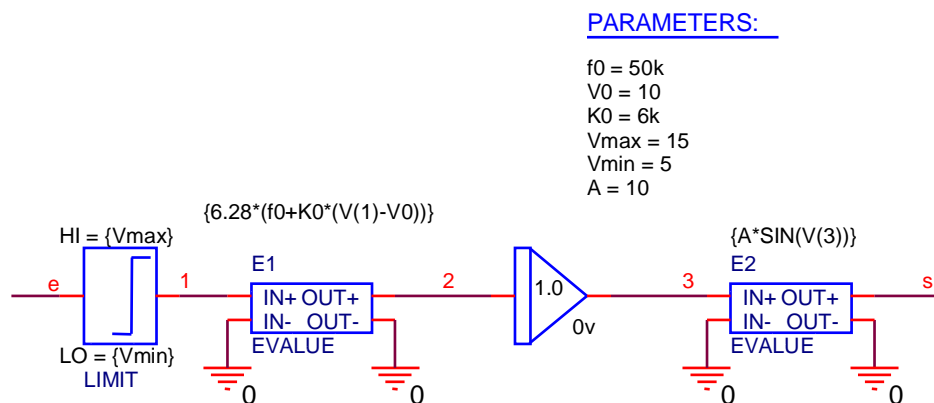


Conclusions:

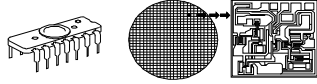
- La fréquence est limitée entre deux valeurs extrêmes paramétrables par V_{min} et V_{max} .
- Il n'y a plus de discontinuité de la fréquence aux variations brusques de la tension d'entrée.
- Le bloc de Laplace engendre un problème de convergence lors de la simulation et il faut cocher dans la fenêtre de paramétrage l'option "skip the initial transient bias point circulation" pour le résoudre. L'inconvénient est que toutes les valeurs transitoires définies dans le paramètre IC des composants sont inhibées.

2.3- 3^{ème} solution: on remplace le bloc de Laplace par un bloc INTEG

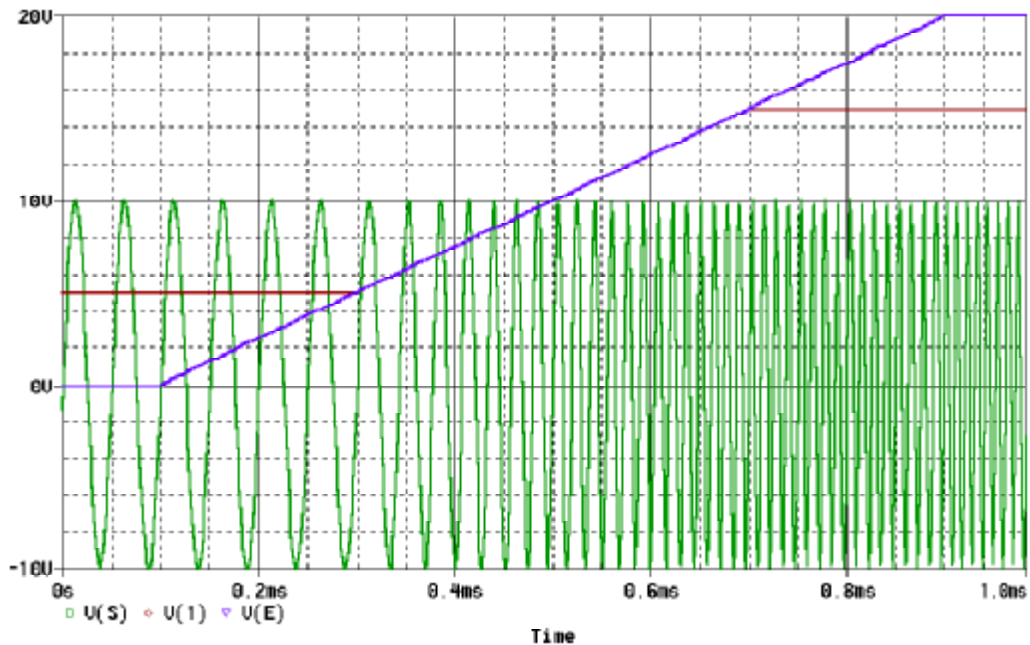
→ réaliser le schéma suivant:



→ Les réglages sont identiques, désélectionner l'option "skip the initial transient bias point circulation".



→ Retrouver la réponse temporelle suivante : ($t=1ms$ et $max\ step\ size=1\mu s$ « pas de calcul »)



Conclusion: plus de problème de convergence et de discontinuité de la fréquence.

3- Réalisation du macro modèle

3.1- Edition de la « Netlist »

→ Editer la « Netlist » du dernier schéma sans le générateur d'entrée $V1$ et la résistance de sortie $R1$

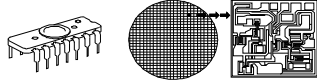
remarque : il est très important de repérer les entrées et sorties par des « Net Alias » (ici e et s)

Sous schématic <P Spice>® <Create Netlist> puis <View Netlist>

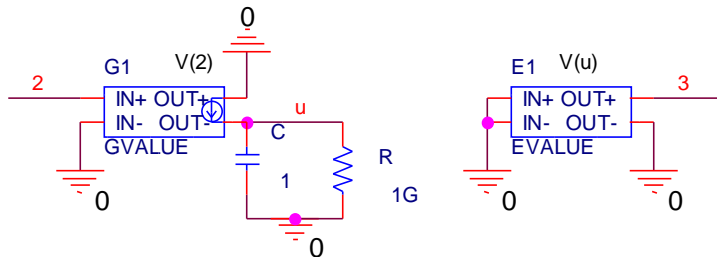
On trouve le listing suivant :

```
* source VCO
E_LIMIT1 1 0 VALUE {LIMIT(V(e),{Vmin},{Vmax})}
E_E1 2 0 VALUE { {6.28*(f0+K0*(V(1)-V0))} }
G_INTEG1 0 $$U_INTEG1 VALUE {V(2)}
C_INTEG1 $$U_INTEG1 0 {1/1.0}
R_INTEG1 $$U_INTEG1 0 1G
E_INTEG1 3 0 VALUE {V($$U_INTEG1)}
.IC V($$U_INTEG1)=0v
E_E2 S 0 VALUE { {A*sin(V(3))} }
.PARAM K0=6k Vmin=5 Vmax=15 V0=10 f0=50k A=10
```

Remarque : Le bloc *INTEG* est un modèle constitué de blocs élémentaires *GVALUE* et *EVALUE* ainsi que de deux éléments discrets $R=1GW$ et $C=1F$: charge d'un condensateur à courant proportionnel à la tension d'entrée $V(2)$. $i = C \frac{du}{dt}$ et i prop à $V(1)$. En prenant $C=1F$ on a $V(2) = \int V(1).dt$



Modèle du bloc INTEG (pour information)



```
* source INTEG
G_G1      0 U VALUE { V(2) }
C_C       0 U 1
R_R       0 U 1G
E_E1      3 0 VALUE { V(u) }
```

3.2- Création du fichier.lib (intègre la syntaxe des macro modèles)

→ Ouvrir une nouvelle page (ou un fichier déjà existant) sous un éditeur de texte et coller la « Netlist » précédente puis ajouter les lignes comme dans l'exemple suivant : (respecter la syntaxe)

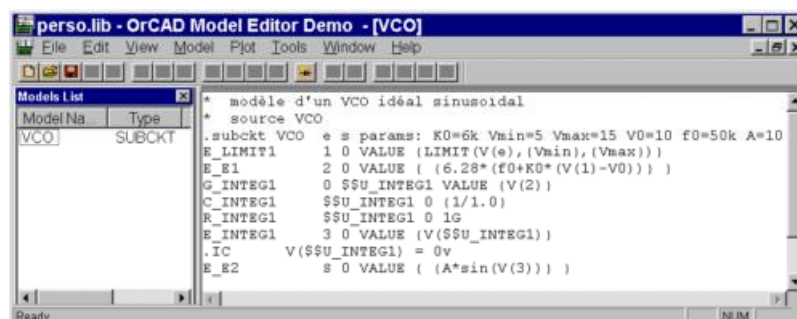
```
* modèle d'un VCO idéal sinusoïdal
* source VCO
.subckt VCO e s params: K0=6k Vmin=5 Vmax=15 V0=10 f0=50k A=10
E_LIMIT1  1 0 VALUE { LIMIT(V(e),{Vmin},{Vmax}) }
E_E1      2 0 VALUE { {6.28*(f0+K0*(V(1)-V0))} }
G_INTEG1  0 $$U_INTEG1 VALUE { V(2) }
C_INTEG1  $$U_INTEG1 0 {1/1.0}
R_INTEG1  $$U_INTEG1 0 1G
E_INTEG1  3 0 VALUE { V($$U_INTEG1) }
.IC       V($$U_INTEG1) = 0v
E_E2      S 0 VALUE { {A*sin(V(3))} }
.ends
```

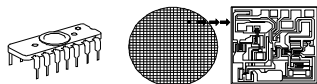
remarques :
- Après l'instruction **.subckt** on écrit le nom, les entrées, les sorties et éventuellement la lignes des paramètres si on veut que ceux ci soient modifiables dans le symbol final.
- Il est important de valider la dernière ligne après **.ends**
- Les commentaires sont signalés par une étoile. (inopérants pour la simulation)
→ enregistrer le fichier texte avec l'extension **.lib** (exemple : perso.lib)

3.4- Edition du macro modèle et attribution d'un symbole

→ Sous « Capture Cis », ouvrir le fichier de modèle (perso.lib)
<File>→<Open> →<Pspice Library> sélectionner le fichier perso.lib

La fenêtre suivante apparaît :

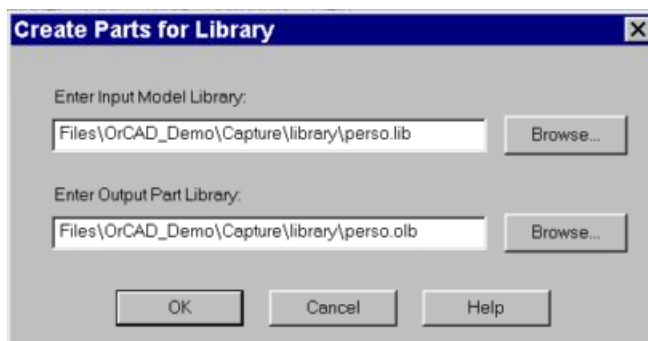




remarque : il est possible dans cet éditeur de modifier le listing.

→ Sous « *Orcad Model Editor* », créer le symbole associé : <File> → <Create Parts>

La fenêtre suivante apparaît :



remarques : Dans la première ligne on sélectionne le fichier *perso.lib* que l'on vient de créer, puis apparaît automatiquement dans la deuxième ligne un nom de fichier *perso.olb* dans le même répertoire. (*On peut éventuellement modifier cette deuxième ligne en choisissant un répertoire ou un nom.olb différents*)

Après validation une fenêtre de dialogue apparaît dans laquelle toutes les opérations effectuées sont indiquées ainsi que les éventuelles erreurs.

→ Valider cette dernière fenêtre et le symbole est créé.

4- Utilisation du symbole et transformation de son apparence

4.1- Utilisation du symbole

→ Ouvrir un nouveau projet (*exemple : VCO.dsn*).

→ importer le symbole *VCO*

Sous « *Schématic* » : <Place> → <Part> → <Add Library>

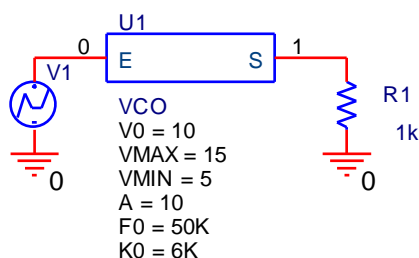
Ajouter la nouvelle library créée *perso.olb*

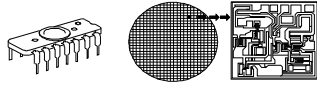
Sélectionner le composant *VCO*

On obtient le symbole suivant :



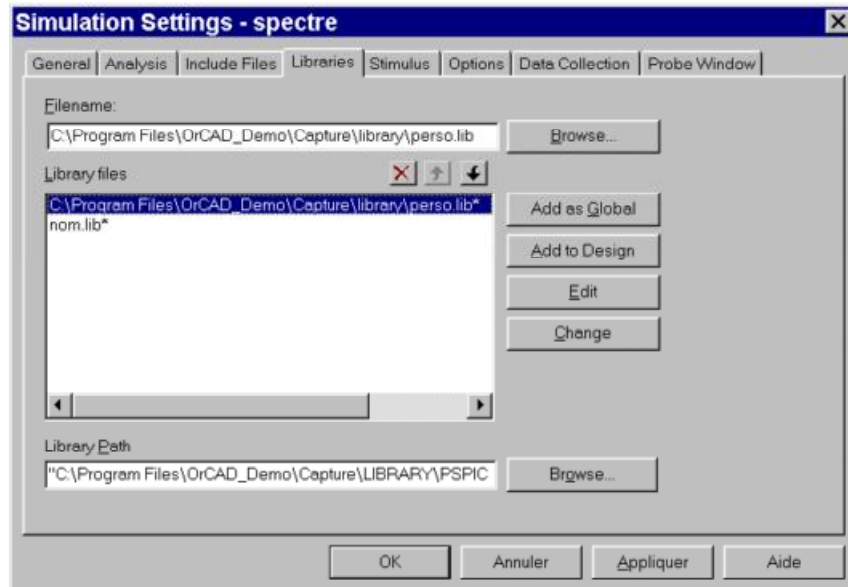
→ Réaliser le schéma suivant:





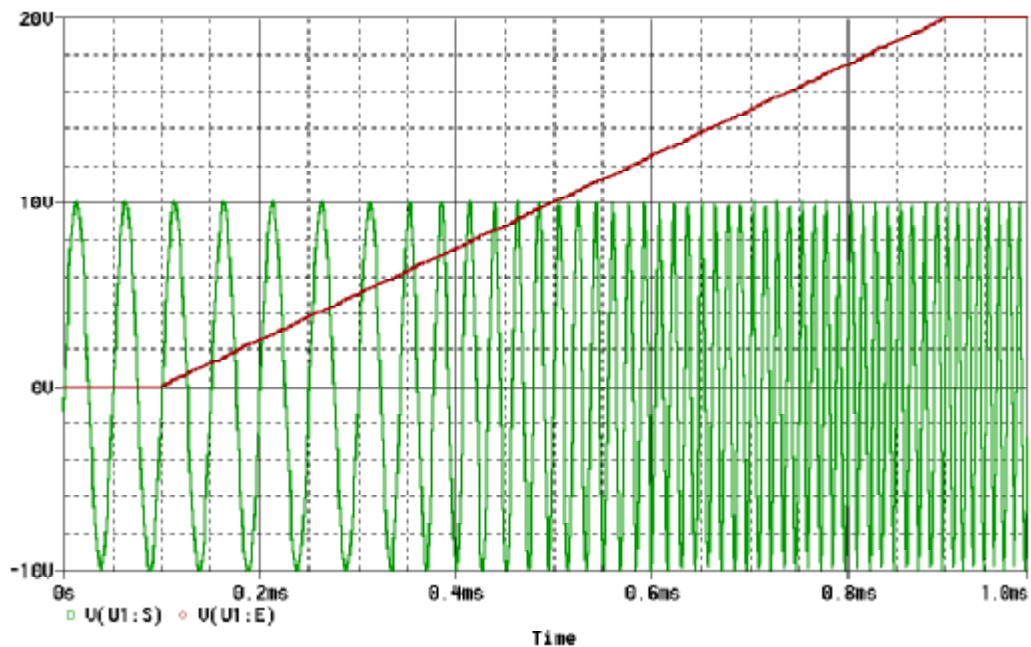
→ Paramètres de la souce VI ($t1=0$ $V1=0$ $t2=0.1m$ $V2=0$ $t3=0.9ms$ $V3=20$ $t4=1ms$ $V4=20$)

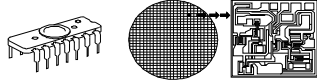
→ Ajouter la librairie (*perso.lib*) contenant le macro modèle dans la fenêtre de paramétrage:



remarque : en déclarant cette librairie comme « *Global* », elle sera automatiquement insérer dans tous les nouveaux projets.

→ Retrouver la réponse temporelle suivante : ($t=1ms$ et $max\ step\ size=1\mu s$ « pas de calcul »)





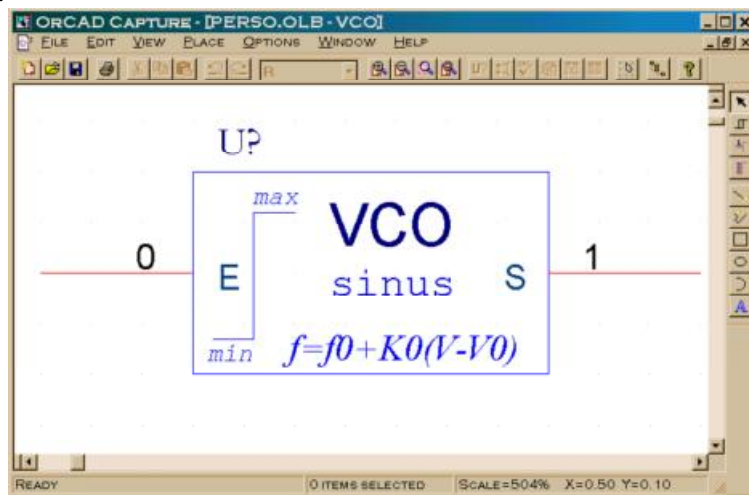
4.2- Modification du symbole

→ Editer le symbole :

Sous « Capture » : <File> → <Open> → <Library> sélectionner la library *perso.olb* puis éditer le symbole *VCO* (double clic)

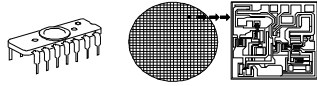
→ Dans la fenêtre d'édition effectuer les modification souhaitées.

Exemple :



→ Enregistrer le nouveau symbole : <File> → <Save>.

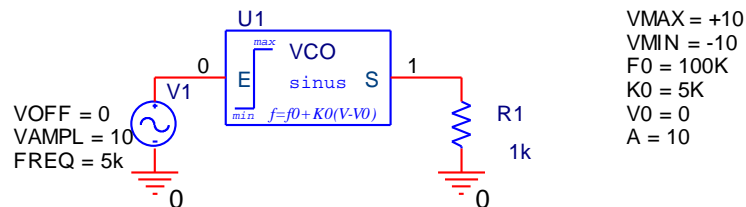
Remarque : dans les anciens projets contenant l'ancien symbole, il ne change pas de forme.



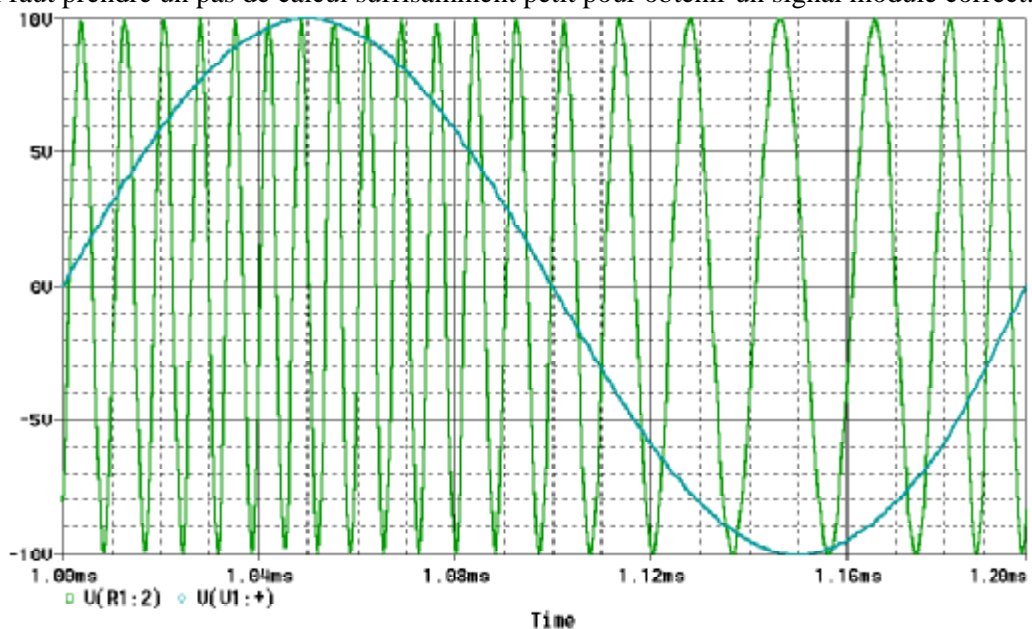
5- Exemple d'utilisation : modulation de fréquence

5.1- Spectre d'un signal modulé en fréquence

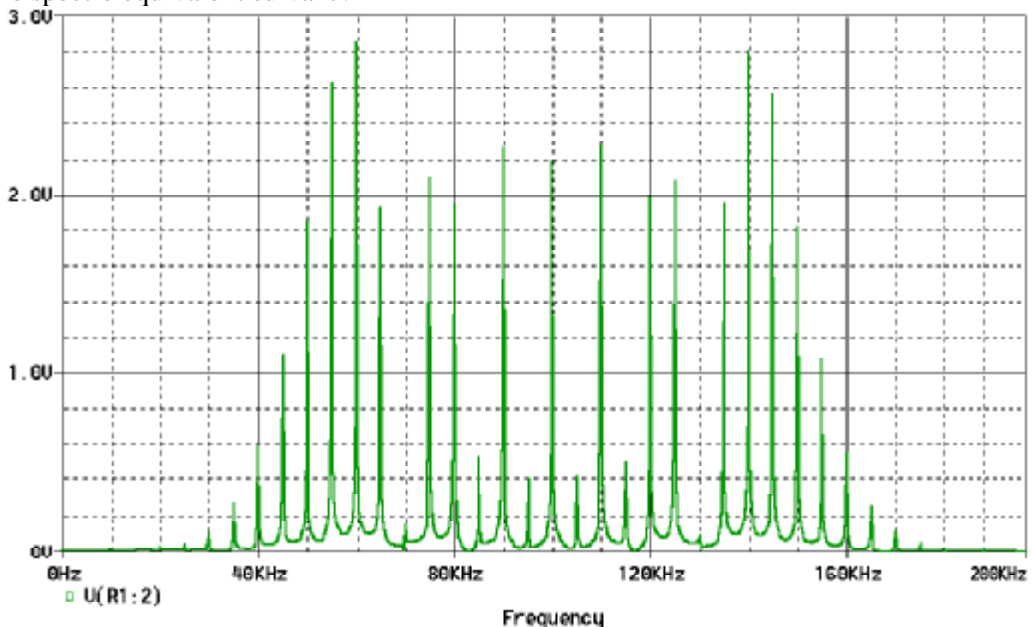
→ Dans un nouveau projet, réaliser le schéma suivant :



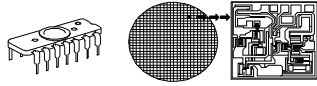
→ Retrouver la réponse temporelle suivante : ($t=5ms$ et $max\ step\ size=0.5\mu s$ « pas de calcul »)
remarque: il faut prendre un pas de calcul suffisamment petit pour obtenir un signal modulé correct.



→ Afficher le spectre équivalent suivant :



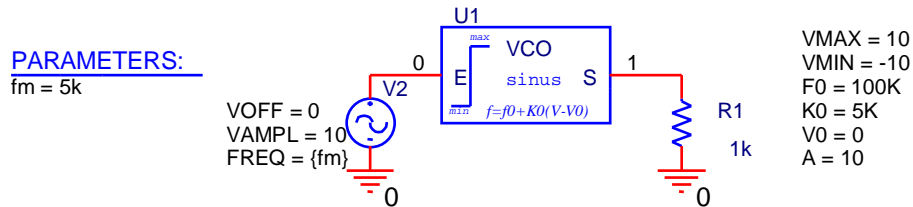
→ Exploiter le spectre: vérification des amplitudes (coef de Bessel), vérification de la règle de Carson.....



5.2- Invariance de la largeur de bande utile

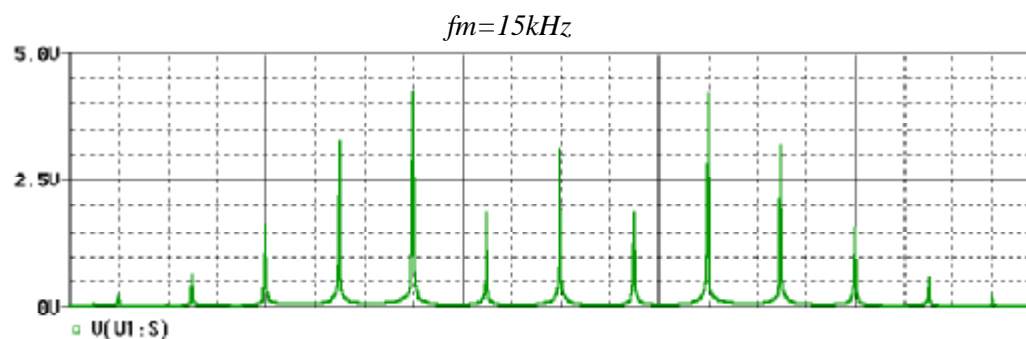
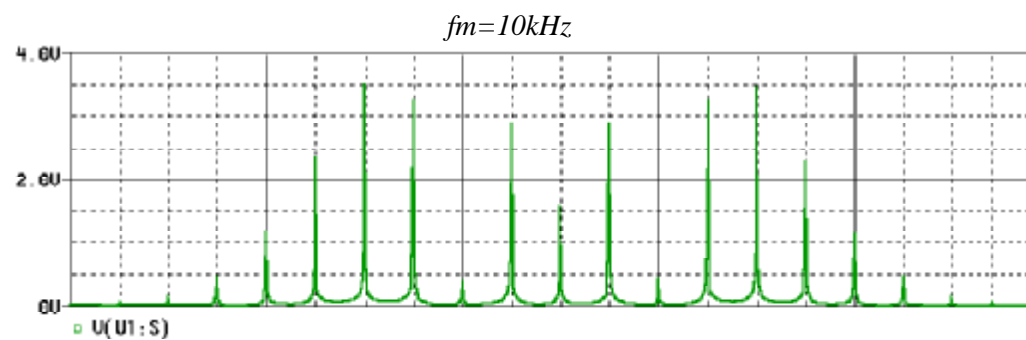
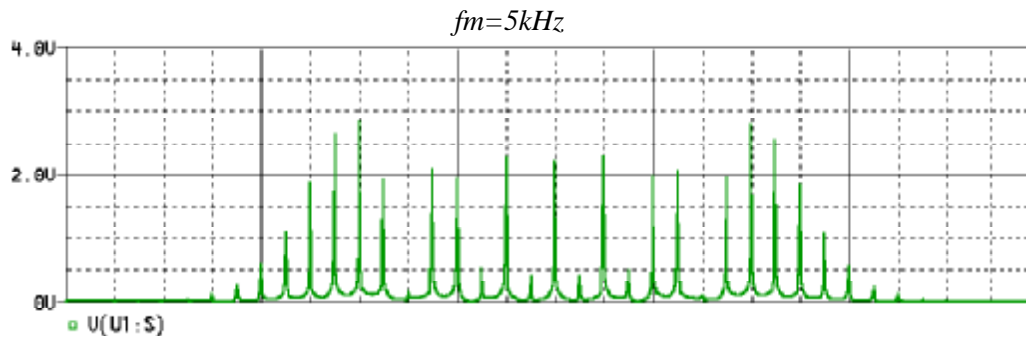
On veut vérifier que dans le cas de la modulation de fréquence, la largeur de bande utile ne change pas lorsque la fréquence du signal modulant varie.

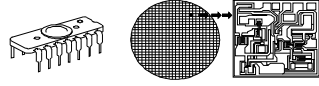
→ Déclarer la fréquence du signal d'entrée comme paramètre f_m .



→ Définir f_m comme paramètre global dans la fenêtre de paramétrage et choisir 3 valeurs.

→ Afficher les différents spectres:





5.3- Influence de l'indice de modulation à f_m constante

Equations :

$$e_m(t) = A_m \cdot \sin(\omega_m t) \quad \text{signal modulant}$$

$$f = f_0 + K_0 \cdot e_m(t) \quad \text{avec } V_0 = 0V \quad (\text{modulation de fréquence})$$

$$f = f_0 + K_0 \cdot A_m \cdot \sin(\omega_m t) \quad \text{soit } df = K_0 \cdot A_m \quad \text{excursion maximale de fréquence}$$

$$\omega(t) = 2\pi f(t) = \omega_0 + 2\pi K_0 A_m \sin(\omega_m t) = \frac{dq}{dt}$$

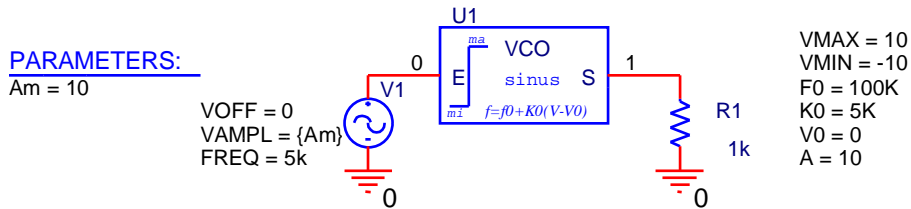
$$q(t) = \omega_0 t - \frac{2\pi K_0 A_m}{\omega_m} \cos(\omega_m t) + j_0 \quad \text{on choisit } j_0 = 0$$

$$s(t) = A \cdot \sin q(t) = A \cdot \sin(\omega_0 t - m \cdot \cos \omega_m t)$$

$$m = \frac{2\pi K_0 A_m}{2\pi f_m} = \frac{df}{f_m}$$

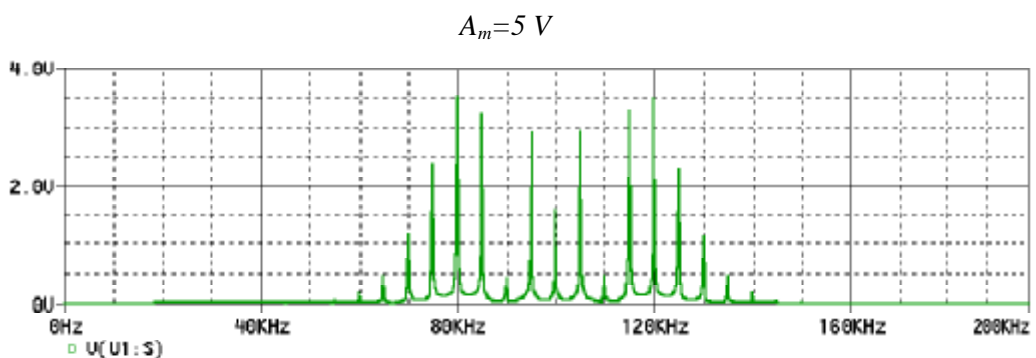
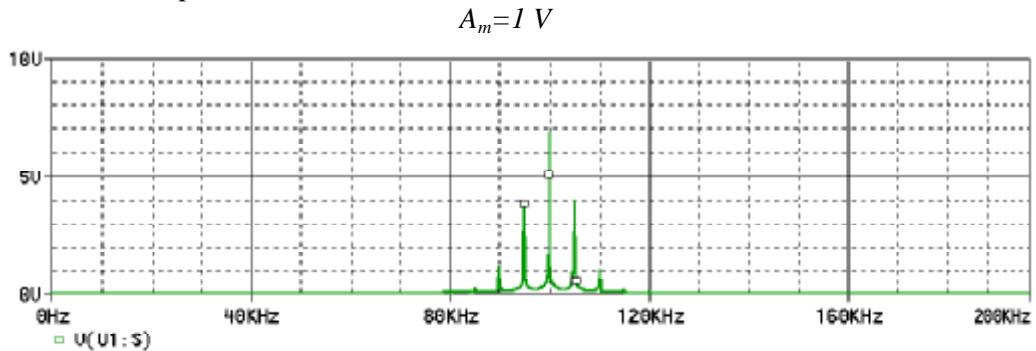
indice de modulation

→ Déclarer l'amplitude du signal d'entrée comme paramètre A_m .



→ Définir A_m comme paramètre global dans la fenêtre de paramétrage et choisir plusieurs valeurs

→ Afficher les différents spectres:



→ Calculer pour chaque cas l'indice de modulation

→ Conclure sur l'influence de m sur le spectre. (*largeur de bande, puissance transportée,...*)