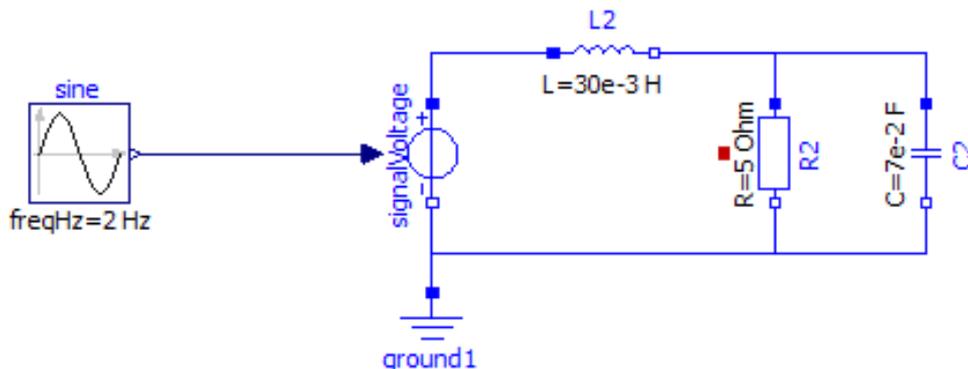
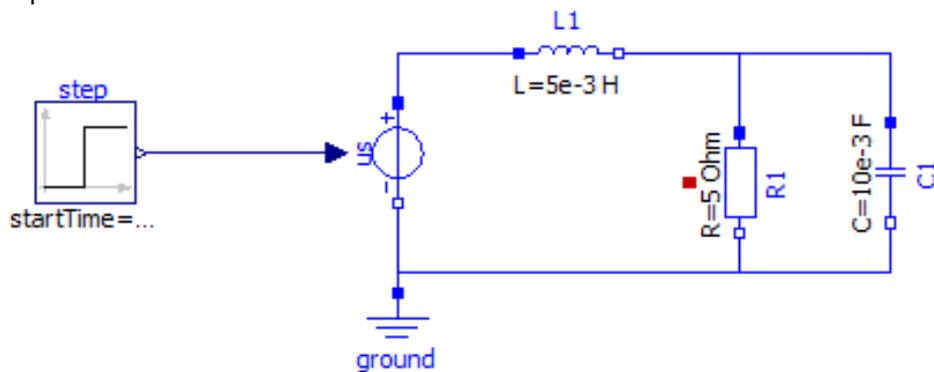


Cours :
Identifier et caractériser les grandeurs électriques
Ressources OpenModelica pour le cours

Les modèles OpenModelica nécessaires aux simulations sont fournis ci-après. Il suffit de faire un clic droit sur les images des montages électriques puis faire « enregistrer le lien sous... » ou « enregistrer la cible du lien sous... » selon le navigateur.

V – Comportement des modèles électriques en fonction de la fréquence du signal électrique sinusoïdal

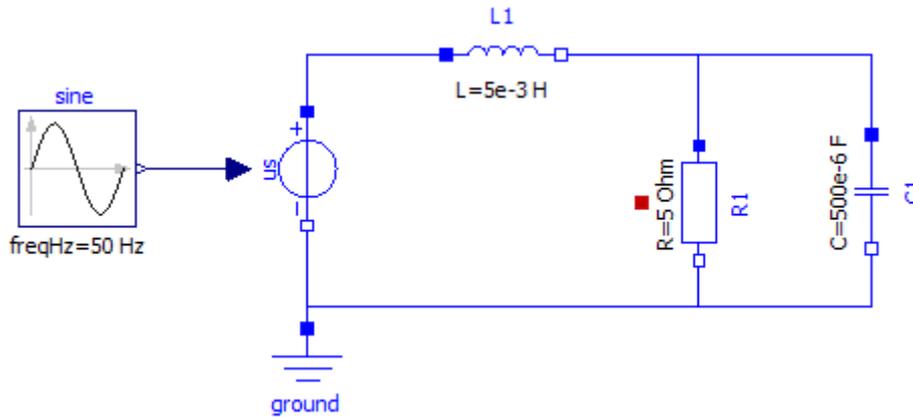
A partir des modèles ci-dessous, observer les régimes transitoires et permanents. Le premier montage est alimenté par une source continue, le deuxième par une source de tension alternative. Dans les deux cas la simulation correspond à une mise sous tension.



VI – Définitions des grandeurs efforts (tension) et flux (courant) instantanés dans les modèles électriques

Interprétation en alternatif sinusoïdal

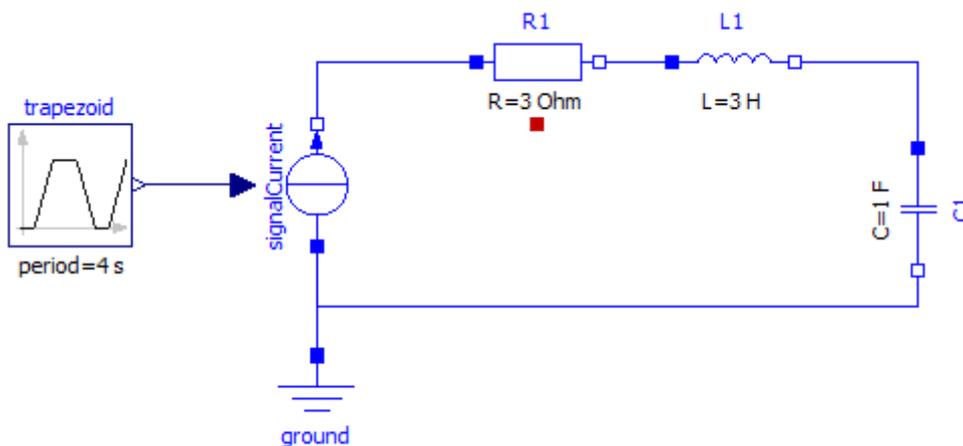
Observer les déphasages tension / courant dans les différents modèles élémentaires (résistance, inductance et capacité) à partir du modèle ci-après.



Interprétation sur signal non sinusoïdal (triangle)

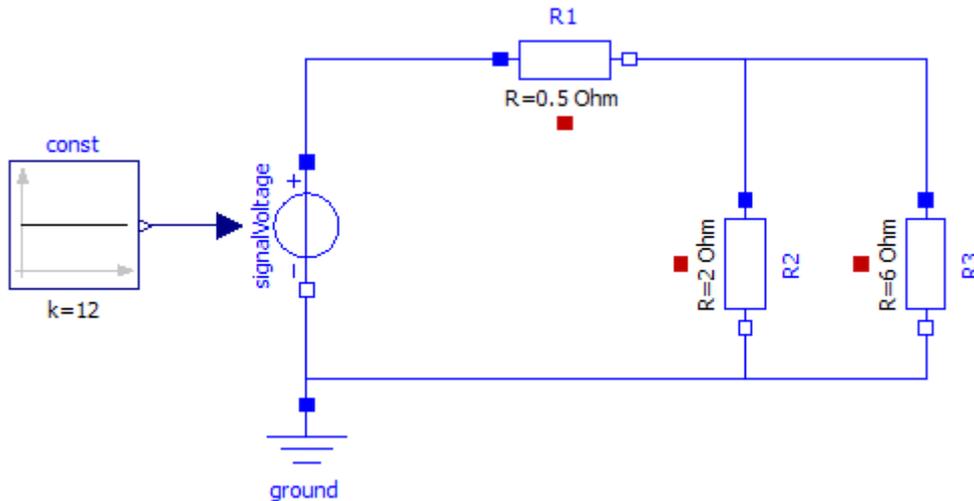
Observer les tensions aux bornes des différents modèles élémentaires (résistance, inductance et capacité) à partir du modèle ci-après.

Remarque la source est une source de courant, elle impose donc le courant (assez rare).



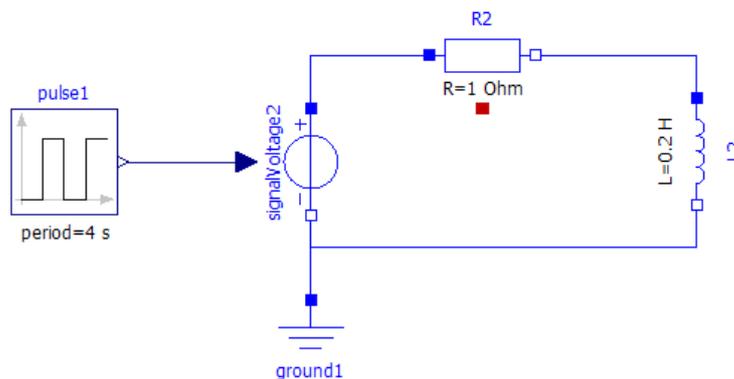
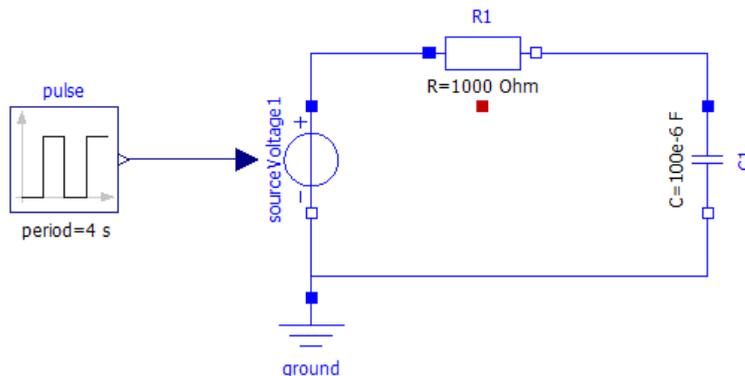
VII - Lois de Kirchhoff (Loi des mailles et loi des nœuds)

A partir du circuit ci-après, vérifier la validité des lois de Kirchhoff en **courant continu**.



IX – Expressions mathématiques, Equations différentielles

A partir des circuits suivants, observer l'évolution de l'intensité et des tensions aux bornes des modèles élémentaires électriques (R, L et C)



Ressource pour tracer avec un tableur $u_c(t)$ à partir de l'équation différentielle résolue dans le cours : [chargeCircuitRC_eleve.ods](#)

Ressource pour tracer avec Python (via Google Colaboratory) à partir de l'équation différentielle résolue dans le cours : [chargeCircuitRC_eleve.ipynb](#)