INP TOULOUSE ENSEEIHT



Élément de module : POO

Filière : $2^{\grave{e}me}$ année EEEA Option EEES

Rapport Bureau d'Étude Programmation Orienté Objet

Réalisé par :

YACOUBI Mohamed Reda

Encadré par :

Pr. POIRIER Jean-René

Table des matières

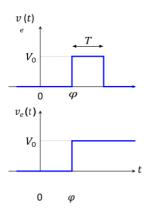
Introduction	3
La classe source	3
La classe solution	4
Simulations	5

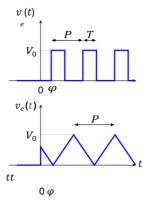
Introduction

L'objectif de ce BE est de développer en programmation orientée objet la simulation de circuits électroniques répondant à différents types de sources. Ainsi la résolution par différentes méthodes numériques des équations différentielles modélisant l fonctionnement de ces circuit.

La classe source

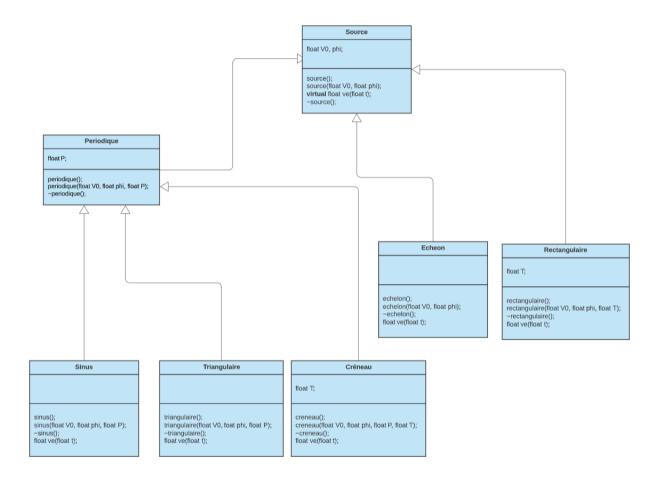
Dans cette classe, on va définir les différents types de sources suivant : Réctanglulaire, triangulaire, échelon, créneau ainsi que le sinusoidal)





Pour cela, nous avons créer d'abord un organnigramme pour bien comprendre l'architecture de notre code et faciliter la programmation tout en utilisant les différentes notions vue au cours. (Héritage, l'encapsulation, polyomorphisme...

\rightarrow Organigramme



La classe solution

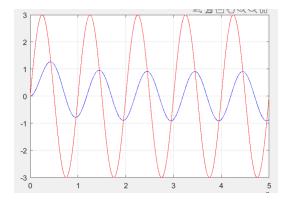
Afin de simuler la réponse des circuits électroniques excités par les différentes sources définies dans la classe source. Nous devons faire la résolution d'équations différentielles représentant la réponse des circuits par différentes méthodes numériques à savoir Euler, Range kutta ...

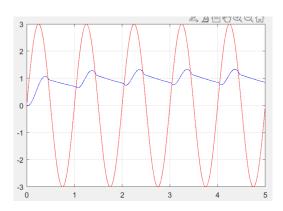
\rightarrow Organigramme Solution float u0, u1, tf; int n; void sol_euler1(); void sol_euler1(); void sol_euler2(); void sol_heun(); void range_kutta4(); virtual float fct1(float u, float t); virtual float fct2(float u, float u1, float t); circuit1 solution(); solution(); solution(); Circuit2 circuit1(); circuit1(float u0, float tf, int n); -circuit1(); circuit2(); circuit2(float u0, float u1, float tf, int n); -circuit2(); Circuit_A Circuit_B Circui_C Circuit_D float R,C; float R,L,C; float R,L,C; flat R1, R2, C, V_BE; float fct1(float u, float t); $\label{eq:float_theorem} \begin{array}{l} \text{float fct1(float u, float t);} \\ \text{circuit_B(float R1,R2,, float C, source *s,} \\ \text{float u0, float tf, int n);} \end{array}$ $\label{eq:continuity} \begin{array}{ll} \text{float fct2(float u, Ffloat u1,float t);} \\ \text{circuit_C(f);} \\ \text{circuit_C(float R,float L, float C, source *s, float u0, float u1, float tf, int n);} \end{array}$ $\label{eq:loss_problem} \begin{array}{ll} \text{float fct2(float u, Ffloat u1,float t);} \\ \text{circuit_D(float R,float L, float C, source *s,} \\ \text{float u0, float u1, float tf, int n);} \end{array}$ circuit_A(); circuit_A(float R, float C, source *s, float u0, float tf, int n); ~circuit_A(): -circuit_D();

En effet, nous avons divisé l'étude sur deux sortes de circuits, circuit d'ordre 1 et circuit d'ordre 2, c'est pour cela nous avons créé deux sous classes nommées Circuit1 et Circuit2. Afin de simuler l'ensembles des circuits données en sujet, nous avons introduit différentes méthodes numériques en utilisant des fonctions virtuelles et du polymorphisme dynamique pour bien structurer le code.

Simulations

→ Circuit A et B lorsque l'entrée est un signal sinusoidail :





\rightarrow Circuit A et B à un échelon :

