

Cartouche du document

Année : ING 1 - Matière : UML - Activité : Travail dirigé

Objectifs

Il s'agit dans ce travail dirigé d'étudier la modélisation la vie d'un objet à travers les diagrammes d'états.

Sommaire des exercices

- 1 - Les objets Livre et Adherent du système Bibliothèque
- 2 - Description d'un stimulateur cardiaque

Corps des exercices

1 - Les objets Livre et Adherent du système Bibliothèque

Énoncé :

On reprend dans cet exercice la gestion de la bibliothèque. Plus particulièrement à la vie des objets de la classe ExempleLivre et ceux de la classe Adherent.

On ajoutera la classe ExempleLivre. Le livre est l'oeuvre et un exemplaire de livre correspond à une production physique du livre.

On rappelle quelques règles de gestion de la bibliothèque.

- Un adhérent peut emprunter des livres.
- Il ne peut avoir au plus cinq emprunts en cours.
- S'il tarde trop à rendre un livre, il est suspendu d'emprunt pendant 1 semaine.
- On rappelle régulièrement un adhérent suspendu pour qu'il rende les exemplaires hors délai.

Question 1)

Énoncé de la question

Représenter le diagramme d'états d'un objet de la classe ExempleLivre.

Question 2)

Énoncé de la question

Représenter le diagramme d'états d'un objet de la classe Adherent.

2 - Description d'un stimulateur cardiaque

Énoncé :

Le fonctionnement de la classe Coil Driver (Pilote de la bobine)

- Par défaut, l'objet est dans l'état Desactive.
- Il est active quand l'objet Reed Switch se ferme. Cet état Activé est un super état.
- Si le pilote est dans l'état Activé et que le Reed Switch s'ouvre, le pilote passe dans l'état Desactive.
- Diagramme d'état du super état Activé
 - Le premier état est l'état Pause.
 - Si on est dans l'état Pause et que la bobine envoie une impulsion electro-magnetique, on passe dans l'état Reception Bit. On initialise deux variables Count et Byte à 0 et un timer est lance pour un temps de tps1.
 - Quand le pilote de bobine est dans l'état Reception Bit, il attend un certain temps (time out).

- Si le pilote de la bobine reçoit une impulsion avant le time out, alors on incrémente de 1 la variable count et on relance le timer.
- Si le temps est écoule avant une autre impulsion alors on passe dans l'état Attente Bit.
 - On décode la variable Count pour obtenir un bit et on place ce dernier dans la variable Byte en cours de construction.
 - On lance un timer pour un temps tps2.
 - Si la variable Byte est pleine, on envoie un signal à l'objet Gnome pour lui signaler qu'il va recevoir la variable Byte et on remet la variable Byte à 0.
- Dans l'état Attente Bit :
 - Si aucune impulsion n'est envoyée avant le time out, on passe dans l'état Pause et on lance un timer pour un temps tps3 pour espacer les transmissions de paquets (les bits de Byte).
 - Si une impulsion est envoyée avant le time out, alors on initialise la variable Count à 0 et on passe dans l'état Reception Bit.
- Dans l'état Pause en venant de l'état Attente Bit, s'il y a un time out avant une impulsion, on passe dans l'état Attente de transmission.
- Transmission des bits
 - La transmission se fait bit par bit après un certain temps d'attente tps4. Quand ce temps s'est écoule, on calcule le bit à envoyer et on passe dans l'état Transmission Bit.
 - Après la transmission du bit, on passe dans l'état Pause si le dernier bit a été envoyé ou dans l'état Attente de Transmission sinon.

Question 1)

Énoncé de la question

Etablir le diagramme d'états du pacemaker