

Συστήματα Γνώσης

Ακαδημαϊκό Έτος: 2024 – 2025

Εργασία 1: Εργασία σε Protege (Ανάπτυξη Οντολογίας Ηλεκτρονικού Κυκλώματος)

Μαρία Μακρή 4225

Ελένη Ευαγγελία Κούμπου 4065

Στο πλαίσιο της εργασίας για την ανάπτυξη οντολογίας ενός ηλεκτρονικού κυκλώματος, σχεδιάστηκαν και υλοποιήθηκαν κλάσεις που μοντελοποιούν τα συστατικά στοιχεία ενός συστήματος γνώσης. Ακολουθεί μια ανάλυση των κλάσεων που αναπτύχθηκαν, συμπεριλαμβανομένων των ιδιοτήτων τους, των περιορισμάτων τους, καθώς και των σχέσεων τους με άλλες κλάσεις της οντολογίας.

Κλάσεις:

1. System:

- Ρόλος:** Αντιπροσωπεύει μια γενική κλάση που ενσωματώνει και διαχειρίζεται τα διάφορα συστατικά στοιχεία, όπως τα ηλεκτρονικά εξαρτήματα, τους αισθητήρες και τις εισόδους/εξόδους.
- Ιδιότητες:**
 - hasInput(required multiple): Αναφέρεται στις εισόδους του κυκλώματος. Συνδεεί τη κλάση System με την κλάση Input.
 - hasOutput(required multiple): Αναφέρεται στις εξόδους του κυκλώματος. Συνδεεί τη κλάση System με την κλάση Output.
 - hasSensor(required multiple): Αναφέρεται στους αισθητήρες που παρακολουθούν την λειτουργία του κυκλώματος. Συνδεεί τη κλάση System με την κλάση Sensor.
 - hasComponent(required multiple): Αναφέρεται στα εξαρτήματα που συνθέτουν το κύκλωμα (adder,multiplier). Συνδεεί τη κλάση System με την κλάση ElectronicComponent.

2. ElectronicComponent: με υποκλάσεις τις Adder & Multiplier

- Ρόλος:** Περιγράφει τα επιμέρους εξαρτήματα που αποτελούν το κύκλωμα. Τα εξαρτήματα αυτά μπορεί να είναι είτε αθροιστές είτε πολλαπλασιαστές.
- Ιδιότητες:**
 - hasInput(required multiple): Αναφέρεται στις εισόδους του συστήματος. Συνδέει τη κλάση ElectronicComponent με την κλάση Input και Sensor.
 - hasOutput(required multiple): Αναφέρεται στις εξόδους του συστήματος. Συνδέει τη κλάση ElectronicComponent με την κλάση Sensor και Output.
 - hasStatus(single): Καθορίζει την κατάσταση του εξαρτήματος (Κανονική Λειτουργία,Βραχύκυκλωση,Αποκοπή του πιο σημαντικού ψηφίου).
 - MonitoredBy(multiple): Καθορίζει ποιοί αισθητήρες παρακολουθούν τη λειτουργία του εξαρτήματος. Συνδέει τη κλάση ElectronicComponent με την κλάση Sensor και Output.
 - ComponentTo(multiple): Καθορίζει σε ποιο σύστημα ανήκει το κάθε εξάρτημα (adder,multiplier). Συνδέει τη κλάση ElectronicComponent με την κλάση System.

3. Adder:

- **Ρόλος:** Υπεύθυνη για την εκτέλεση αριθμητικών προσθέσεων.
- **Ιδιότητες:** Κληρονομεί τις ιδιότητες της κλάσης ElectronicComponent.

4. Multiplier:

- **Ρόλος:** Υπεύθυνη για την εκτέλεση πολλαπλασιασμών μεταξύ αριθμητικών τιμών.
- **Ιδιότητες:** Κληρονομεί τις ιδιότητες της κλάσης ElectronicComponent.

5. Input:

- **Ρόλος:** Αντιπροσωπεύει τις τιμές που λειτουργούν ως είσοδοι στο σύστημα.
- **Ιδιότητες:**
 - a) `InputValue(required multiple)`: Καθορίζει την αριθμητική τιμή της εισόδου.
 - b) `InputTo(required multiple)`: Ορίζει για ποιο εξάρτημα προορίζεται η τιμή εισόδου. Συνδέει την κλάση Input με την κλάση ElectronicComponent και System.
 - c) `MonitoredBy(multiple)`: Καθορίζει ποιοι αισθητήρες παρακολουθούν τη λειτουργία του εξαρτήματος. Συνδέει τη κλάση Input με την κλάση Sensor και Output.

6. Output:

- **Ρόλος:** Αντιπροσωπεύει το τελικό αποτέλεσμα του κυκλώματος.
- **Ιδιότητες:**
 - a) `OutputValue(required multiple)`: Καθορίζει την αριθμητική τιμή της εξόδου.
 - b) `OutputTo(required multiple)`: Ορίζει για ποιο εξάρτημα προορίζεται η τιμή εξόδου. Συνδέει την κλάση Output με την κλάση ElectronicComponent και System.
 - c) `MonitoredBy(multiple)`: Καθορίζει ποιοι αισθητήρες παρακολουθούν τη λειτουργία του εξαρτήματος. Συνδέει τη κλάση Output με την κλάση Sensor και Output.

7. Sensor:

- **Ρόλος:** Αντιπροσωπεύει τους αισθητήρες που παρακολουθούν τη λειτουργία των εξαρτημάτων του συστήματος. Οι αισθητήρες είναι απαραίτητοι για την παρακολούθηση και τη διασφάλιση της σωστής λειτουργίας του συστήματος, καθώς και για την ανίχνευση προβλημάτων.
- **Ιδιότητες:**
 - a) `Measures(required single)`: Καθορίζει την αριθμητική τιμή του αισθητήρα.
 - b) `Monitors(required single)`: Ορίζει το εξάρτημα που παρακολουθεί ο αισθητήρας. Συνδέει την κλαση Sensor με την κλάση ElectronicComponent και Input και Output.
 - c) `SensorTo(required multiple)`: Καθορίζει σε ποιο σύστημα ανήκει ο κάθε αισθητήρας. Συνδέει τη κλάση Sensor με την κλάση System.
 - d) `hasStatus(single)`: Καθορίζει την κατάσταση του αισθητήρα(Κανονική Λειτουργία,Βραχύκυκλωση).

8. OperationCycle:

- **Ρόλος:** Αντιπροσωπεύει τους κύκλους(χρονικές στιγμές) στις οποίες λειτουργεί το σύστημα. Σε κάθε χρονική στιγμή καταγράφονται οι τιμές των εισόδων, των αισθητήρων και της εξόδου, οι οποίες θα δίνονται ως αντικείμενα στο πρόγραμμα.
- **Ιδιότητες:**
 - a) `hasInput1,2,3,4(required single)`: Αντιπροσωπεύει τις τιμές εισόδου Input 1-4.
 - b) `hasOutput(required single)`: Αντιπροσωπεύει την τιμή εξόδου OUT.
 - c) `hasSensorM1,M2,M3(required single)`: Αντιπροσωπεύει τις τιμές των αισθητήρων M1,M2,M3.

Η οντολογία έχει αναπτυχθεί με τέτοιο τρόπο ώστε να είναι γενική και να μην δεσμεύεται από το συγκεκριμένο κύκλωμα και την συγκεκριμένη τοπολογία.

Ακολουθεί το διάγραμμα της οντολογιάς:

