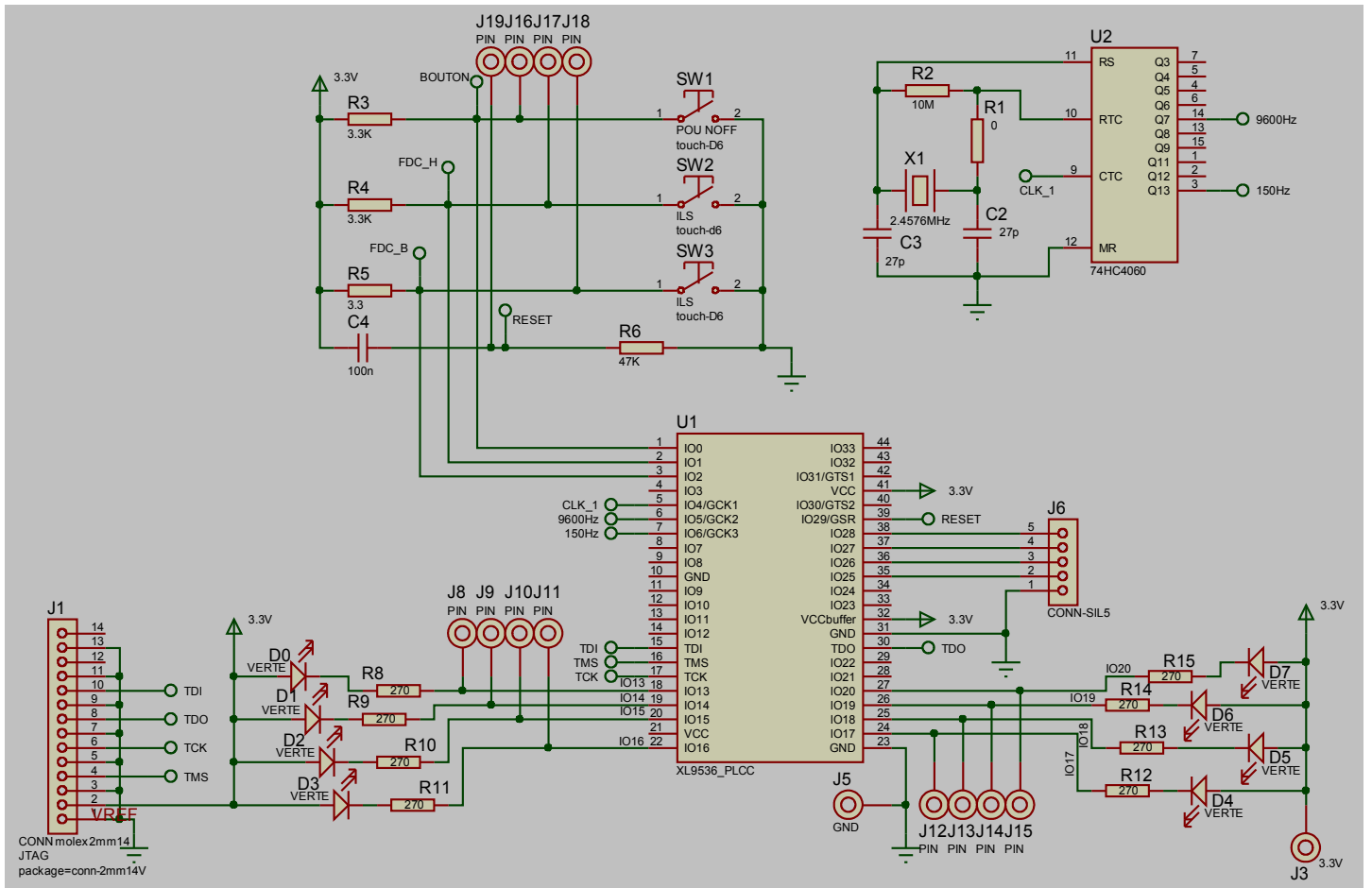
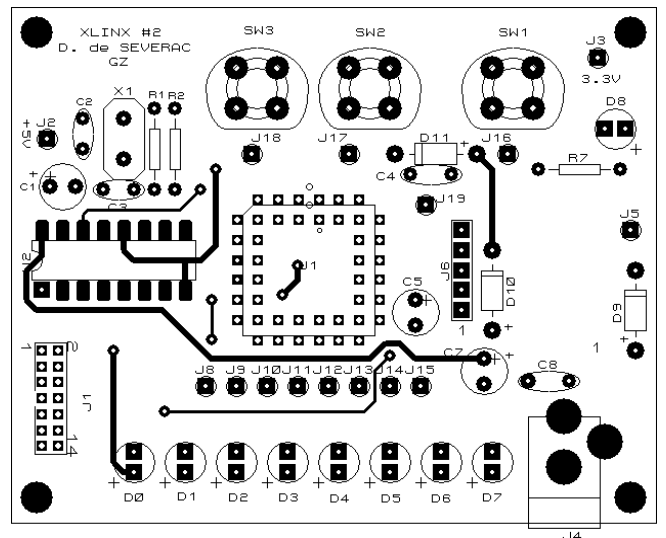
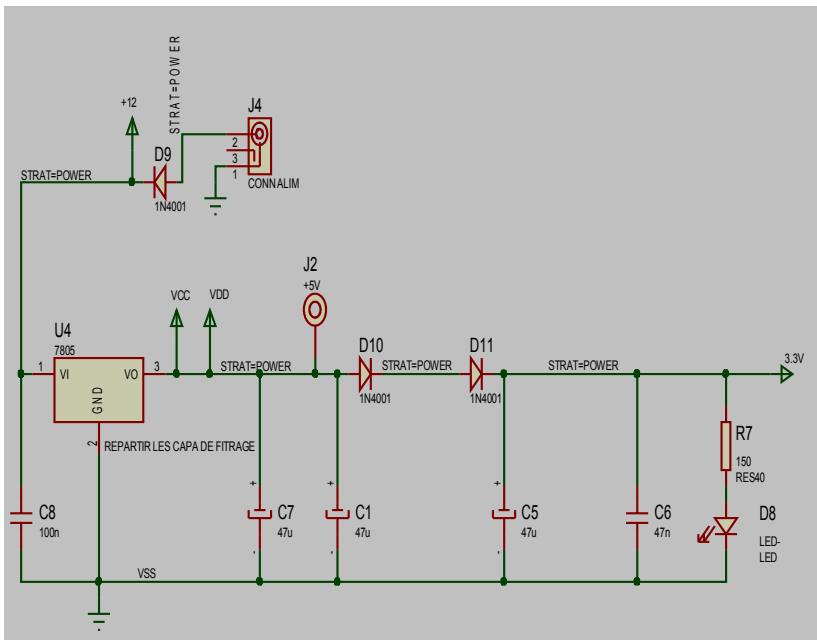


Maquette CPLD xilinx



Partie alimentation

Maquette vue de dessus



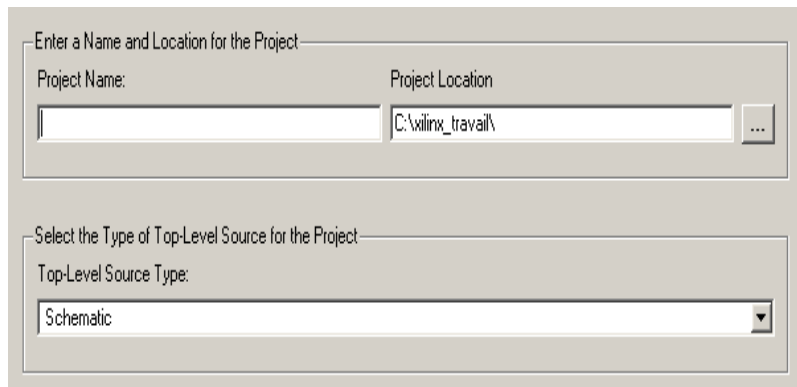
TPN°1

Travail prévu :

- Démarrer le logiciel Xilinx ISE, créer un projet, saisir le schéma : juste un comptage de l'horloge 150 Hz sur les 8 leds.
- Simuler pour vérifier le chronogramme obtenu
- Programmer la maquette, vérifier/expliciter/commenter vos observations

1 Démarrer un nouveau projet

- Dans le logiciel ISE faire : file >> new project on obtient la fenêtre ci-contre.
Dans la fenêtre project location Choisir un un dossier si possible sur un disque local de données du PC. Par ex : D:\nomduprof\xilinx\nomélève
Attention : pas d'espace ici donc ne pas utiliser Mes documents
- Donner un nom au projet par ex TP1comptage
- Top level (le fichier principal) est souvent schematic car il permet une interconnection graphique des différents sous modules (schémas, diagr d'états, code VHDL...)
- Next pour fenêtre suivante

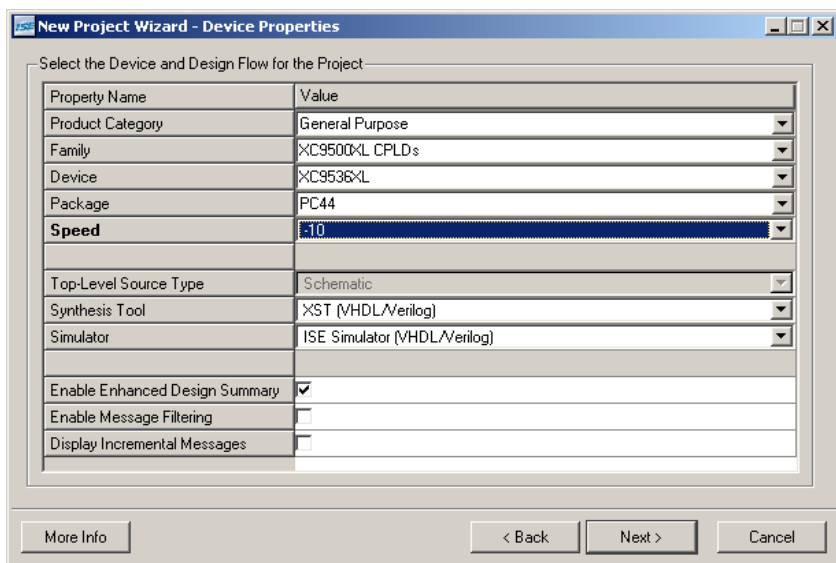


- Choisir alors (voir fenêtre ci-contre) le composant et le boîtier de la maquette (l'option XL précise 3,3V)
- Next pour fenêtre suivante

- Cliquer sur create a new source (on pourrait sauter cette étape pour prendre plus tard un fichier déjà existant. On utiliserait alors add copy of a source pour récupérer dans ce projet une copie issue d'un autre).

- valider l'icône schematic
- Donner un nom au fichier principal. Par ex: TP1.
- Fenêtre suivante Valider (si vous êtes satisfait du résumé fourni)
- Puis 2 fois next pour sortir sans ajouter de fichier supplémentaire
- Cliquer sur finish (après avoir vérifié le résumé à l'écran)

Remarque : Tant qu'on n'a pas cliqué sur finish tout peut être modifié.



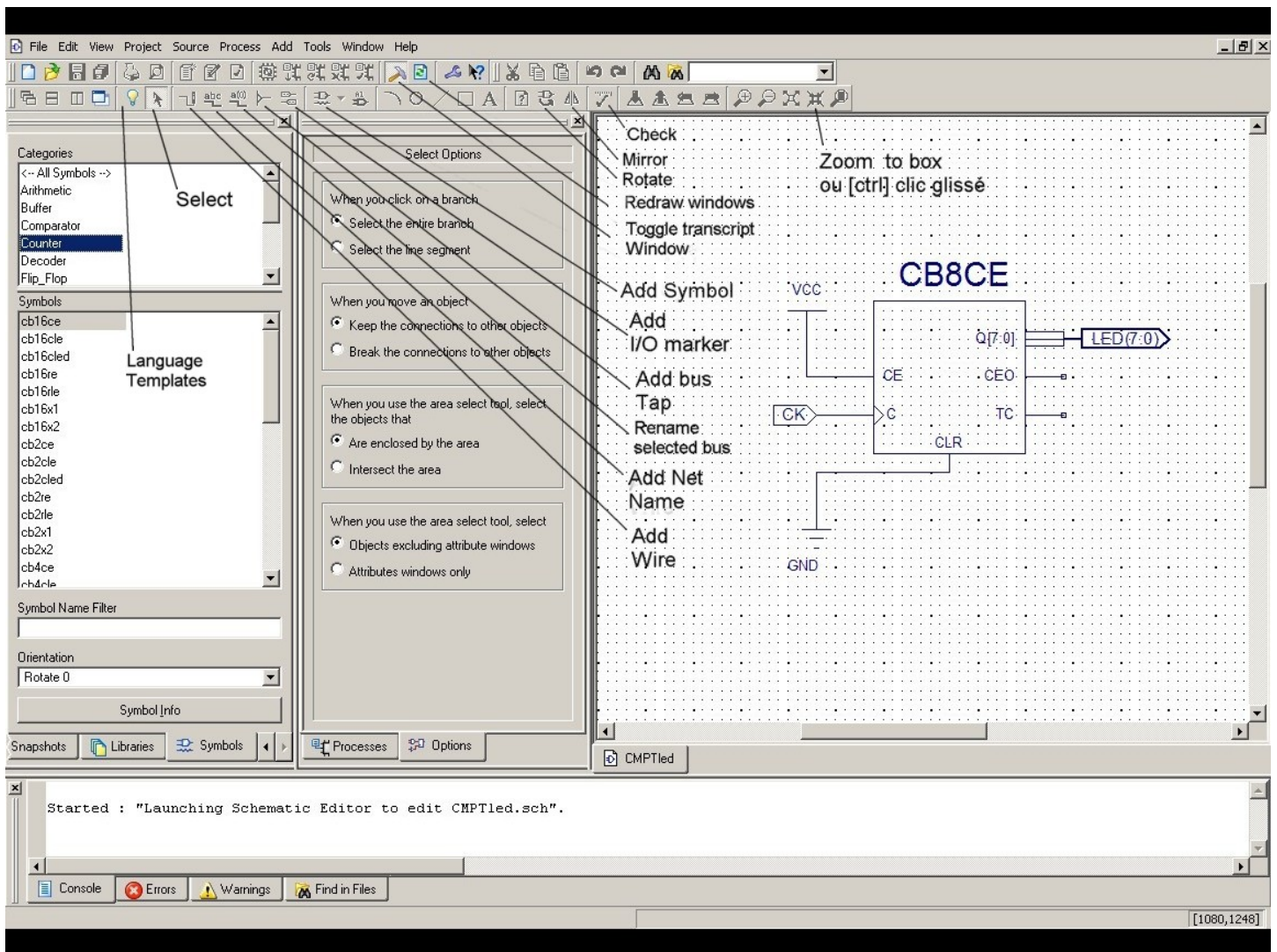
2 Saisie du schéma

Si tout s'est bien passé, on se retrouve dans l'éditeur de schéma (page suivante)

- Faire **view** >> layout vertically pour pour augmenter la taille des fenêtres qui serviront de sélecteur de composant et d'options d'édition.
Remarque : si des fenêtres sont rétives ou ont disparu, on peut toujours revenir à l'aspect de l'écran de départ avec **view** >> **restore default layout**
- Cliquer sur l'icône **Add symbol** (voir page suivante)
Choisir counter dans **categories**. Choisir CB8CE dans **symbols** (cascadable binary 8 bits counter). Le poser sur le schéma puis touche échap pour n'en mettre qu'un seul.
remarque : si on connaît le nom exact du symbol, on peut mettre all symbols dans categories puis taper le nom dans symbols filter
remarque : La **doc** du compteur est accessible par click droit>>object properties>> **symbol info**
- Ajuster le zoom avec **zoom to box** ou [ctrl] click glissé
- Ajouter les broches de **masse et d'alim** (et éventuellement le cartouche)
icône Add symbol puis general dans categories
- Ajouter les broches d'entrée/sortie : cliquer sur l'icône **add I/O marker** (input/output)
En général on peut laisser l'option add an automatic marker, il trouve tout seul si c'est une entrée ou une sortie et s'il faut rajouter un fil ou un bus
remarque : l'option remove marker enlève l'E/S en laissant le nom sur le fil (connection locale qui ne peut ressortir sur une broche ou en simulation). On l'utilise après une simulation pour que certaines sorties ne soient pas affectées à des broches du composant.
- **Renommer** vos broches d'E/S : **icône Select** puis double click ou clic droit>> object properties
par ex : le bus XLXN_3(7:0) en LED(7:0) et le fil XLN_4 en CLK
- Ajouter les connections manquantes : icône **Add wire**. Double click ou [échap] pour terminer

remarque : bien regarder les différentes options pour connecter (par ex : auto bus tap)
de même bien regarder les différentes options dans select pour **modifier des connexions** existantes (entire branch, line segment, keep connection...)

- Cliquer sur check pour vérifier votre schéma



3 Simulation

3.1 Créer un fichier de simulation

- Fermer le fichier schéma ou/et faire apparaître les onglets **sources** et **process** dans les fenêtres de gauche.
- En haut de la fenêtre source changer Synthesis/implementation en **behavioral simulation**
- clic droit sur votre fichier schéma >> new source >> clic sur test bench waveform et lui donner un nom. Valider l'association au fichier schéma.
- Pour ce premier test prendre 100ns pour les états hauts et bas de l'horloge.
- Un chronogramme apparaît sur l'écran mais les sorties n'ont pas été mises à jour
remarque : s'il y avait des entrées autres que l'horloge un clic souris permet la modification
remarque : pour une entrée complexe clic droit >> set value >> pattern wizard
- Enregistrer ce fichier

Lancer la simulation : banc de test

Onglet process >> double clic sur :

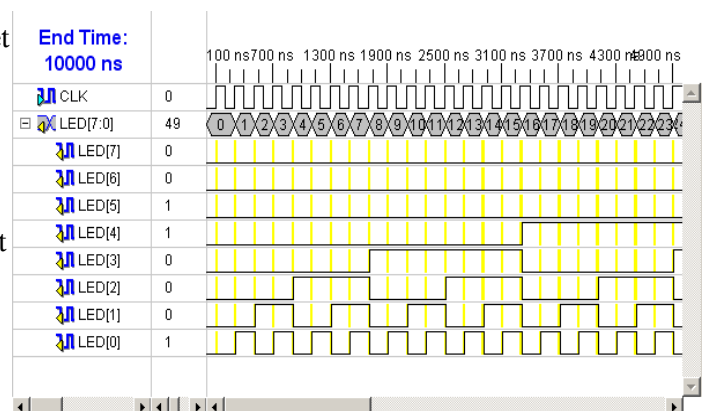
generate expected simulation results

Test bench >> set end of simulation permet de modifier la **durée** de simulation.

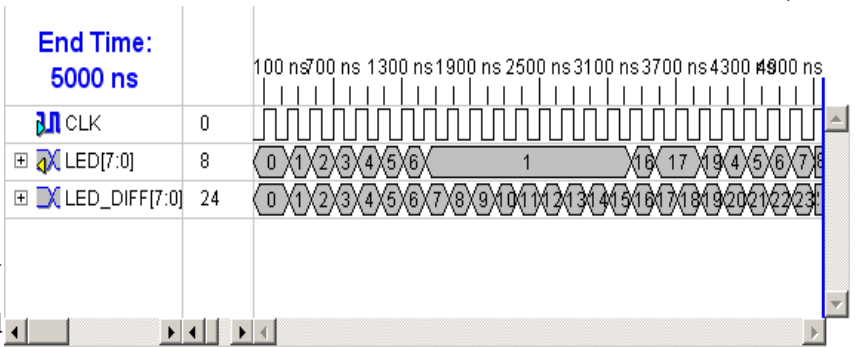
Observer le fonctionnement des marqueurs pour **mesurer des temps**. En déduire le retard de commutation prévu pour les sorties du compteur par rapport à l'horloge

Est-ce un comptage ou un décomptage (justifier)

Observer les changements possibles d'unités sur un bus (hexa signé ...)



On peut demander une **confrontation** entre résultats souhaités et simulés. Pour cela on édite aussi les sorties souhaitées, puis : generate expected simulation results, Quand il propose de modifier les sorties, répondez non, il duplique alors les signaux pour permettre la comparaison. exemple ci contre :



3.2 Simulation comportementale (behavioral simulation)

----- à faire -----

Si la simulation semble donner satisfaction, il reste à

4 Définir le brochage :

Pour que la synthèse puisse avoir lieu, il faut d'abord donner au moins une contrainte de **set up time** des

entrées par rapport à l'horloge : Onglet Sources Changer behavioral simulation en **Synthesis/implementation**.

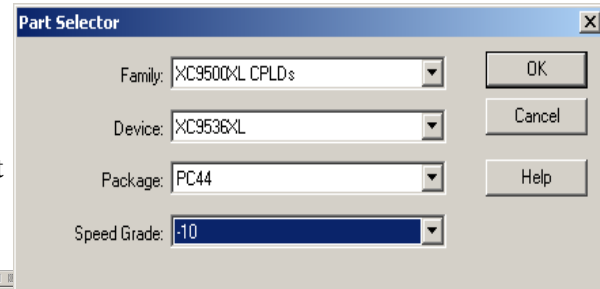
Marquer le fichier schéma pour faire apparaître dans process les actions le concernant.

Développer User constraint (clic sur +). Puis double clic sur **create timing constraint**. Double clic sur pad to set up (le dessin est explicite) 20ns devraient suffire. Enregistrer. Sortir

Le brochage sera ajouté au fichier des contraintes. Cette opération peut se faire graphiquement ou directement avec un éditeur de texte.

Méthode graphique :

Double clic sur **assign package pins**. Vous êtes alors dans le module PACE. Ouvrir le fichier UCF (user constraint file) associé au schéma (Il faut utiliser browse pour revenir dans le répertoire de votre projet). Spécifier la famille, le composant, et le boîtier.



Il suffit alors de faire glisser vos entrées de la liste en bas à gauche sur la broche concernée du boîtier.

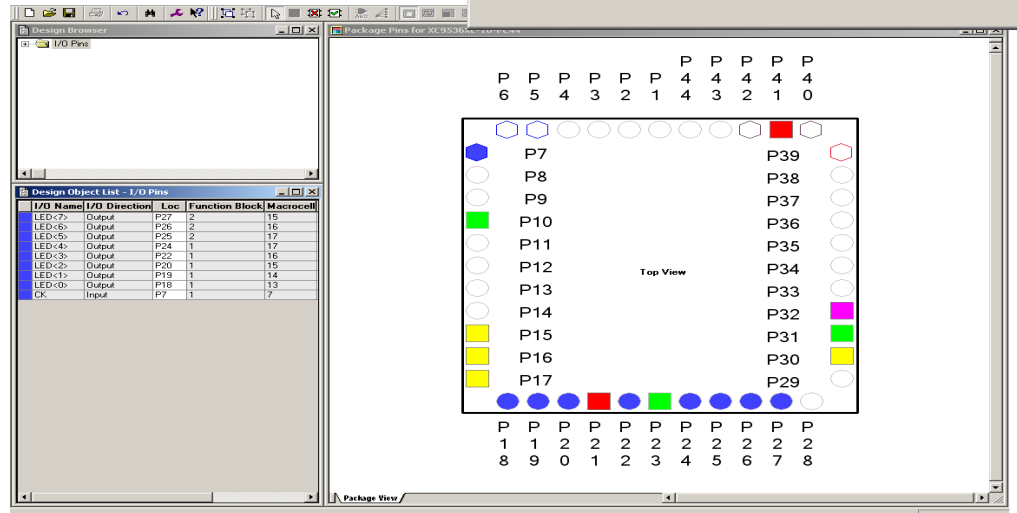
Enregistrer, quitter.

Vérifier éventuellement en ouvrant le fichier texte ucf

5 Programmer le composant

Développer **implement design** puis double clic sur **Fit**

Si tout s'est bien passé un compte rendu apparaît. Vérifier le brochage, l'absence de sortie non désirée en externe. L'absence de message d'erreur.



Si tout semble ok double clic sur **generate programming file**. Brancher maquette hors tension la sonde de programmation à la maquette. Brancher au PC. Mettre la maquette sous tension

Double clic sur **configure device impact**. Laisser l'option par défaut d'identification automatique : il identifie seul le type de composant et de sonde de programmation. Charger le fichier JED. Clic droit sur le composant>>program (laisser les options par défaut) et valider.

miracle ! Ca clignote Enfin !!!

- Est-ce un comptage, un décomptage justifier. Puis vérifier les signaux à l'analyseur logique.
- Pour une meilleure visualisation du motif sur les LEDs, ajouter un précomptage par 15 sur l'horloge du compteur. Ainsi on compte du 10Hz. Corriger les sorties en fonction de la question précédente. Simuler, programmer, vérifier la fréquence par une méthode de votre choix (expliquer)