

UTILISATION D'ISPLEVER NAVIGATOR

C'est un logiciel dédié à la programmation des circuits du fabricant de composants Lattice. Bien d'autres logiciels sont disponibles sur le marché, celui-ci est décrit à titre d'exemple.

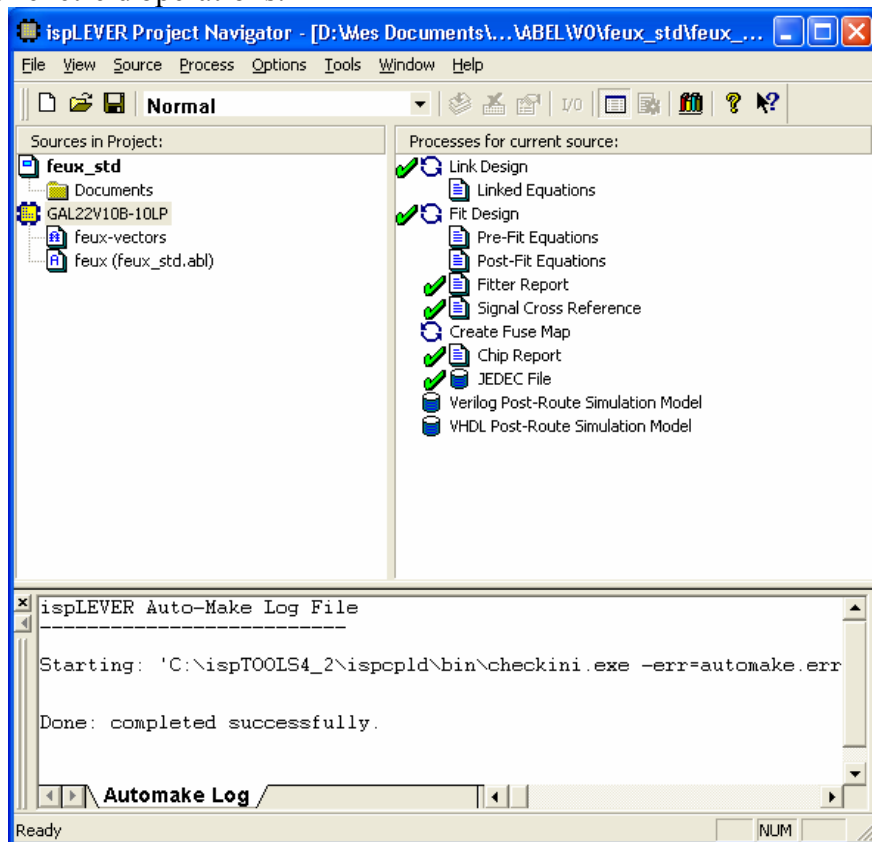
Nous donnons ici quelques informations clés pour sa prise en main rapide.

Sommaire:

1. Navigateur d'ispLEVER.....	2
2. Créer un projet 	3
3. Choisir un composant  ispGAL22V10B-10LJ	4
4. Créer un fichier ABEL ou VHDL	4
5. Affectation des numéros de broches.....	6
6. Créer un schéma	6
6.1. Placement des broches d'entrées sorties	6
6.2. Affectation des numéros de broches.....	7
6.3. Placement des noms d'équipotentiels d'entrées et de sorties.....	7
6.4. Placement des composants	7
6.5. Tracé des équipotentiels	8
7. Exécuter un test	8
8. Voir les temps de propagation de la fonction programmée	12
9. Comment créer le fichier JEDEC ?	13
10. Voir les informations résultant de la compilation pour le composant choisi.....	13
10.1. Équations pour la programmation	14
10.2. Statistiques d'occupations	14
10.3. Fichier JEDEC.....	14
11. Programmer un GAL isp	14
12. Programmer un MACH4	16
13. Raccourcis clavier de la saisie de schéma	17
13.1. Édition	17
13.2. Attributs	17
13.3. Visualisation	17
13.4. Ajouter un élément.....	17
Annexes.....	18

1. Navigateur d'ispLEVER

Il regroupe dans une seule interface tous les éléments nécessaires à la création d'un projet. Il comporte une fenêtre d'opérations.



Le projet "feux_std" contient les éléments sources suivants :



La définition d'un composant



Un document de test



Le document de description de la programmation

Un double clic sur une action donne accès à l'élément sélectionné.

Le navigateur associe à chaque élément du document source un certain nombre d'outils et de documents qui sont visibles en fenêtre d'opérations.

Chaque outil est repéré par un logo qui désigne :



Un outil



L'exécution d'une opération



La production d'un rapport



La production d'un fichier de sortie

Un double clic sur un outil lance l'opération sélectionnée. Suivant son déroulement le logo suivant est affiché :



L'exécution s'est correctement déroulée



L'exécution a échoué, des erreurs ont été détectées

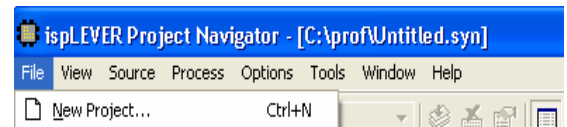


L'exécution est achevée mais des avertissements sont signalés

2. Créer un projet

Il est conseillé de créer un répertoire différent pour chaque projet. De plus prenez garde de ne pas donner le nom de votre projet à l'un de vos documents sources (par exemple feux_std.syn et feux_std.sch ou feux_std.abl).

Dans le menu “ **F**ile ” choisir “ **N**ew project ” :

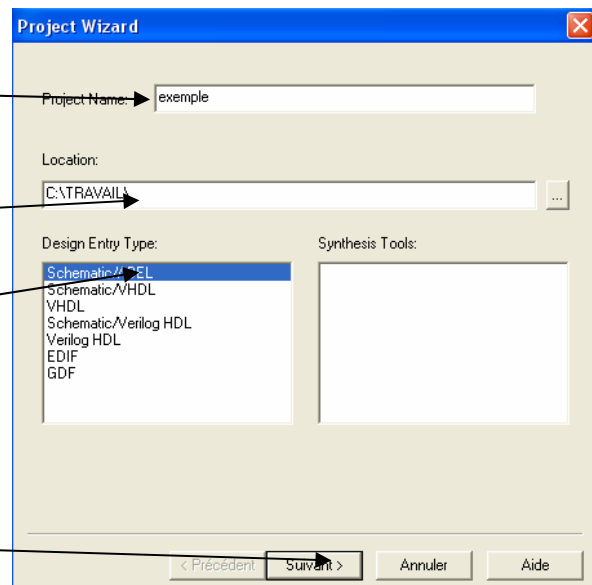


1) Nom du projet
Il portera par la suite l'extension .syn

2) Répertoire de travail

3) Choix du langage
(Ici, langage Abel)

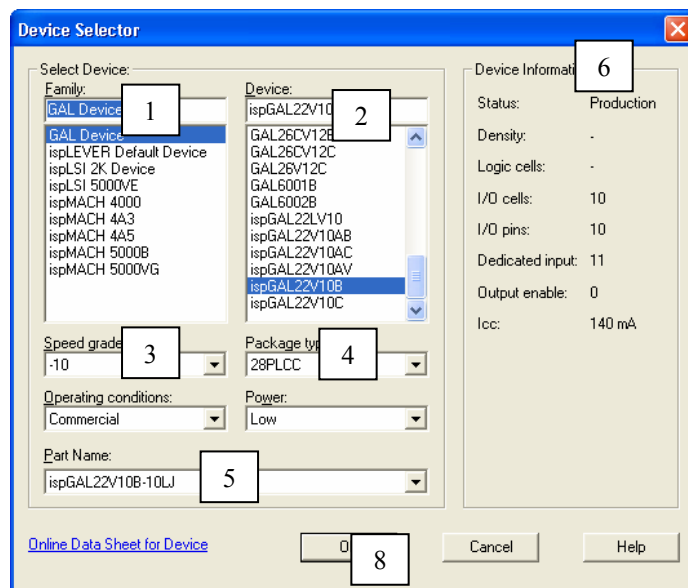
4) Suivant



La première étape de la création va consister à choisir le composant de l'application (voir ci-dessous).

3.Choisir un composant ispGAL22V10B-10LJ

On accède au choix d'un composant directement par le navigateur en cliquant sur le champ correspondant :



On pourra choisir dans l'ordre :

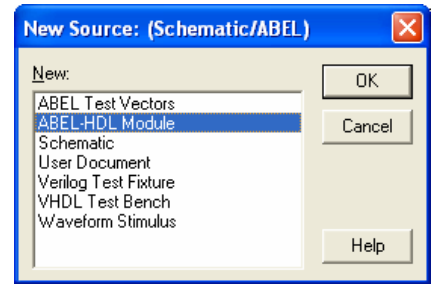
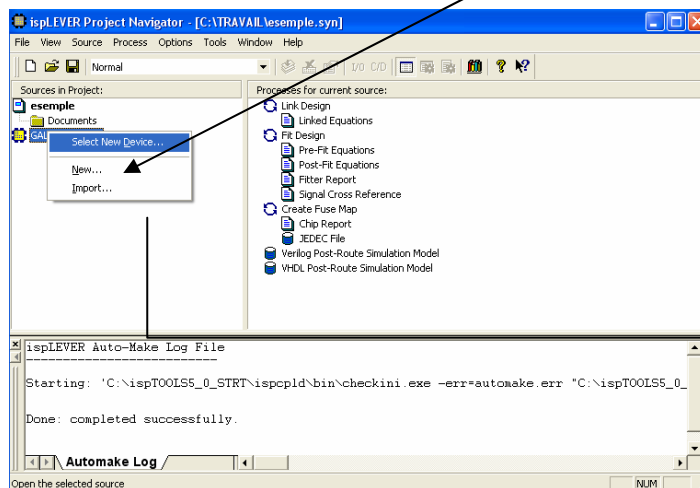
- 1 Une famille de composant PAL, MACH, ISP...
- 2 L'un des éléments de la famille,
- 3 Une vitesse de propagation,
- 4 Un type de boîtier,
- 5 Ou bien directement une référence commerciale.
- 6 Cette section affiche les paramètres du composant,
- 7 Terminer par OK.

4.Créer un fichier ABEL ou VHDL

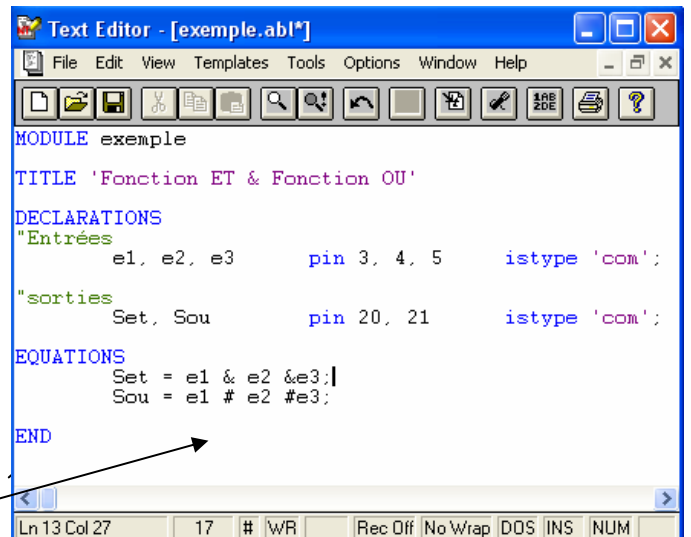
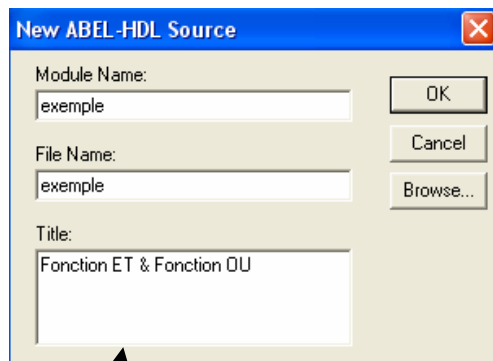
Pour créer et placer un fichier ABEL ou VHDL dans le projet, faire un clic droit dans la zone « Sources in Project », puis « New » et ABEL-HDL module.

Il vous suffit pour son écriture de respecter la syntaxe ABEL. La section de tests peut être placée dans un fichier à part (ABEL Test Vectors) ou dans votre fichier ABEL, tel que cela a été décrit. Que vous procédiez d'une façon ou de l'autre, et si votre fichier comporte une section de test, cet élément apparaîtra de façon distincte dans la fenêtre source.

1) Ouvrir le fichier : Cliquer droit sur le composant puis “ **Select New** ”



2) Sélectionner ABEL-HDL Module



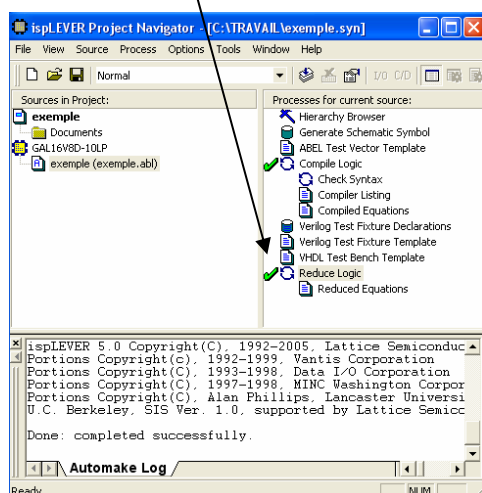
3) Remplir l'entête comme dans l'exemple donné.

4) Etablir les équations comme l'exemple ci-contre

5) Sauvegarder le fichier : **F**ile ⇒ **S**ave

6) Fermer

7) Exécuter l'étape “ **Reduce logic** ” : c'est l'étape du débogage et des fautes de syntaxe.



Rappel sur ABEL:

Syntaxe :

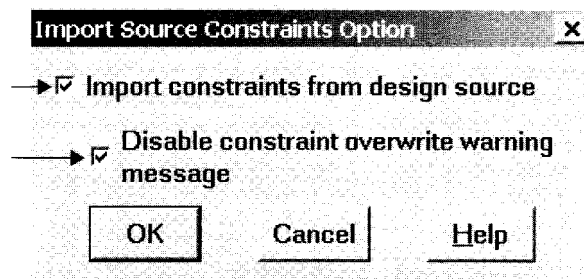
! ⇒ Opérateur NON

& ⇒ Opérateur ET

⇒ Opérateur OU

5. Affectation des numéros de broches

Pour certains composants comme le MACH4, elle est faite normalement par l'outil appelé Constraint editor. Si vous voulez imposer le brochage par votre fichier source ABEL (directive pin) ou par votre fichier de schéma, cela est possible à condition de cocher la case accessible par le menu Tools :

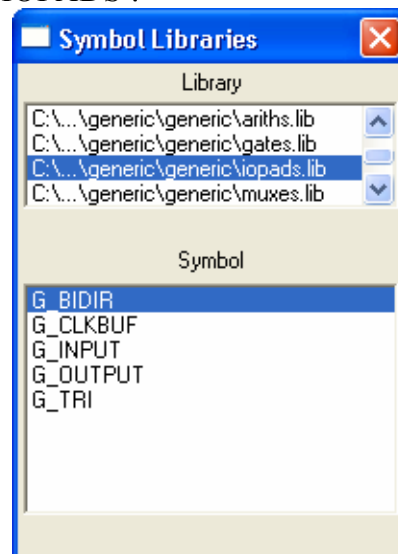


6. Créer un schéma

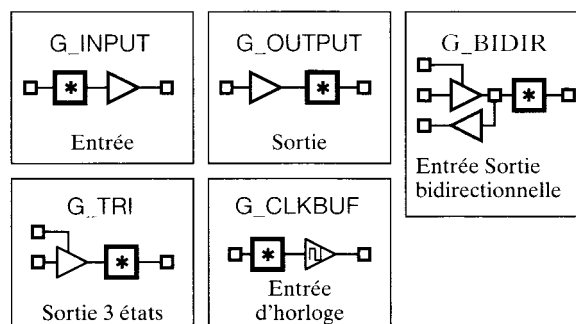
Dès que vous avez demandé la création d'un nouveau document source sous forme schématique, vous avez accès à sa fenêtre d'édition.

6.1. Placement des broches d'entrées sorties

C'est la première étape du placement, ces symboles sont accessibles par la commande Add/Symbol, dans la librairie IOPADS :



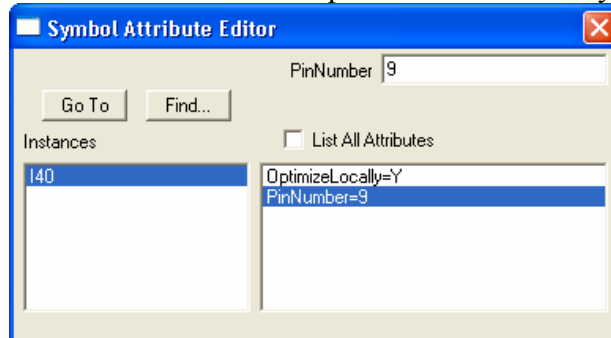
IOPADS contient les modèles d'entrées et sorties :



Il faut être attentif à respecter le modèle de broche convenable. Les broches entrées et broches sorties sont celles que vous utiliserez la plupart du temps.

6.2. Affectation des numéros de broches

Cette étape permet de donner un numéro à chacune des broches du composant. Vous devez procéder à ce travail en vous aidant du schéma structurel sur lequel cette information doit être mentionnée. On accède à la numérotation par Edit/Attribute/Symbol Attribute/:

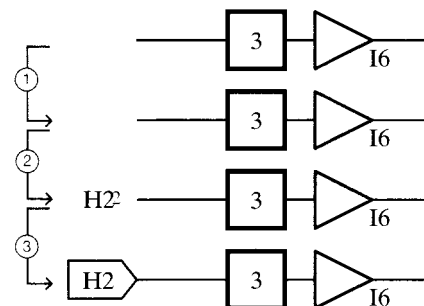
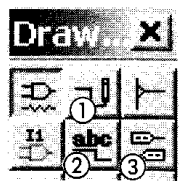


Lorsque vous sélectionnez une broche et que cette fenêtre est active, modifiez l'attribut SynarioPin avec le numéro souhaité.

6.3. Placement des noms d'équipotentielles d'entrées et de sorties

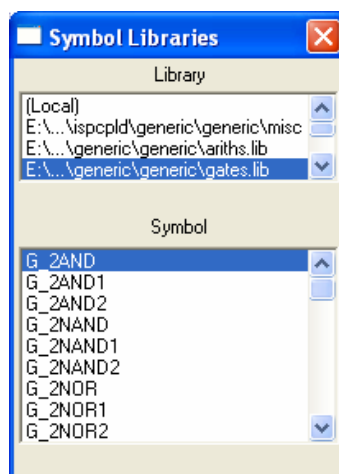
Trois opérations sont nécessaires pour cela ; sur chacune des entrées, placer :

- 1 un segment d'équipotentielle,
- 2 son nom (il doit être d'abord inscrit dans la zone de saisie en bas de fenêtre, puis cliquez à l'extrémité d'équipotentielle),
- 3 un marqueur d'entrée.

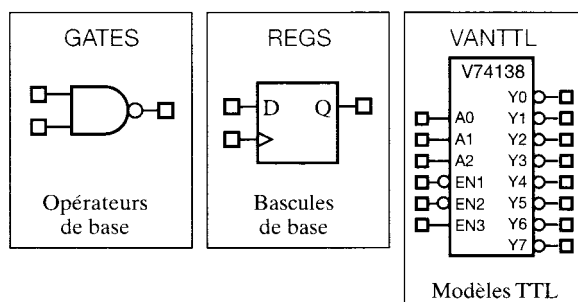


6.4. Placement des composants

L'accès au placement des composants se fait comme pour les entrées par Add/Symbol ou par F2.



Les bibliothèques disponibles sont entre autres : (schéma ci-dessous). Dans le cas où l'on part d'un schéma existant en technologie TTL, il est très simple de recopier celui-ci en utilisant les modèles de la bibliothèque VANTTL (uniquement avec des CPLDs).



Placez les composants de façon à simplifier le futur dessin des équipotentielles.

6.5. Tracé des équipotentielles

On accède à l'outil de tracé par la barre d'outils déjà mentionnée.

Lorsque le schéma est terminé sauvegardez ce travail, vous pouvez alors fermer cette fenêtre si vous le souhaitez.

7. Exécuter un test

Deux possibilités pour exécuter un test :

- soit à partir du "Test vector" d'un fichier abel,
- soit à partir d'un stimulus graphique.

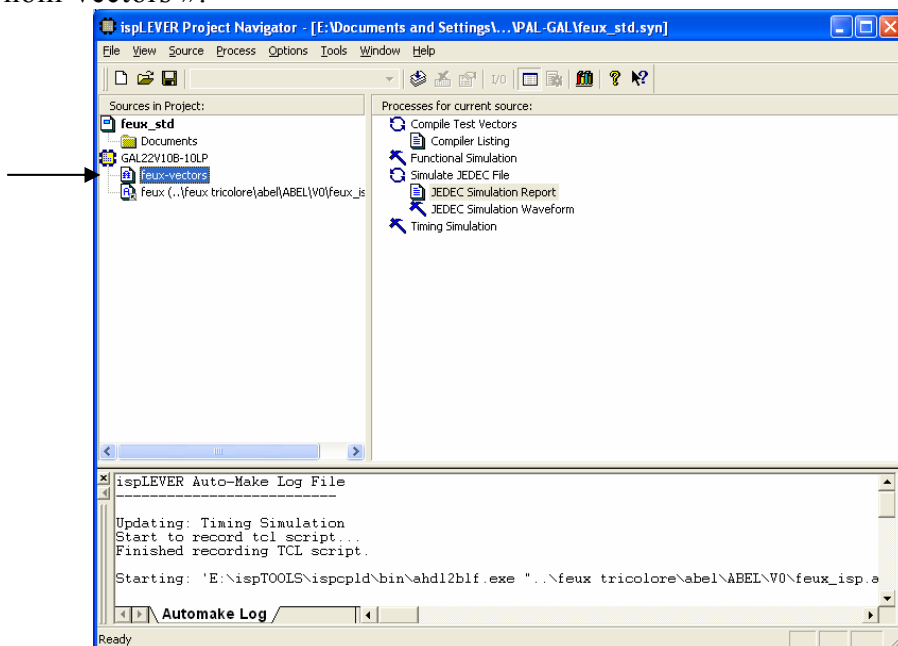
La suite des étapes de la simulation en dernière partie

A) A partir d'un "Test vector" d'un fichier abel

Le test est celui prévu par la section correspondante du fichier ABEL.

Si le document de description est un schéma, vous pouvez définir un fichier de test écrit selon la syntaxe ABEL. Ce fichier peut être créé par la commande New .../ABEL test Vectors. Vous le construirez de la même façon qu'un fichier ABEL en omettant la section d'équations. Les déclarations de broches (pin) doivent être présentes dans le fichier mais les numéros n'ont évidemment pas besoin d'être spécifiés.

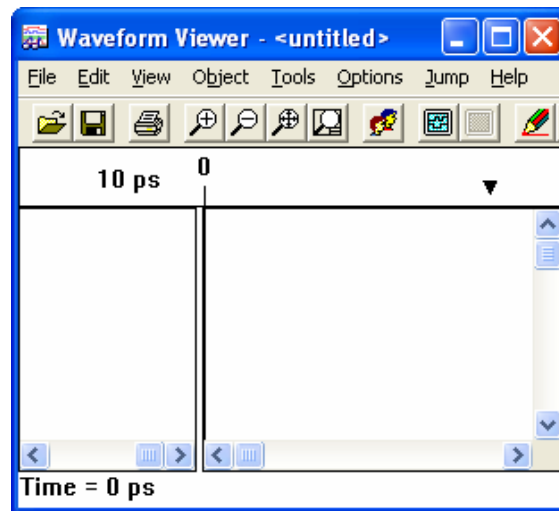
Lorsqu'une section de test est reconnue la fenêtre source comporte une ligne qui lui est dédiée « nom-vectors »:



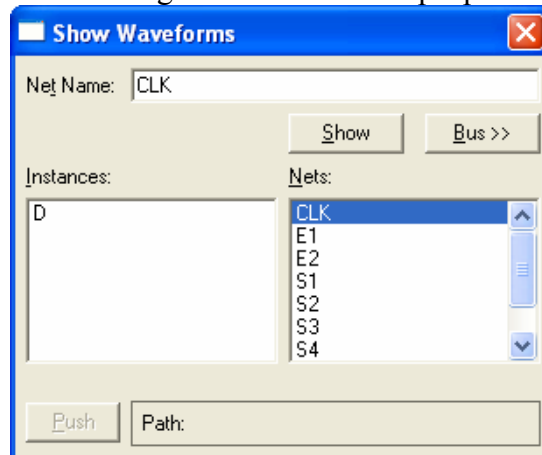
La fenêtre d'opérations fournit une série d'outils de tests.

L'exécution de la ligne « Simulate JEDEC File » et la visualisation de son rapport « JEDEC Simulation Report », lance le test prévu dans la section test du fichier ABEL. Si une anomalie est détectée, un avertissement est généré et un rapport d'exécution affiché.

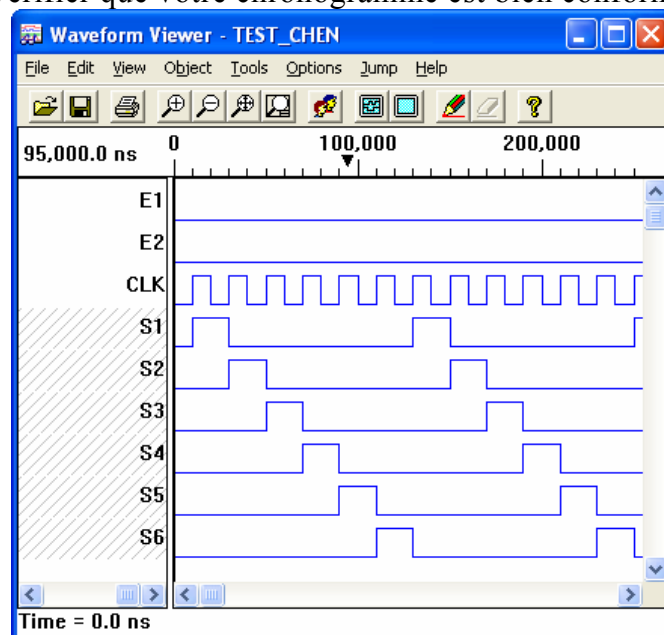
Les trois dernières lignes permettent la visualisation de chronogrammes à des niveaux différents. Un double clic sur la ligne « JEDEC Simulation Waveform » ouvre l'interface de représentation des chronogrammes « Waveform Viewer » :
 Dans laquelle vous pouvez afficher les signaux du projet.



Vous devez choisir les signaux dans la liste proposée : Edit/ Show...

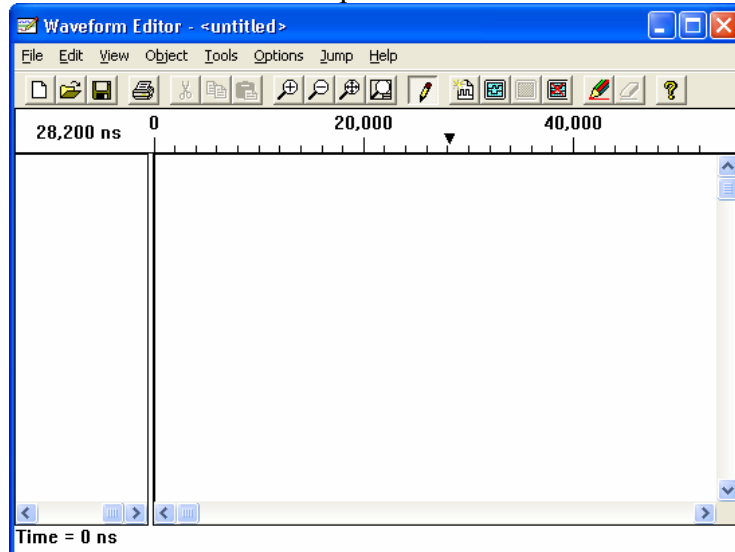


Vous pourrez alors vérifier que votre chronogramme est bien conforme à celui attendu :

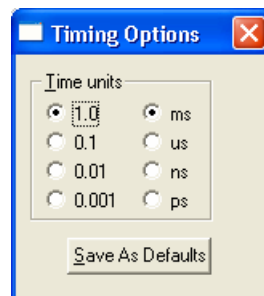


B) A partir d'un stimulus graphique

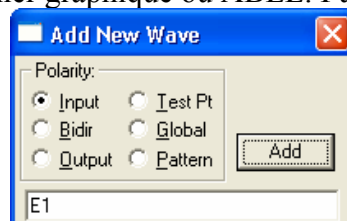
Il est possible de créer des stimulus pour les signaux d'entrées sous forme de saisie graphique. Lancez du menu "Window" d'ispLEVER "Waveform Editor".



Avant toute chose, il faut choisir l'échelle des temps, cela est fait dans le menu "Options", "Timing Options" :



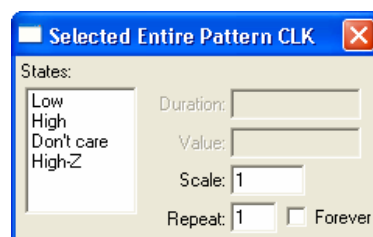
Du menu "Edit" sélectionner "New wave". Choisir le type entrée, puis donner le même nom de NET que celui utiliser dans le fichier graphique ou ABEL. Puis cliquer sur "Add".



Vous pouvez créer plusieurs signaux avant de fermer cette fenêtre.

Un clic sur le nom du signal permet de le sélectionner pour l'éditer. Placer le curseur sur la ligne à éditer, tout en utilisant la barre et le triangle d'indication des temps. Faire un clic, le signal se dessine. Si un nouveau clic est fait un temps plus tard sur l'échelle, le logiciel dessine un niveau logique automatiquement complété sur ce nouvel intervalle.

Il est possible de sélectionner une partie du signal en maintenant le bouton de la souris enfoncé. Mais avant vous devez aller dans le menu "Object" puis "Edit Mode" fait apparaître la fenêtre suivante :

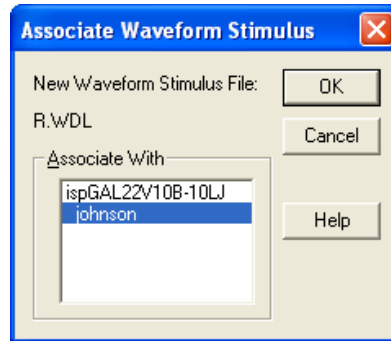


Elle permet d'éditer le signal sélectionné, et par exemple, de faire répéter le signal indéfiniment ("Forever"), option pratique pour faire un signal d'horloge.

Une fois tous les signaux d'entrées créés, sauvegarder votre travail (format wdl).

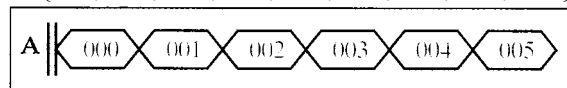
Dans la zone "Sources in project" d'ispLever, faire un clic droit puis "Import" pour insérer votre fichier de stimulus wdl.

Vous devez l'associer au fichier de description (ABEL, VHDL ou graphique) et non au composant :



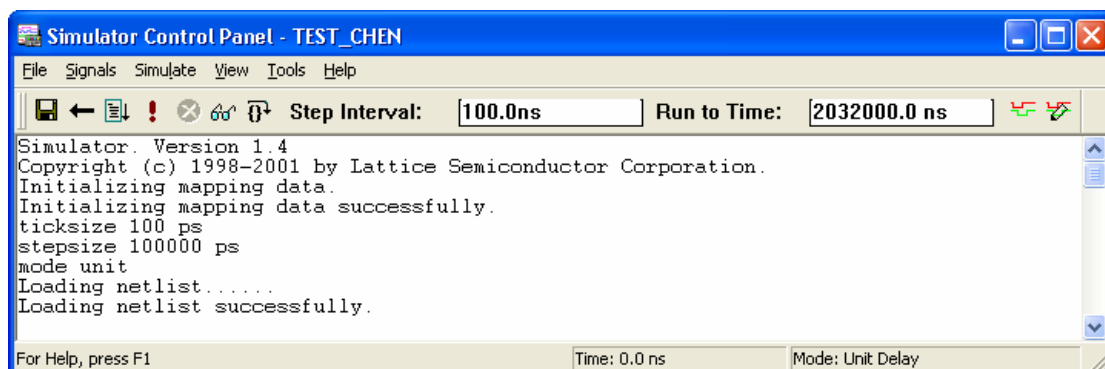
Remarque :

La fonction BUS permet de grouper les informations binaires. Exemple issu du compteur/générateur : $A = \{A8, A7, A6, A5, A4, A3, A2, A1, A0\} \rightarrow$

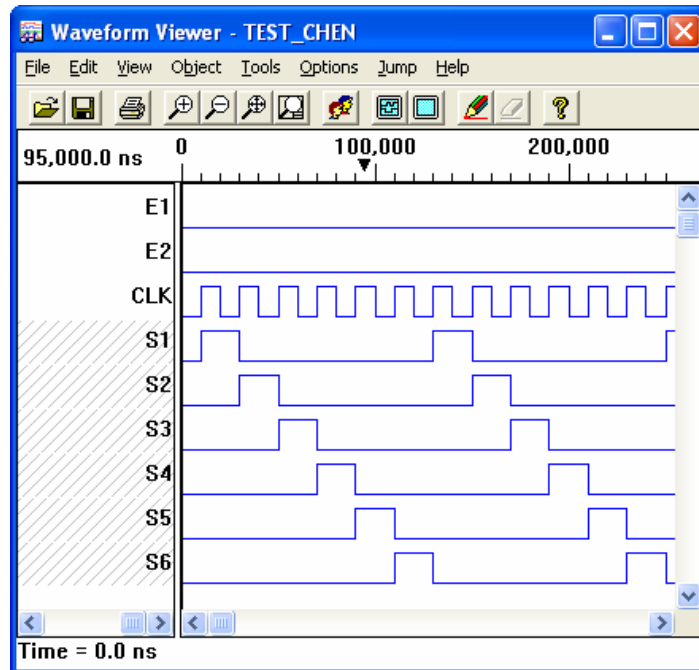


C) Dernières étapes de simulations

A ce stade, nous avons la description des stimulus d'entrées, il ne reste plus qu'à effectuer la simulation. Pour cela, faire un double-clic sur "⚡ Functional Simulation", s'il n'y a pas d'erreur, la fenêtre suivante doit s'ouvrir :



Pour lancer la simulation, faire un clic sur le point d'exclamation ("Run") pour visionner les fichiers de sortie dans "Waveform viewer" :



8. Voir les temps de propagation de la fonction programmée

Certains composants donnent accès à une évaluation directe des temps de propagation des informations dans l'opérateur programmé. Cette information peut être cruciale dans la conception des systèmes électroniques :

Cet exemple est obtenu à partir de l'exercice LOGIQUE grâce à la fonction timing analysis

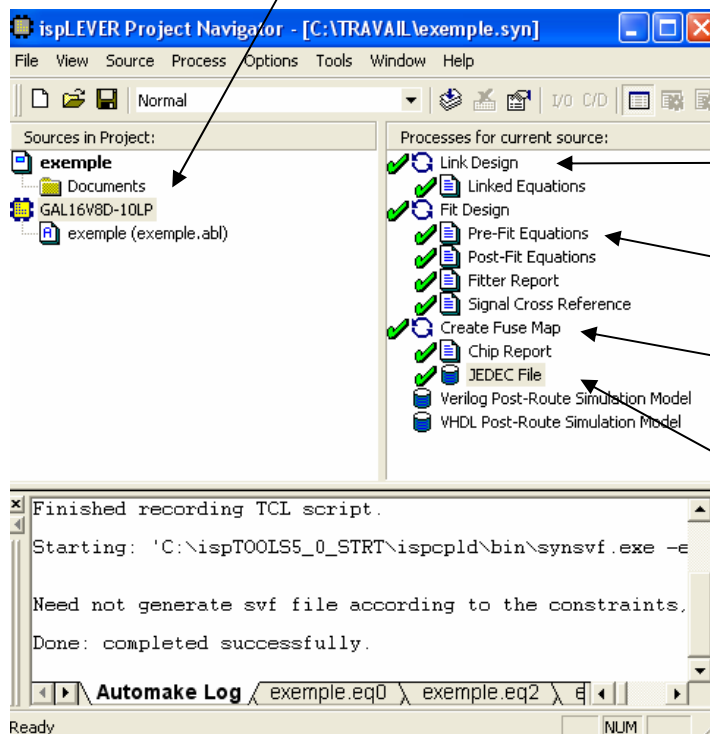
destination (ns)			
Source		del1	del8
	A	15.0	15.0
	B	15.0	15.0
	C	15.0	15.0
	D	15.0	15.0
	E	15.0	15.0

Ce tableau récapitule tous les temps de propagation dans la fonction programmée. Le temps de propagation entre un changement d'état en C et celui résultant en del8 est entouré.

9. Comment créer le fichier JEDEC ?

1) Sélectionner le circuit cible

2) Exécuter l'étape «  Create Fuse Map »



L'étape « **Link** » établit le lien entre les différentes sources d'équations et génère un fichier indépendant du circuit

L'étape « **Fit** » établit les relations entre les équations et le circuit (ici GAL 16V8)

L'étape « **Create Fuse Map** » établit la liste des fusibles à claquer

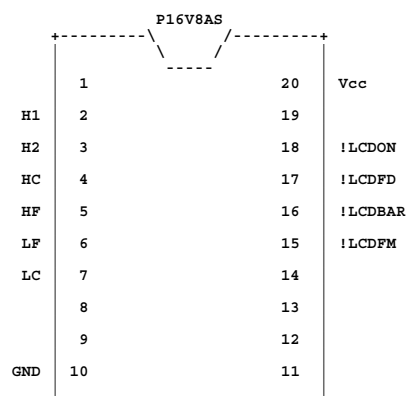
Un click gauche sur « **JEDEC File** » puis **View** permet d'afficher le contenu du fichier JEDEC.

10. Voir les informations résultant de la compilation pour le composant choisi

Vous devez sélectionner le composant dans la fenêtre source. Le document Chip report (PAL) ou Fitter report (MACH) nous montre entre autres informations :

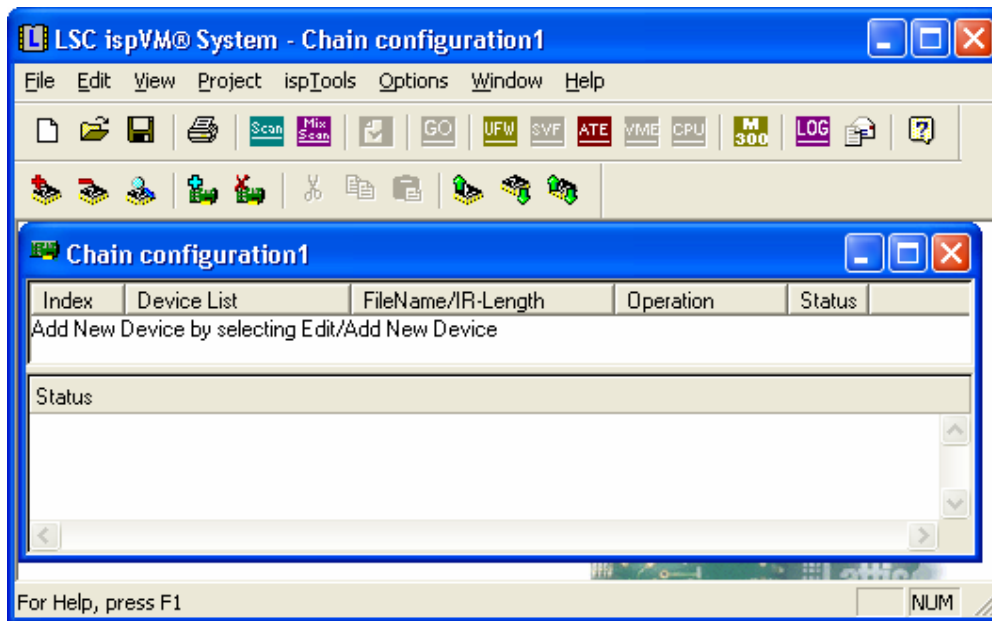
Brochage du composant

Exemple (extrait de Gal16V8) :

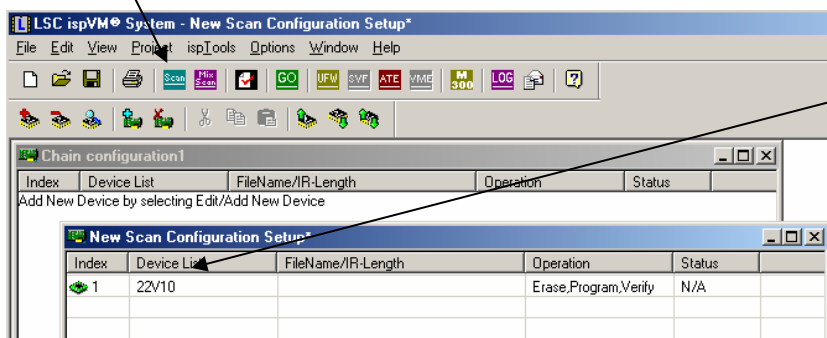


Une vérification est vivement conseillée à ce niveau, avant la mise en place dans le circuit électronique.

Attention : L'écriture « ! LCDON » ne signifie pas que LCDON est complémenté mais que la programmation est faite en utilisant le complément de sortie (voir ci-dessus).

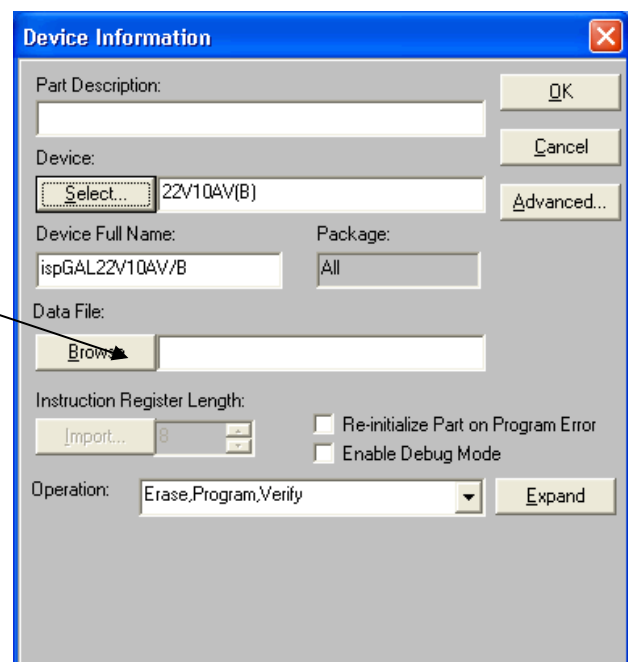


2) En appuyant sur **Scan** disponible dans la barre d'outils, le logiciel doit trouver le composant programmable de la maquette.



3) Double cliquer ensuite sur le composant, le menu suivant apparaît

4) Il faut alors choisir le fichier JEDEC à programmer dans le GAL : Il faut donc renseigner le champ **Data File** en utilisant la fonction **Browse**. Puis faire **OK**



Le composant peut être programmé en faisant "**GO**" dans la barre d'outils

12. Programmer un MACH4

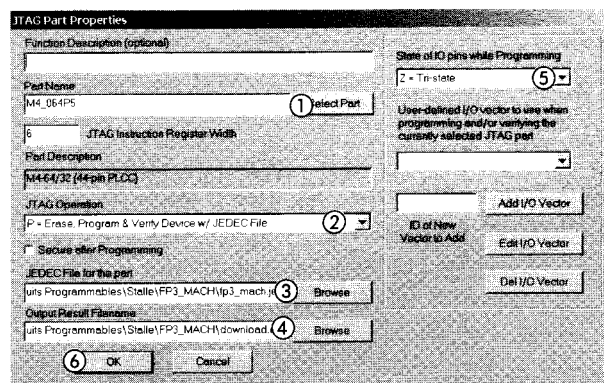
Dans le cas du composant MACH4, la programmation peut se faire directement, le composant restant sur la carte application, à l'aide du câble spécial de connexion à l'ordinateur. Le logiciel de chargement est isp VM System. Vous devriez trouver son raccourci dans le menu démarrer.



LatticePRO

Parmi les trois outils à votre disposition, dans l'environnement ispVM System seul Lattice PRO est à même de programmer les MACH4 par l'intermédiaire d'une connexion JTAG.

Dans cet environnement il faut définir un projet ce qui s'obtient par File/New. Dans la fenêtre du projet vous allez installer un composant par la commande Edit/Add Device.



Vous devez définir :

- 1 Un type de composant (MACH4 64/32 Plcc 44 p.e.)
- 2 L'opération à effectuer (P pour une programmation)
- 3 Le fichier JEDEC de configuration
- 4 Un nom de fichier pour l'écriture d'un rapport
- 5 L'état des entrées/sorties pendant la programmation (Z)
- 6 Puis terminer.

Votre projet doit alors se présenter sous une forme analogue à la suivante :



Après avoir :

- vérifié le choix du type de câble (Project/Advanced options),
- relié la carte au port parallèle de l'ordinateur,
- alimenté votre carte, ... il ne reste plus qu'à lancer la programmation par GO.

L'interface JTAG correspond à un format de communication série synchrone qui permet en outre d'enchaîner plusieurs composants sur une même carte électronique pour la phase de programmation.

13. Raccourcis clavier de la saisie de schéma

13.1.Édition

Ctrl + C : Interrompre la commande en cours

F9 : Annuler

Maj + F9 : Refaire

Ctrl + X : Couper

Ctrl + C : Copier

Ctrl + V : Coller

F5 : Effacer

F6 : Dupliquer

F7 : Déplacer

F8 : Glisser

Ctrl + R : Tourner (mode déplacer)

Ctrl + E : Miroir (mode déplacer)

13.2.Attributs

Ctrl + A : Attributs des symboles

Ctrl + P : Attributs des broches (Pin)

Ctrl + N : Attributs des équipotentiels (Net)

13.3.Visualisation

Ctrl + I : Zoom agrandir (In)

Ctrl + O : Zoom réduire (Out)

Ctrl + f : Zoom ajuster (Fit)

Ctrl + W : Centrer

13.4.Ajouter un élément

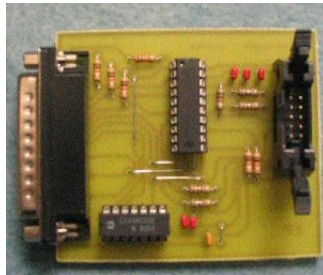
F3 : Fil

F4 : Nom d'équipotentielle

F2 : Un symbole

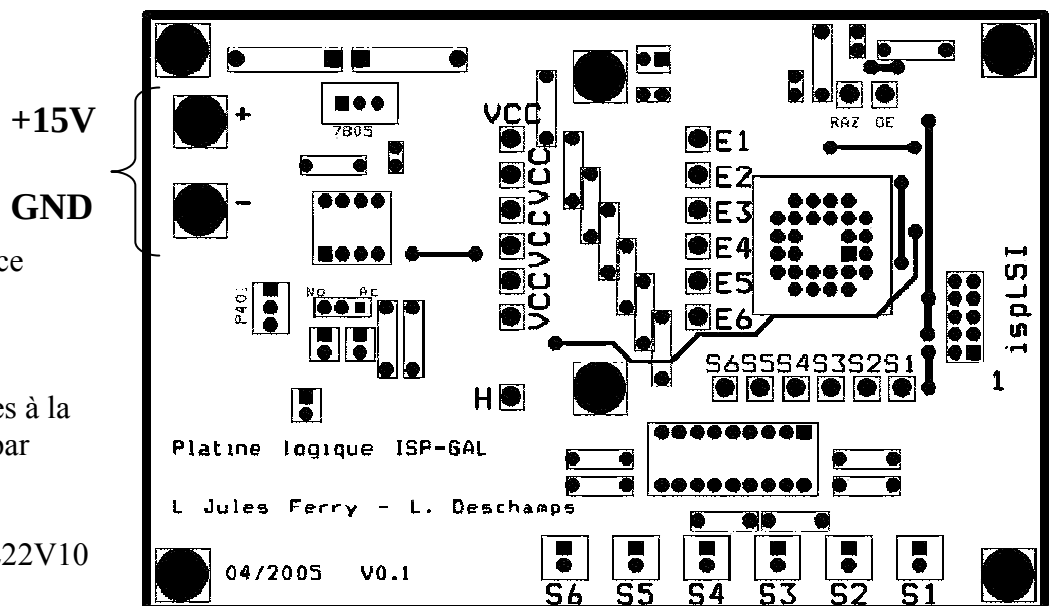
Annexes

Version avec JTAG + connecteur HE10-10 (schéma proposé par Lattice) :

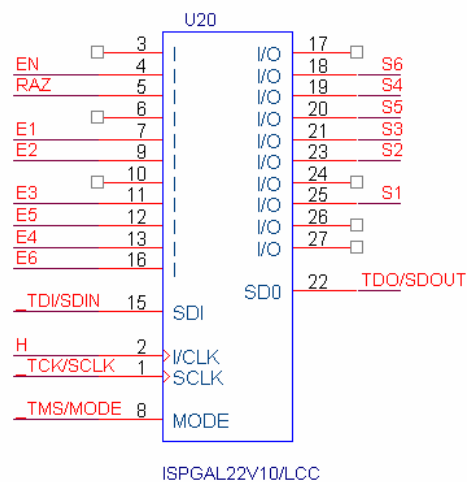


Maquette à base de ISP (GAL22V10) :

H : Horloge de fréquence réglable par P401.
 E1 à E6 entrées du GAL22V10.
 - En l'air E = 0V (reliées à la masse sur la maquette par défaut),
 - Relié à VCC, E= 5V
 S1 à S6 sorties du GAL22V10



Entrées / Sorties du GAL22V10 de la maquette :



D'après "Électronique numérique" Hachette Technique, ISBN 2.01.16.8282.7