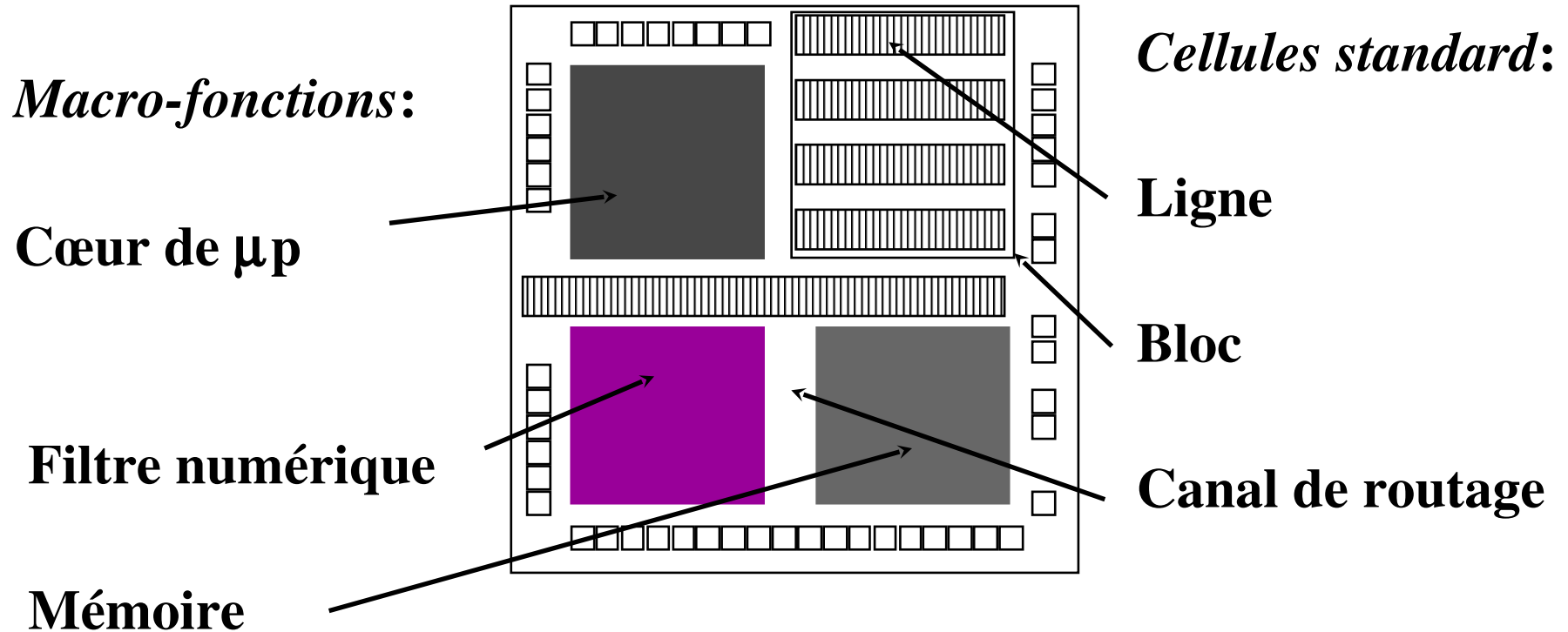


# Backend ASIC : un panorama

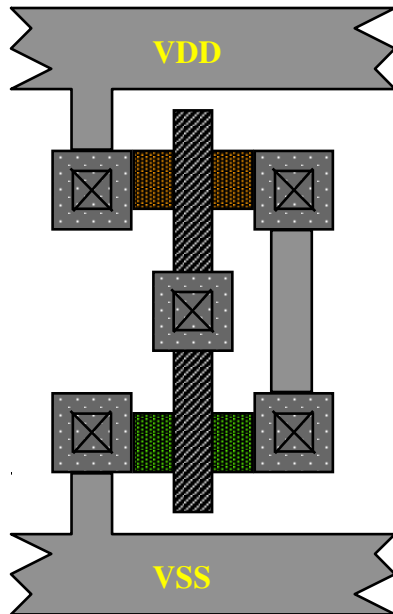
- *Les phases du placement/routage*
- **Bibliothèques de cellules**
- **RAM, ROM et DRAM Embarquées**
- **Génération de Chemins de Données**

# Organisation d'un circuit Numérique

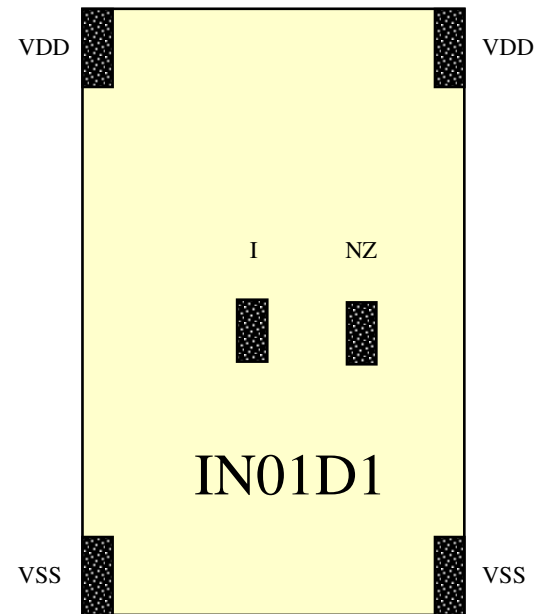


# Notion de cellule standard

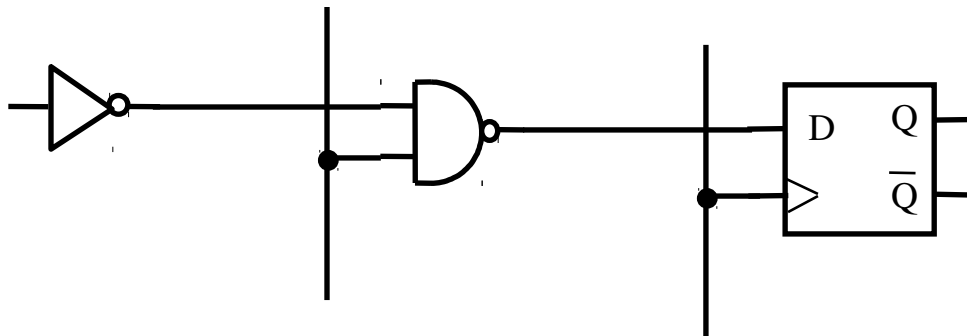
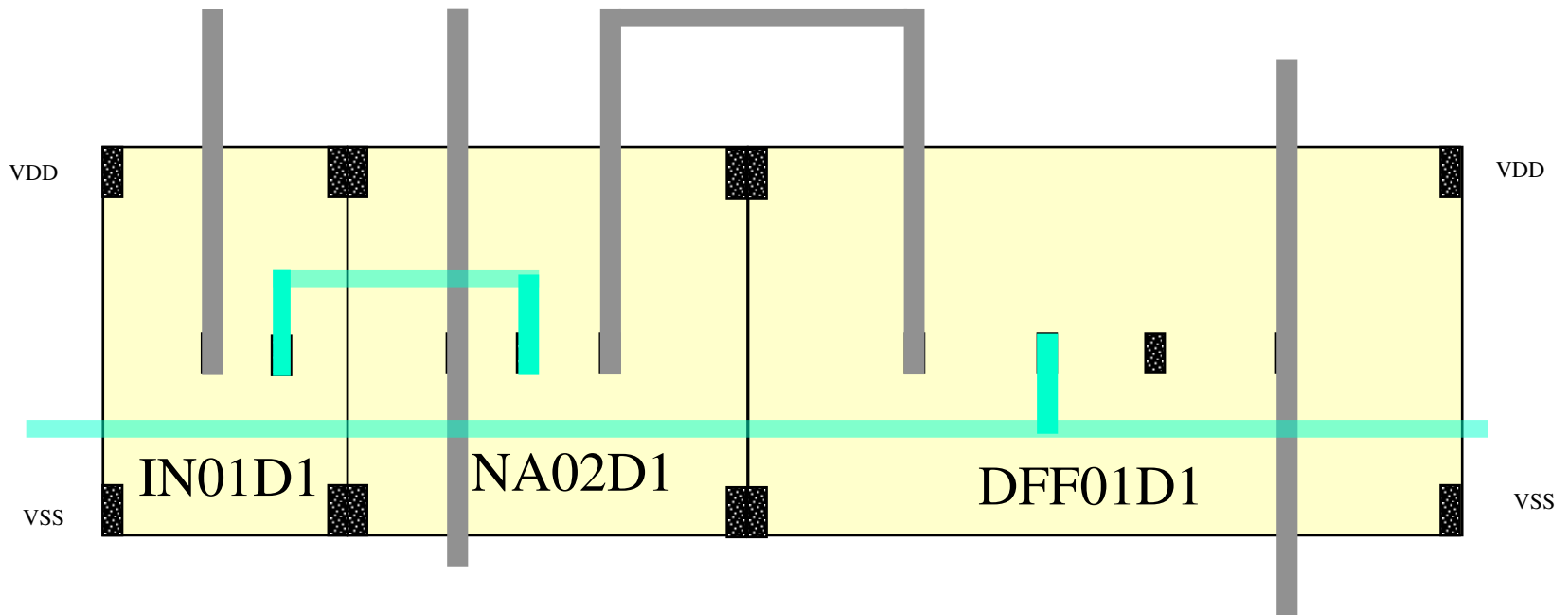
25 objets



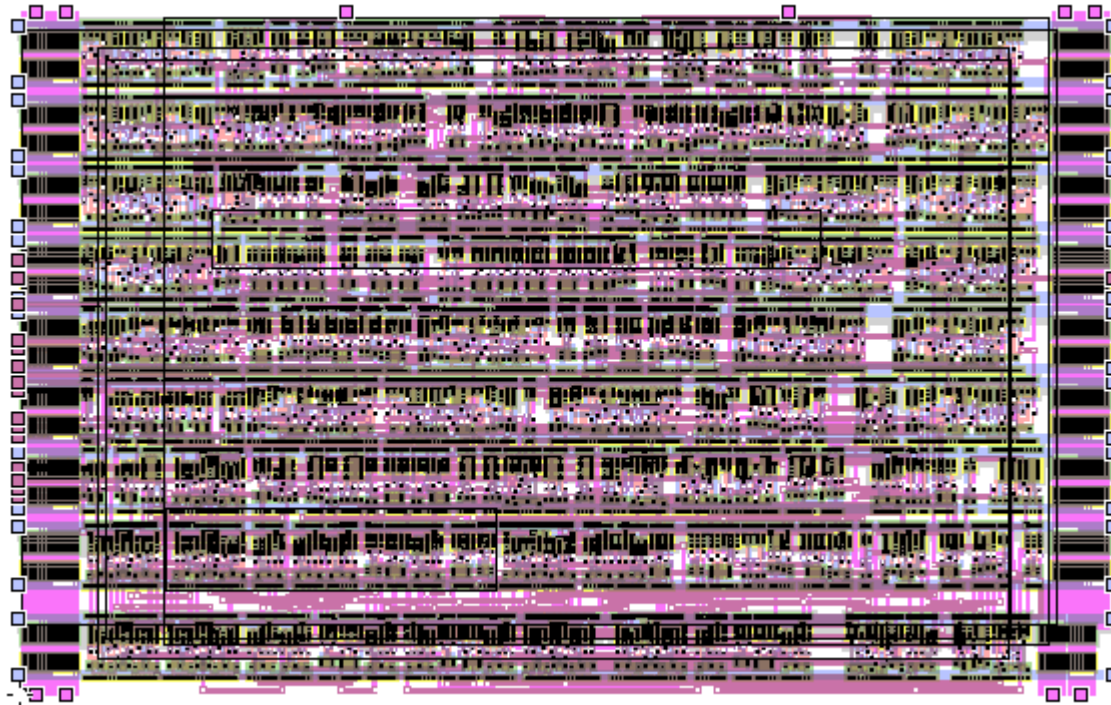
1 objet



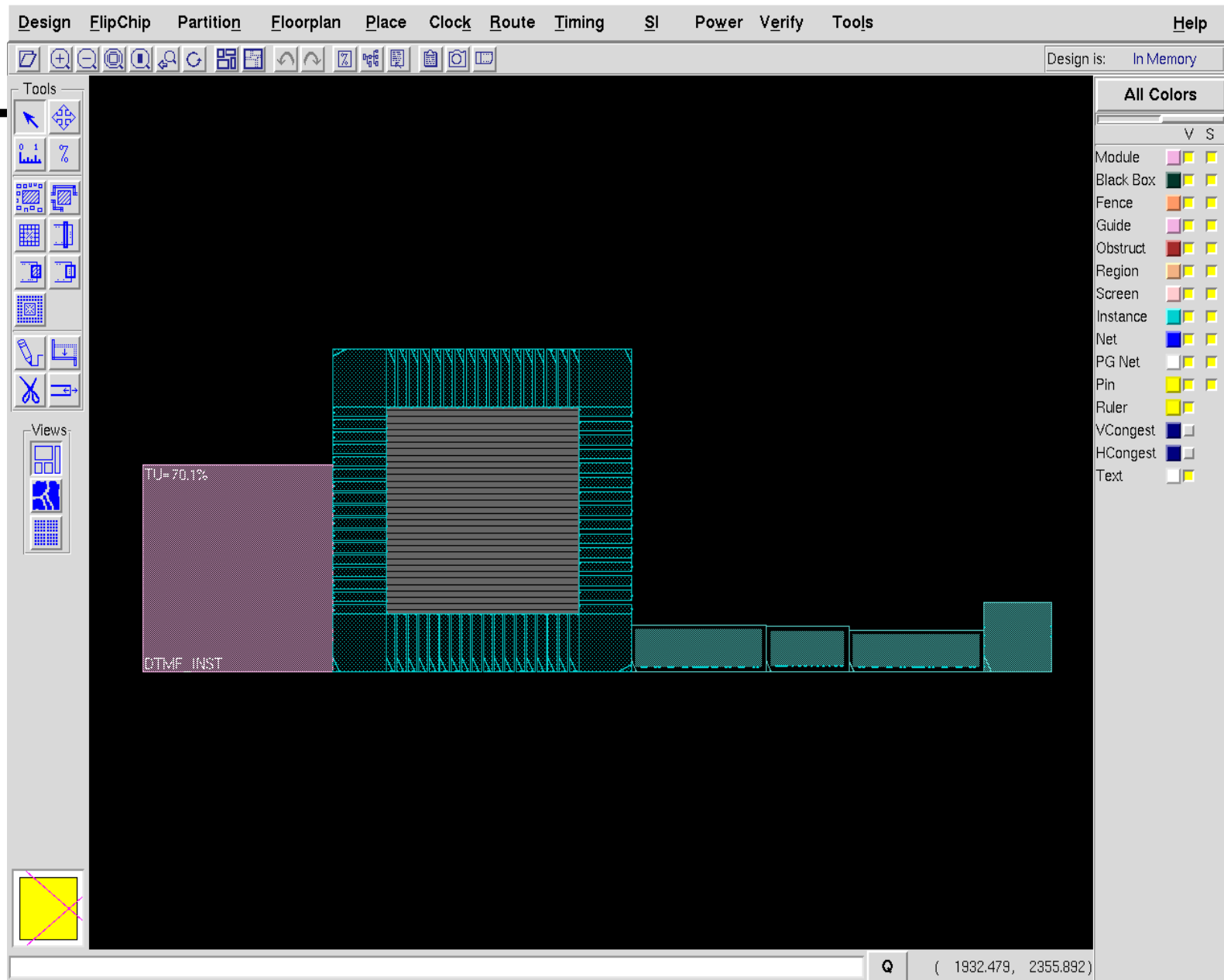
# Assemblage des cellules



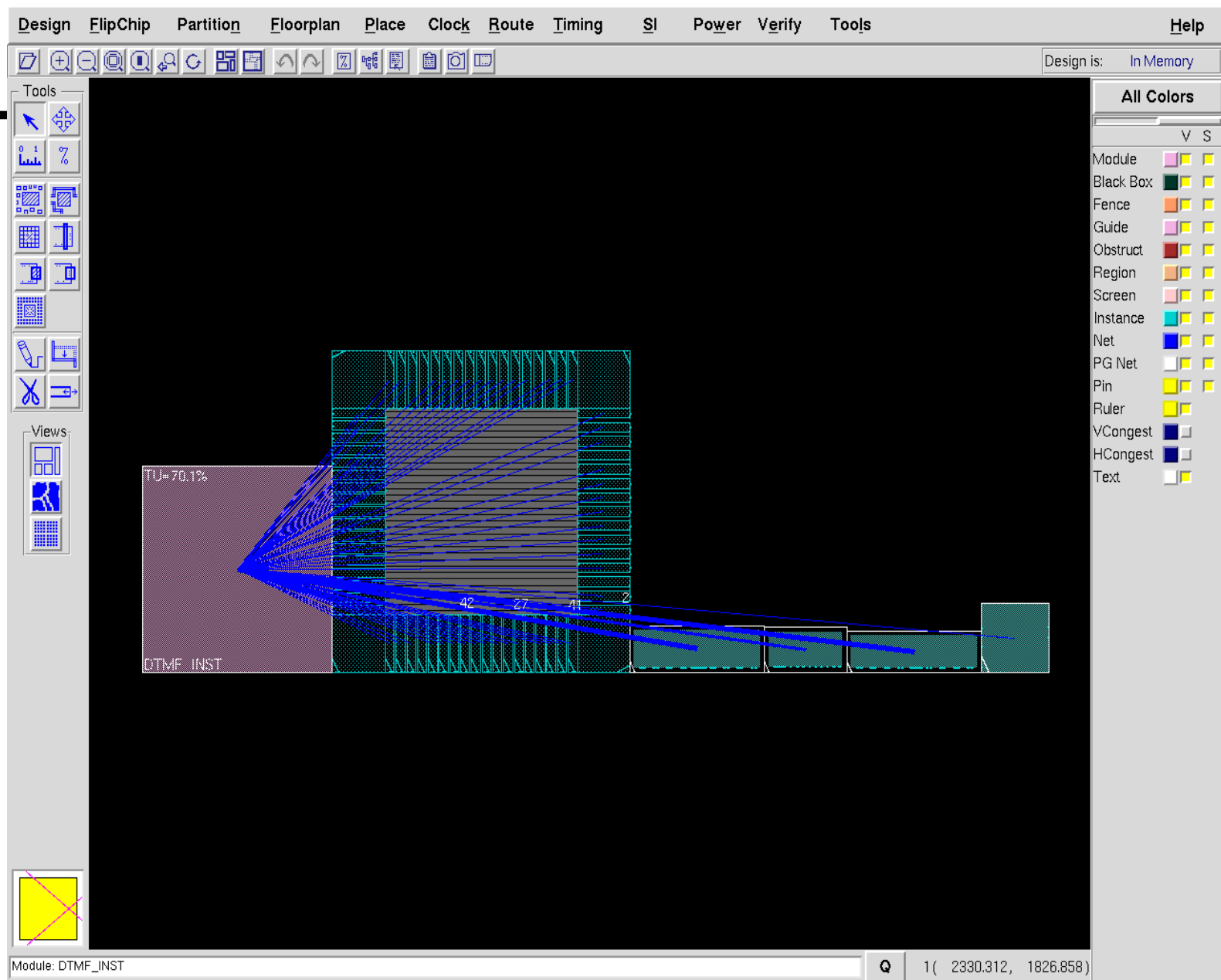
# Un bloc de cellules standard



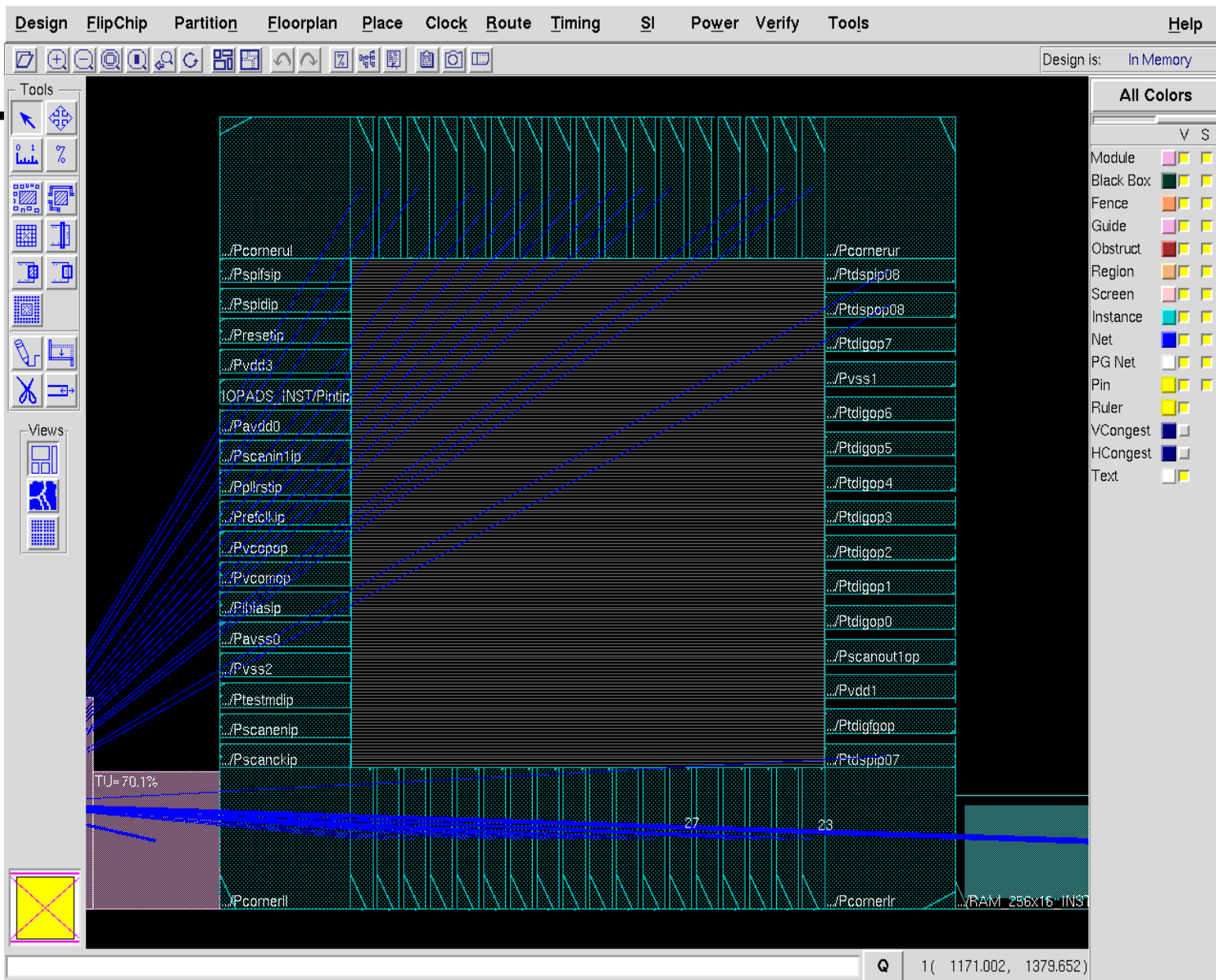
# La fenêtre de travail



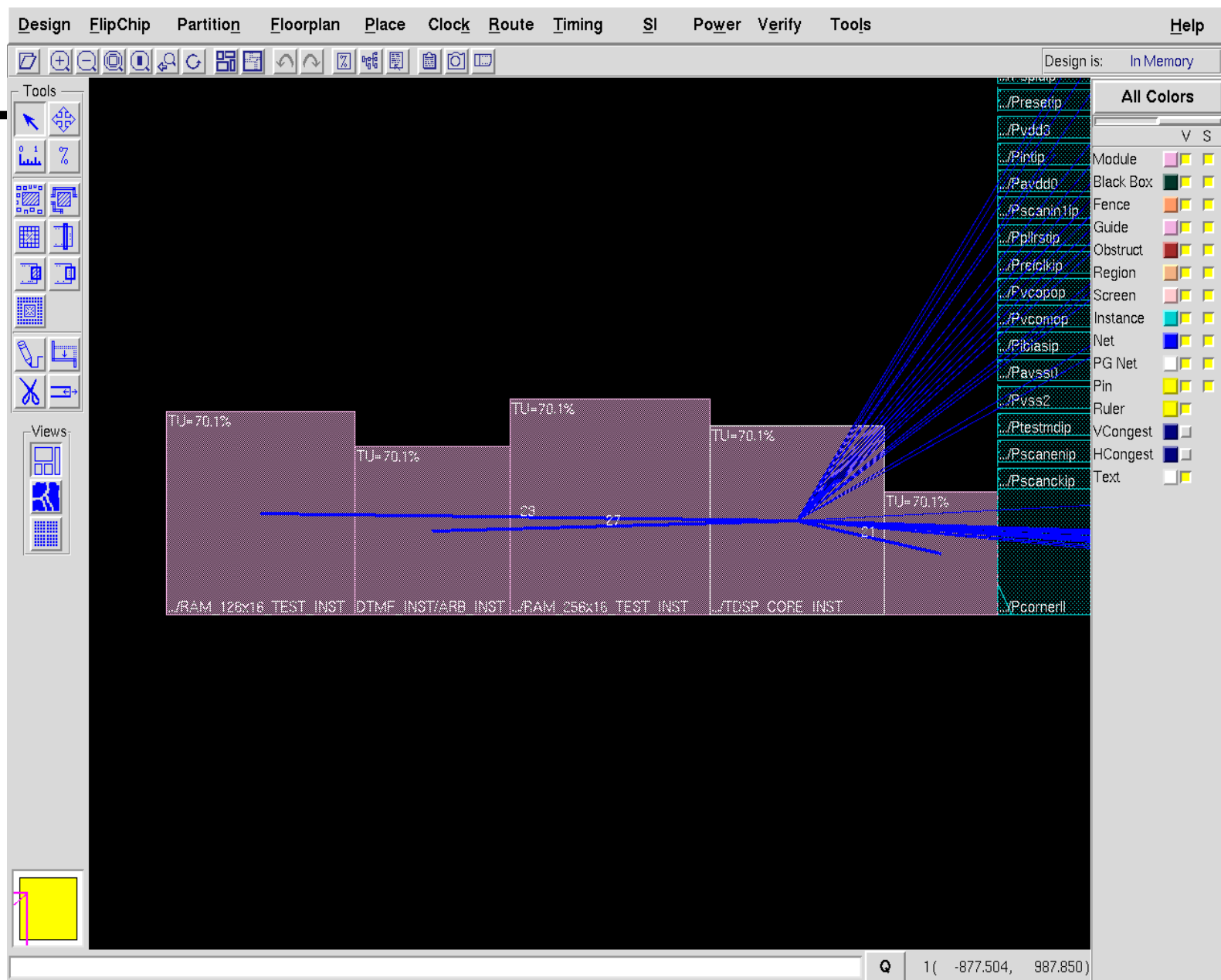
# Vision du « chevelu » global



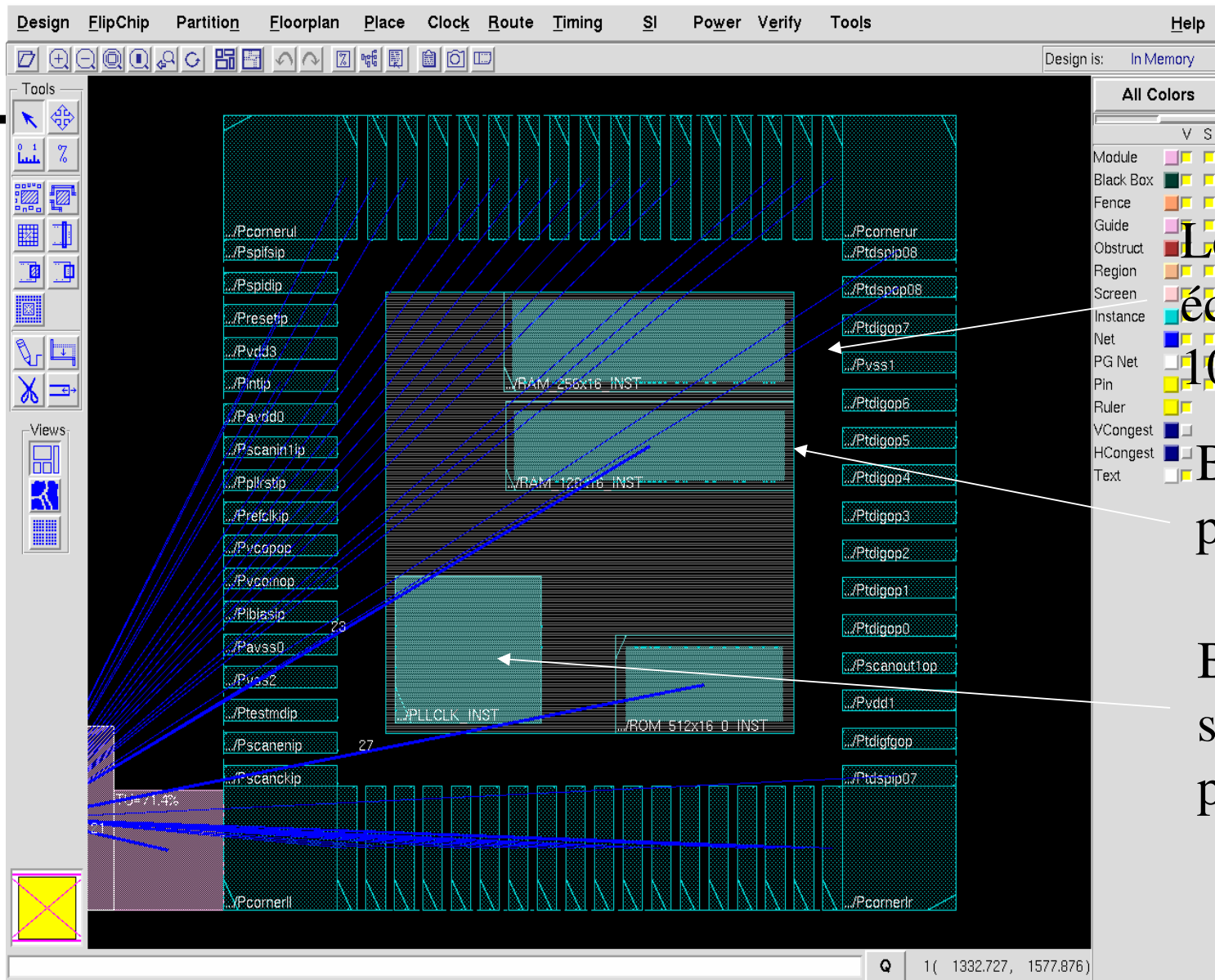
# Les pads (les triangles servent à déterminer les orientations des cellules)



# La structure hiérarchique peut être parcourue



# FLORRPLAN : Premier placement : les blocs durs

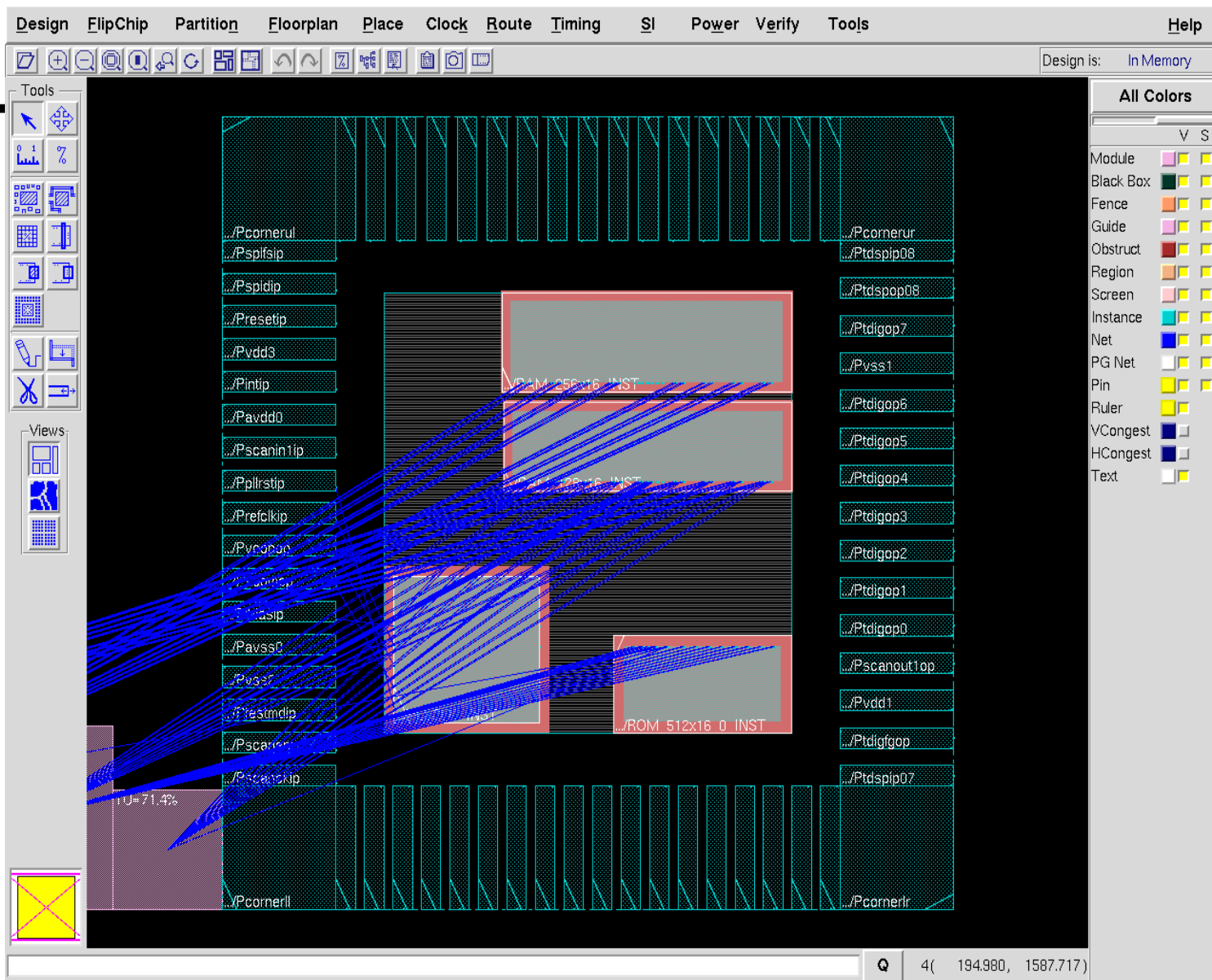


Les pads ont été écartés de 100Um

Bloc dur avec power ring »

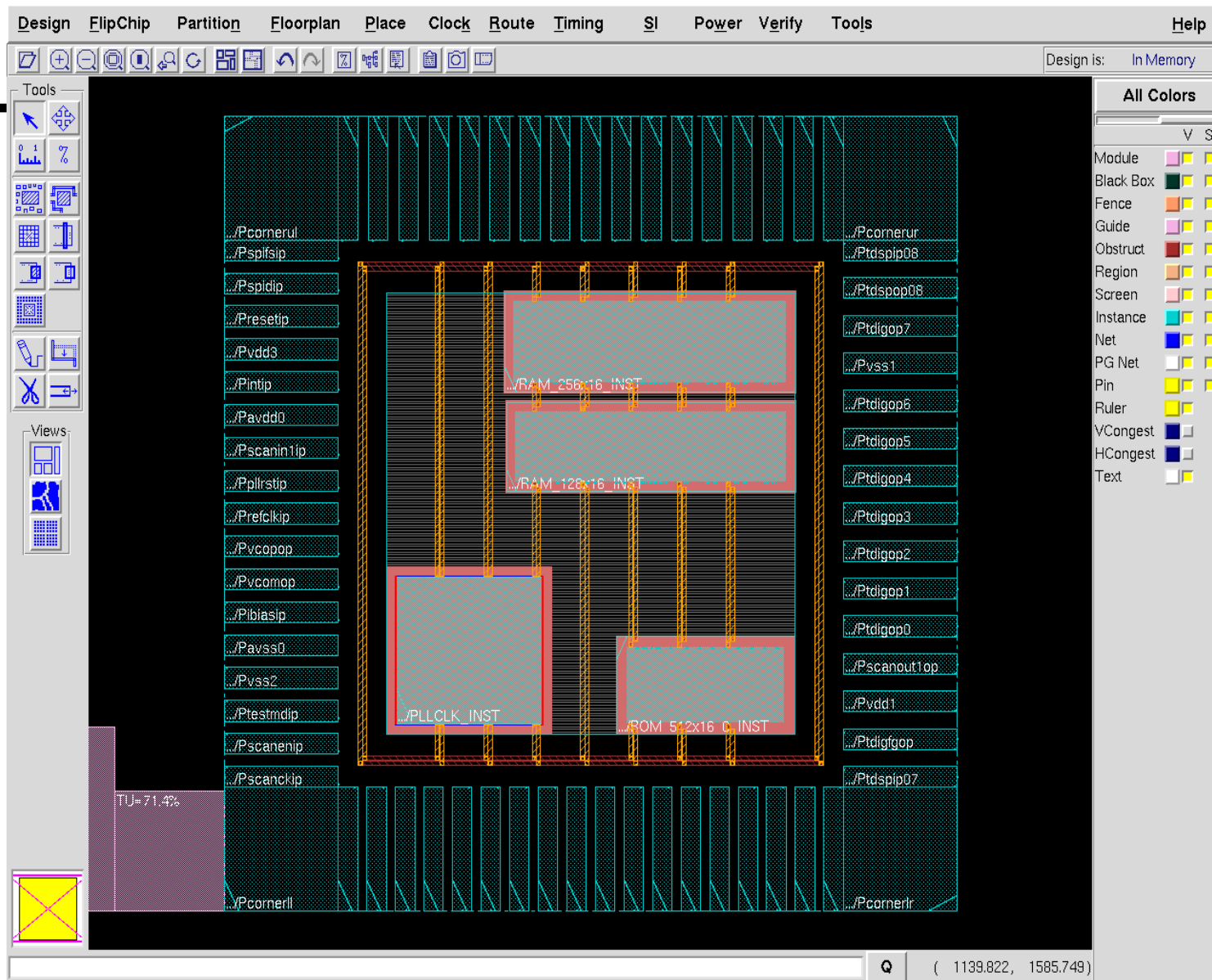
Bloc dur sans power ring »

# FLOORPLAN : Affichages des plages interdites

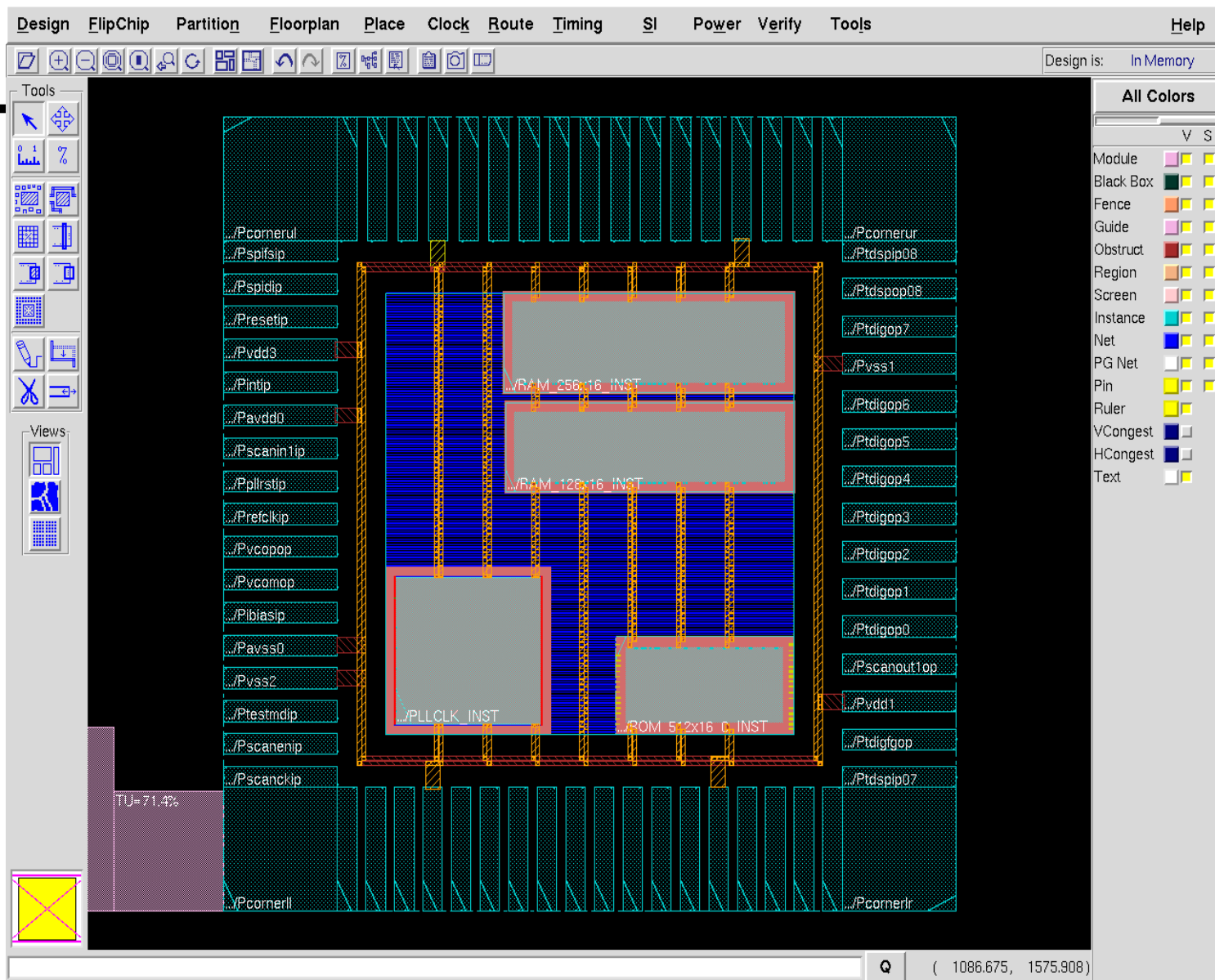




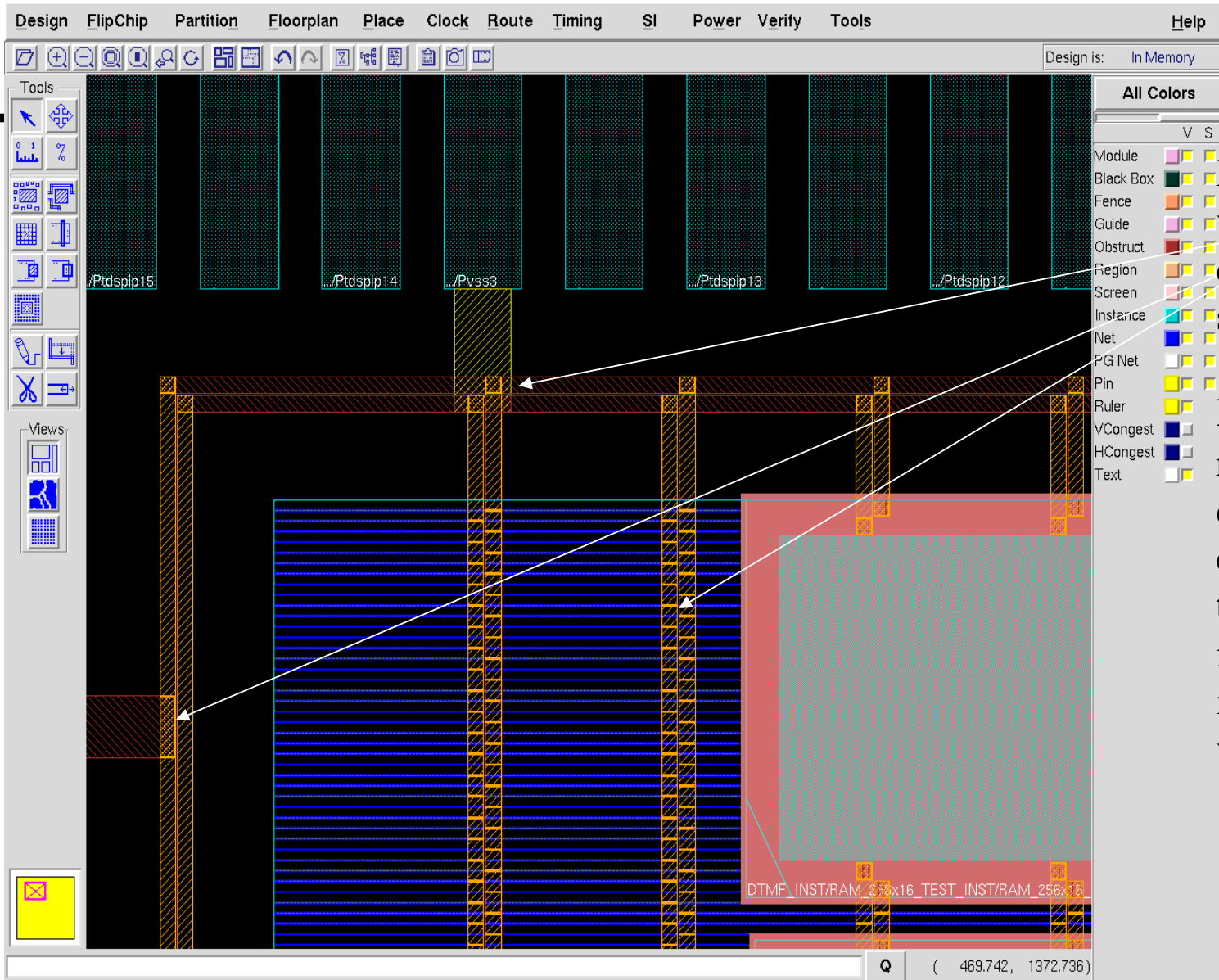
# FLOORPLAN : Routage des « STRIPES »



# FLOORPLAN : Le routage des alimentations est terminé



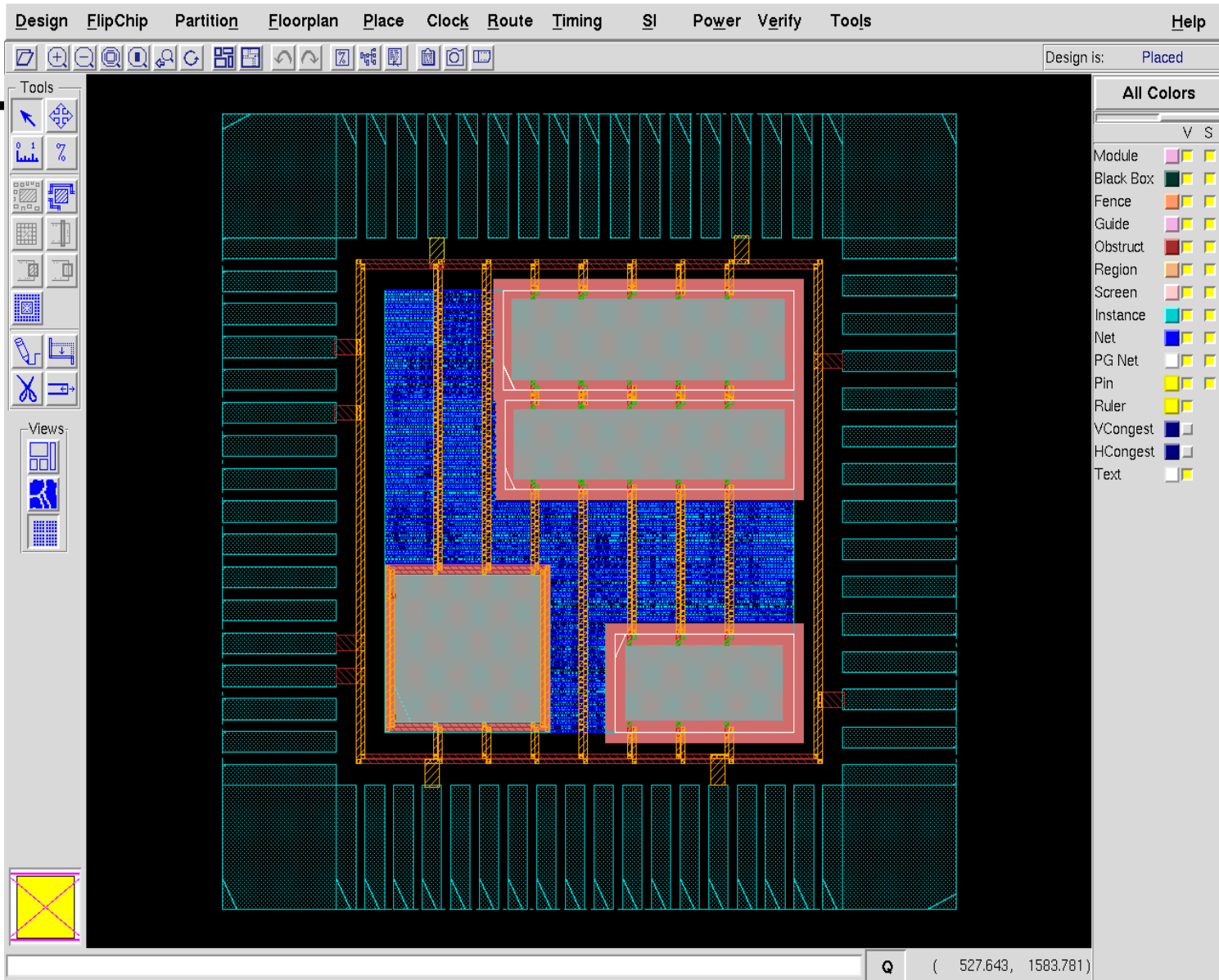
# FLOORPLAN : Détail du routage des alimentations



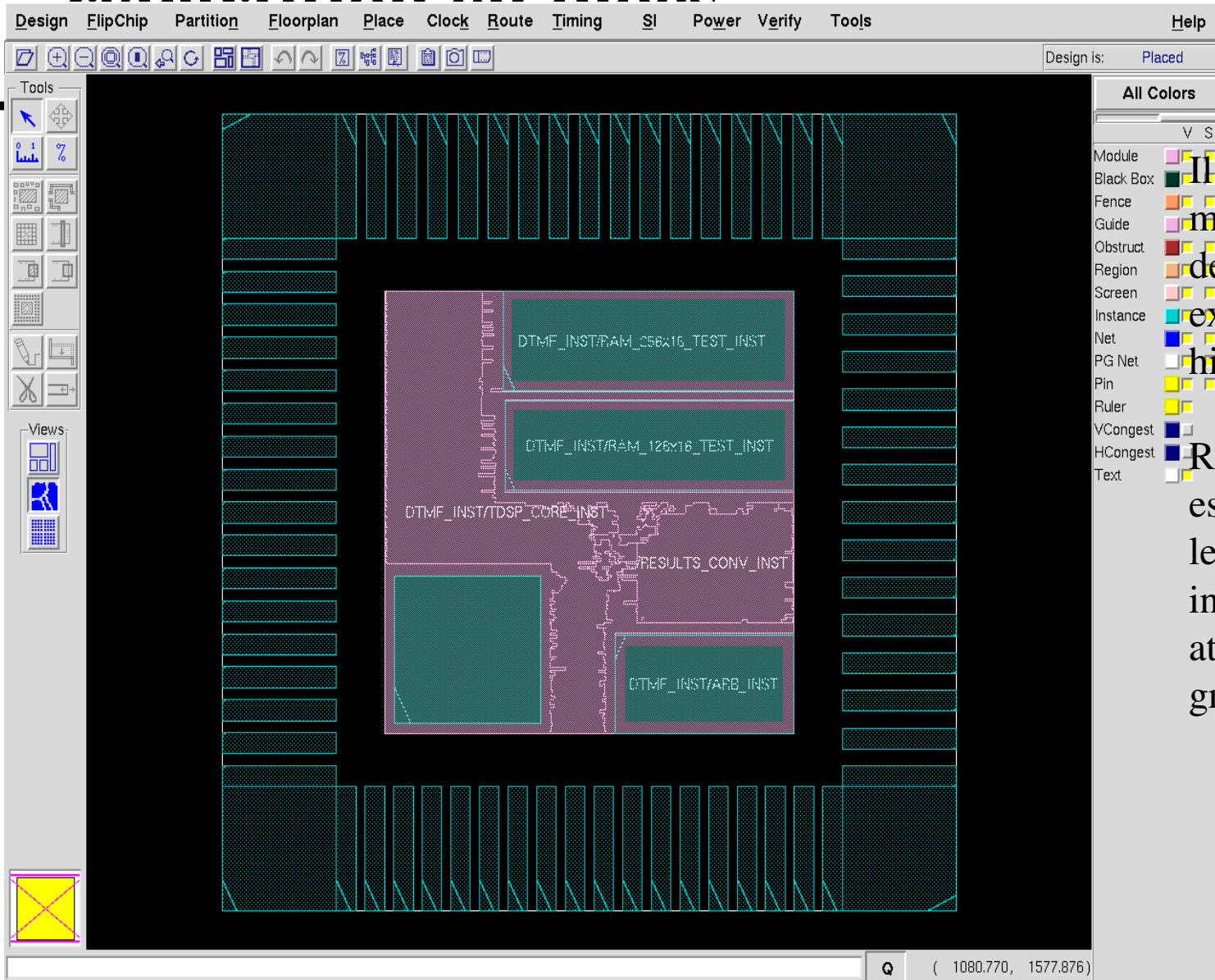
Les tableaux de vias de facteur de forme multiples sont générés.

**Remarque:** une règle générale de technologie est de préférer un tableau de vias me dimensions minimales à un via de grande taille

# PLACEMENT : terminé



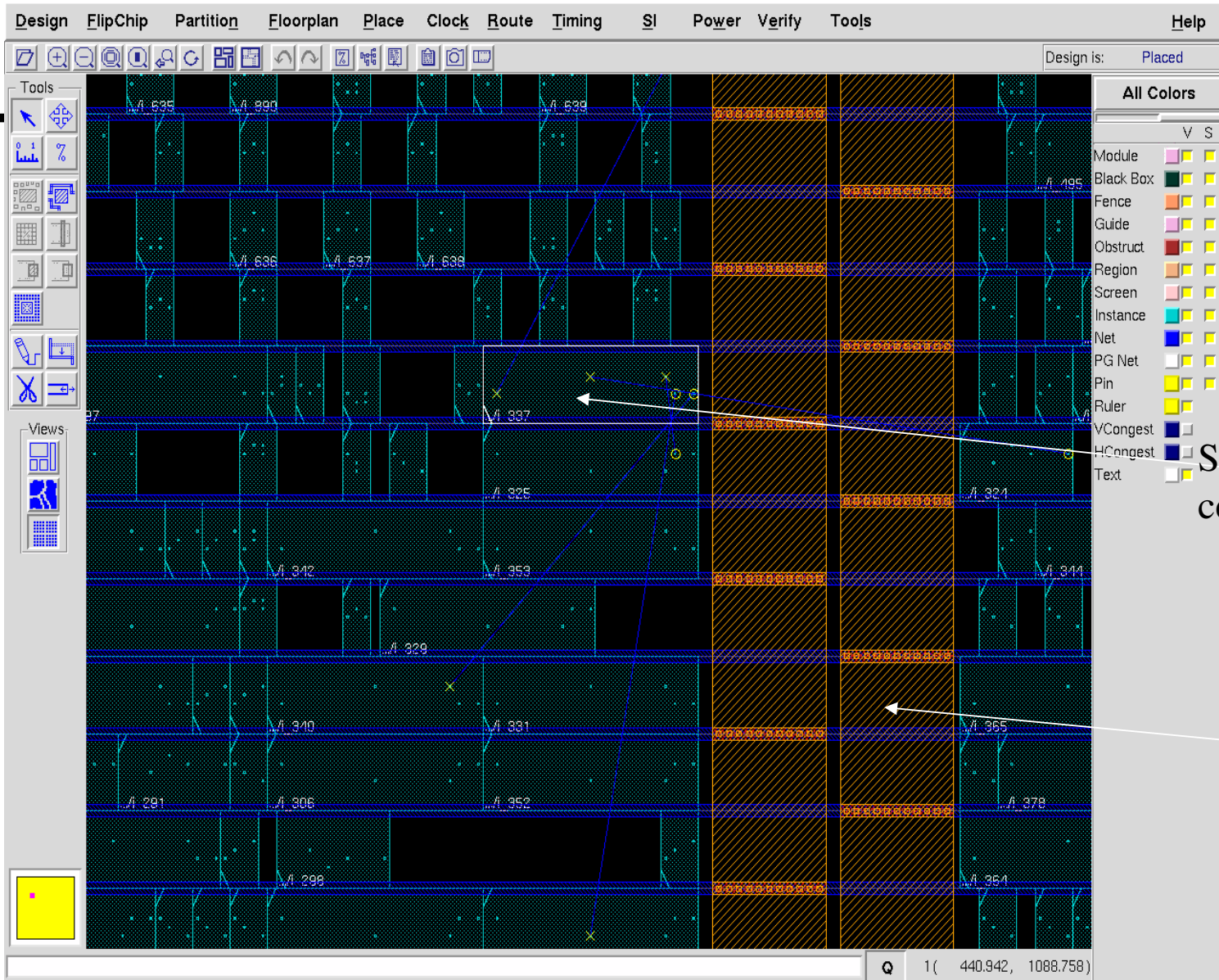
# PLACEMENT : analyse de la répartition en blocs



Il est possible d'examiner la répartition des cellules en explorant la hiérarchie.

Réciproquement, il est possible de guider le placement en indiquant des régions attribuées à des groupes de cellules

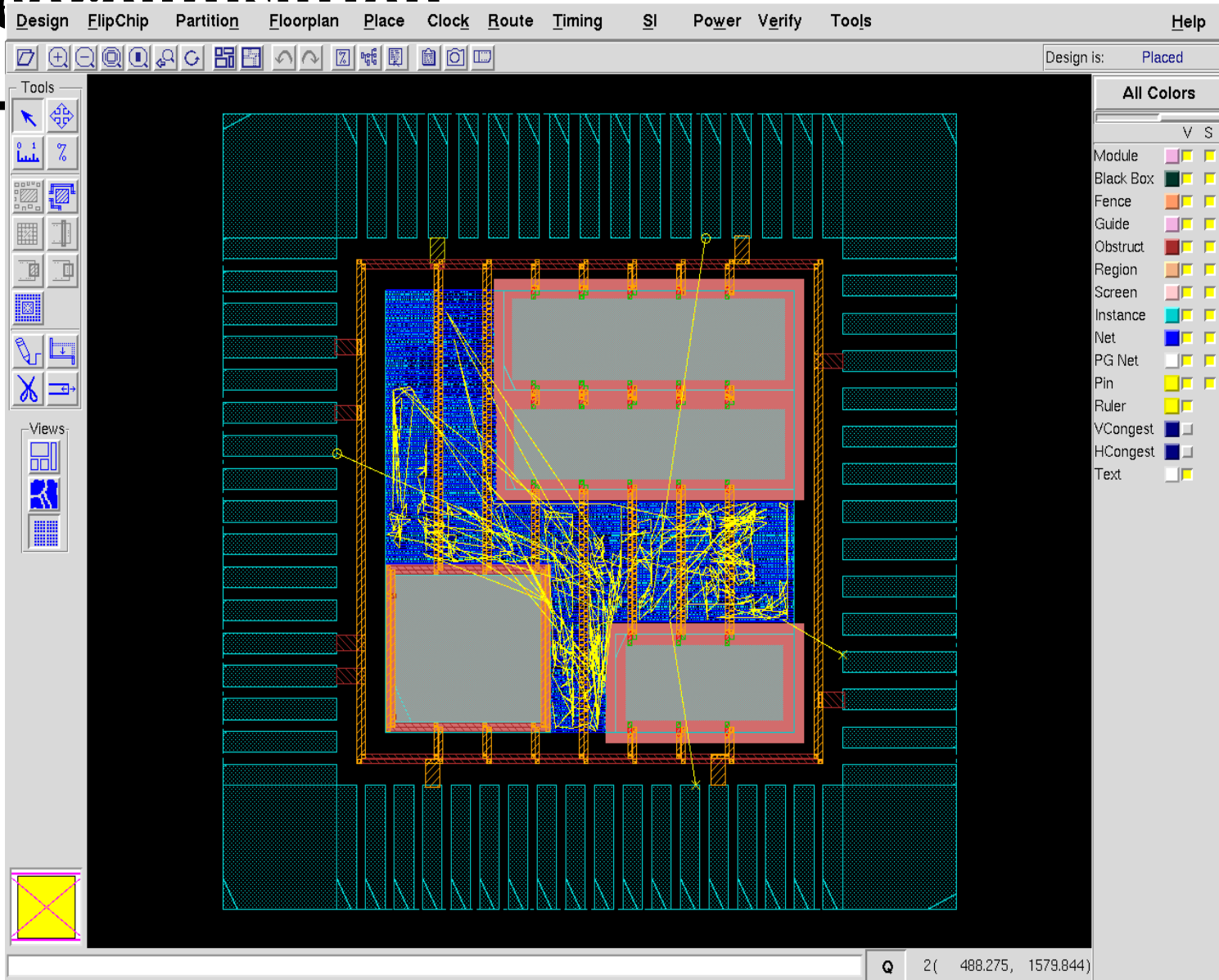
# PLACEMENT : DETAIL



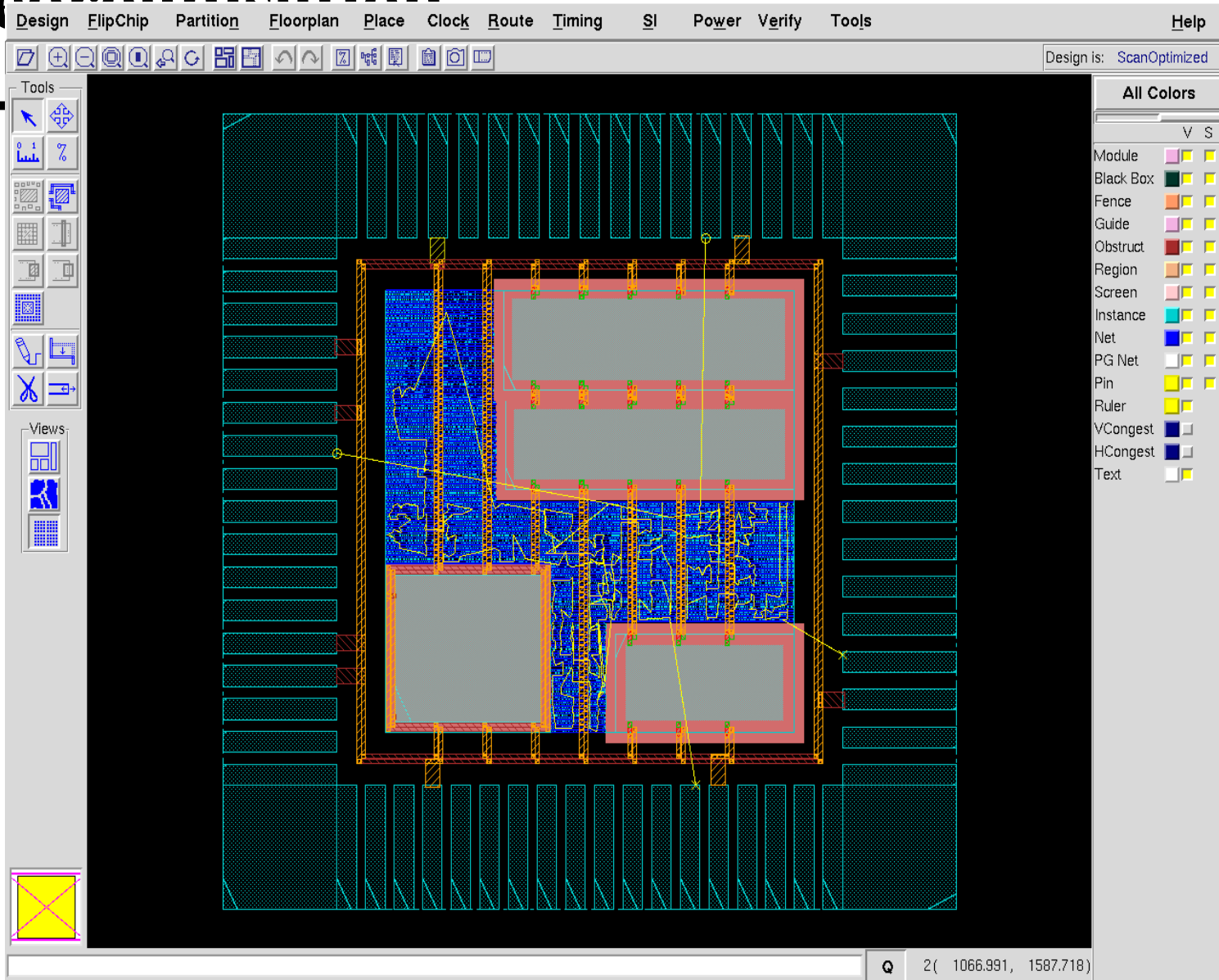
Sélection d'une  
cellule pour examen

Il n'y a pas de  
cellule sous les  
alimentations

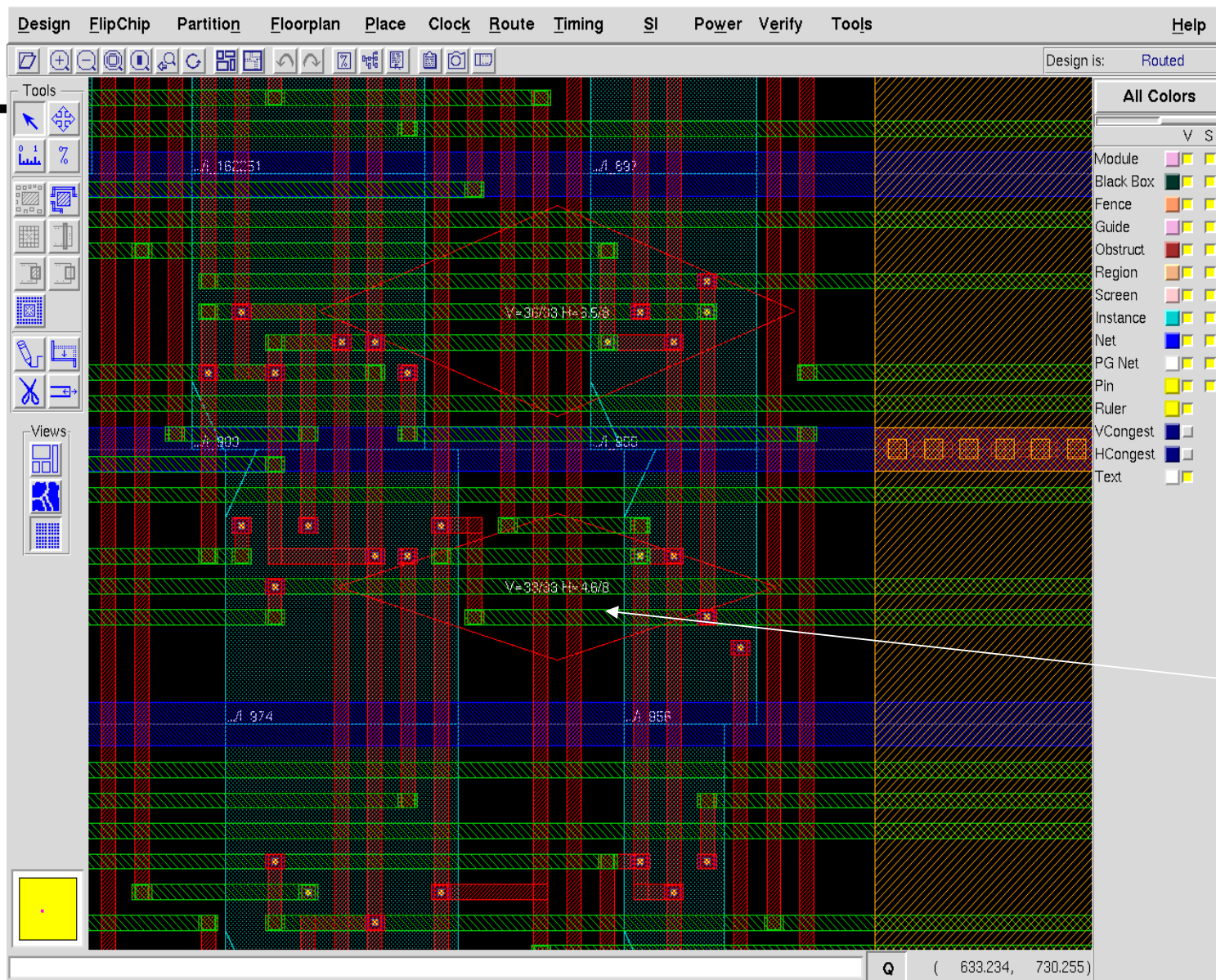
# SCAN : La chaîne de scan avant réorganisation



# SCAN : La chaîne de scan après réorganisation

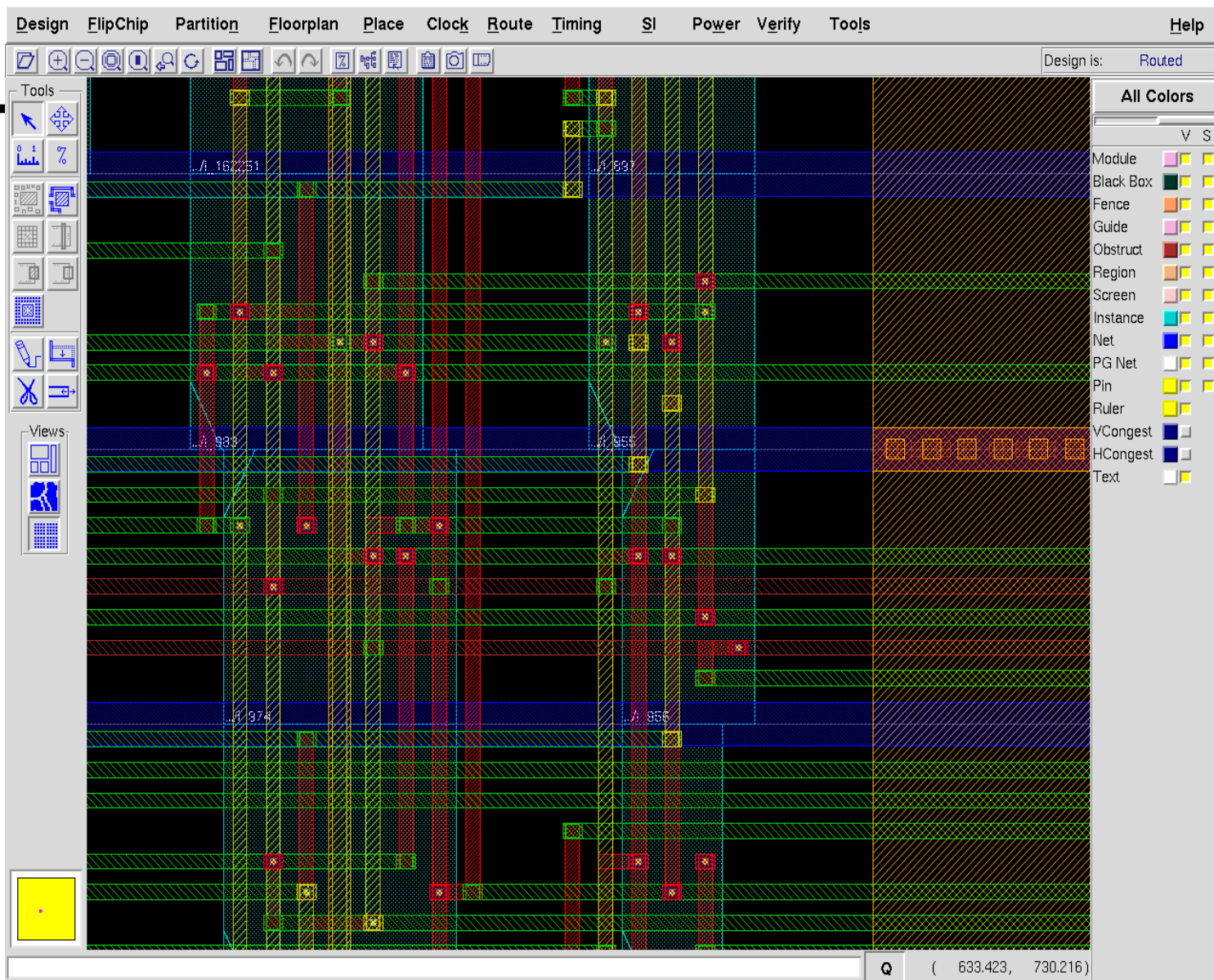


# ROUTAGE : congestions dues aux contraintes trop fortes

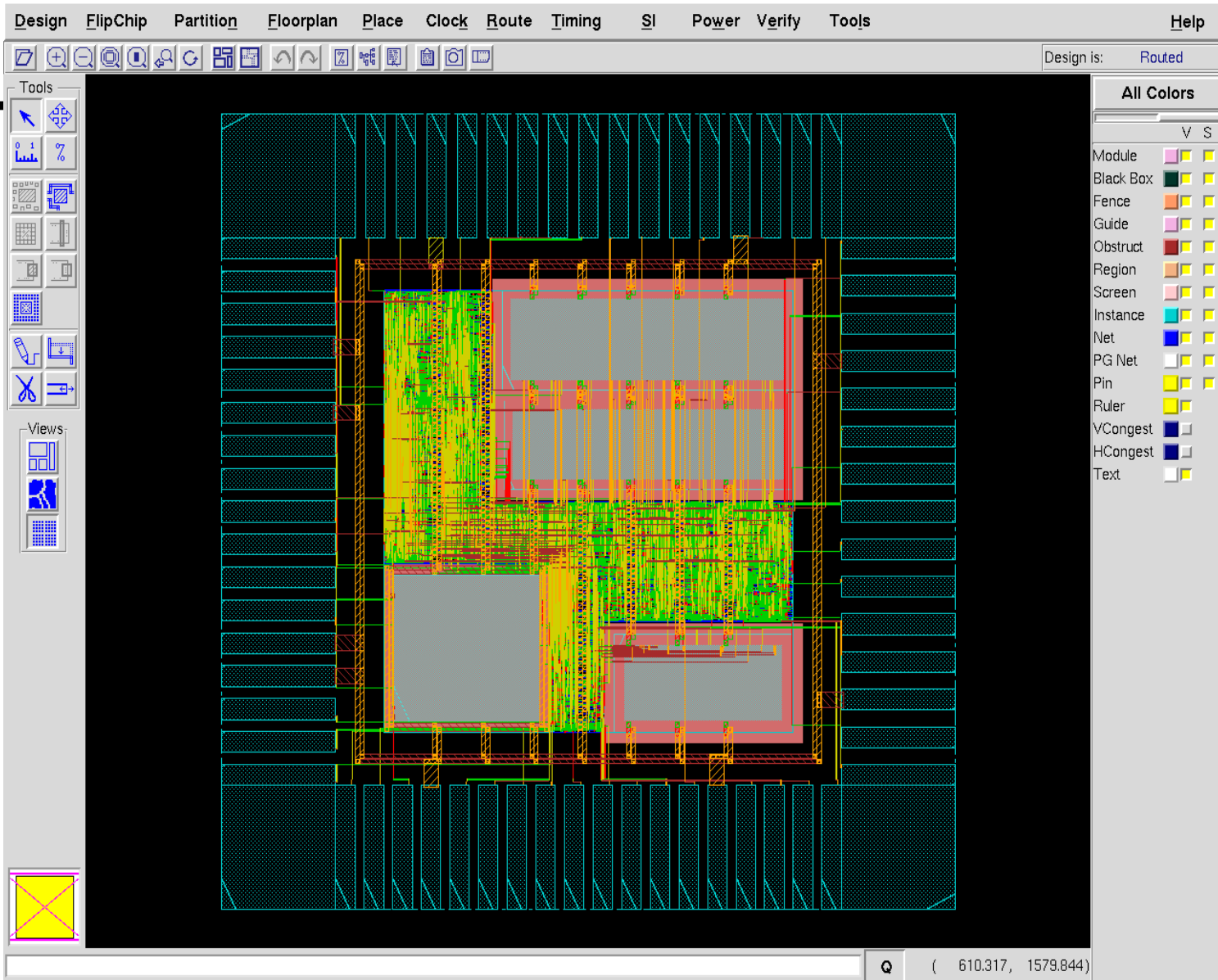


Congestions pointées par le losange.

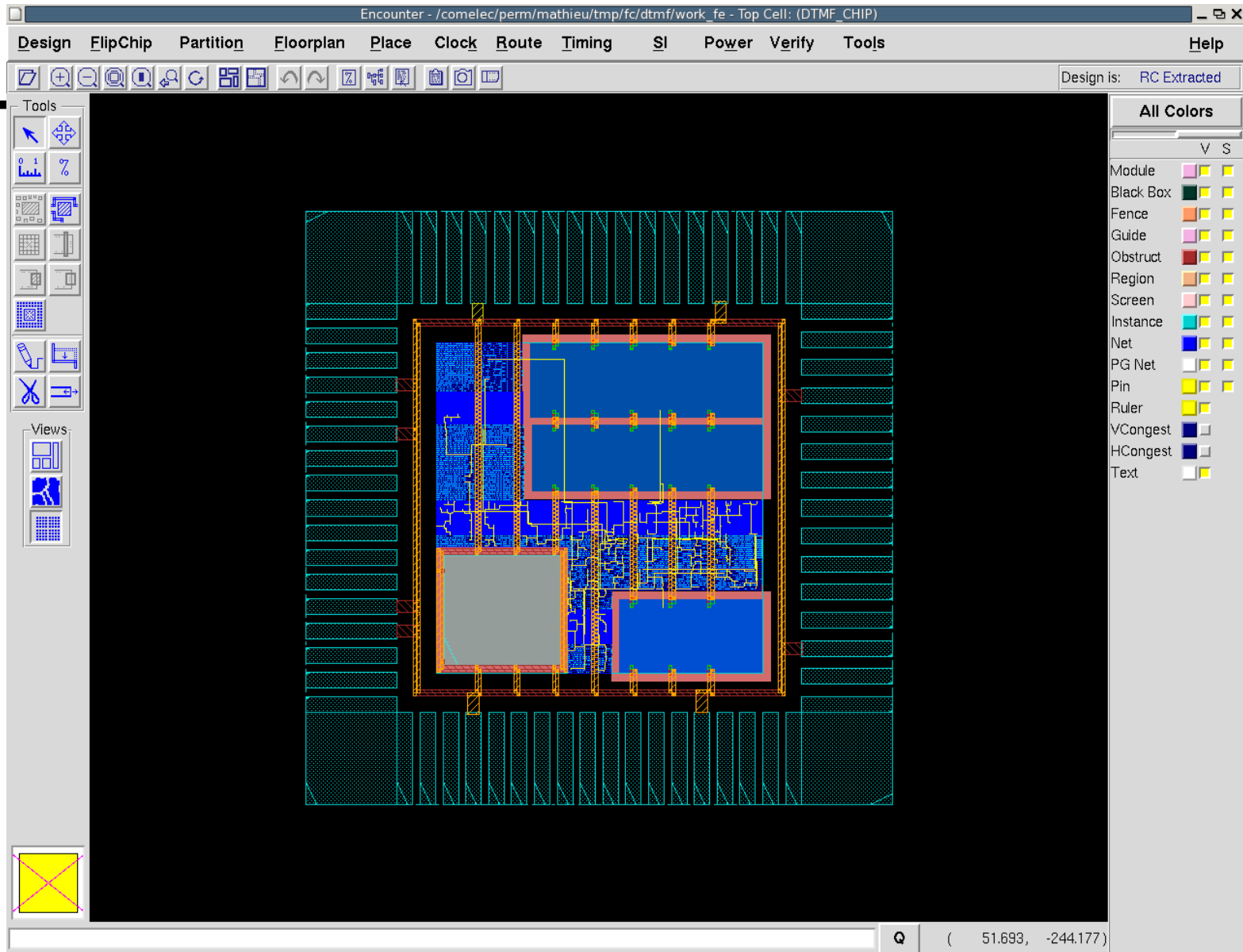
# ROUTAGE : avec 6 niveaux de métal c'est mieux



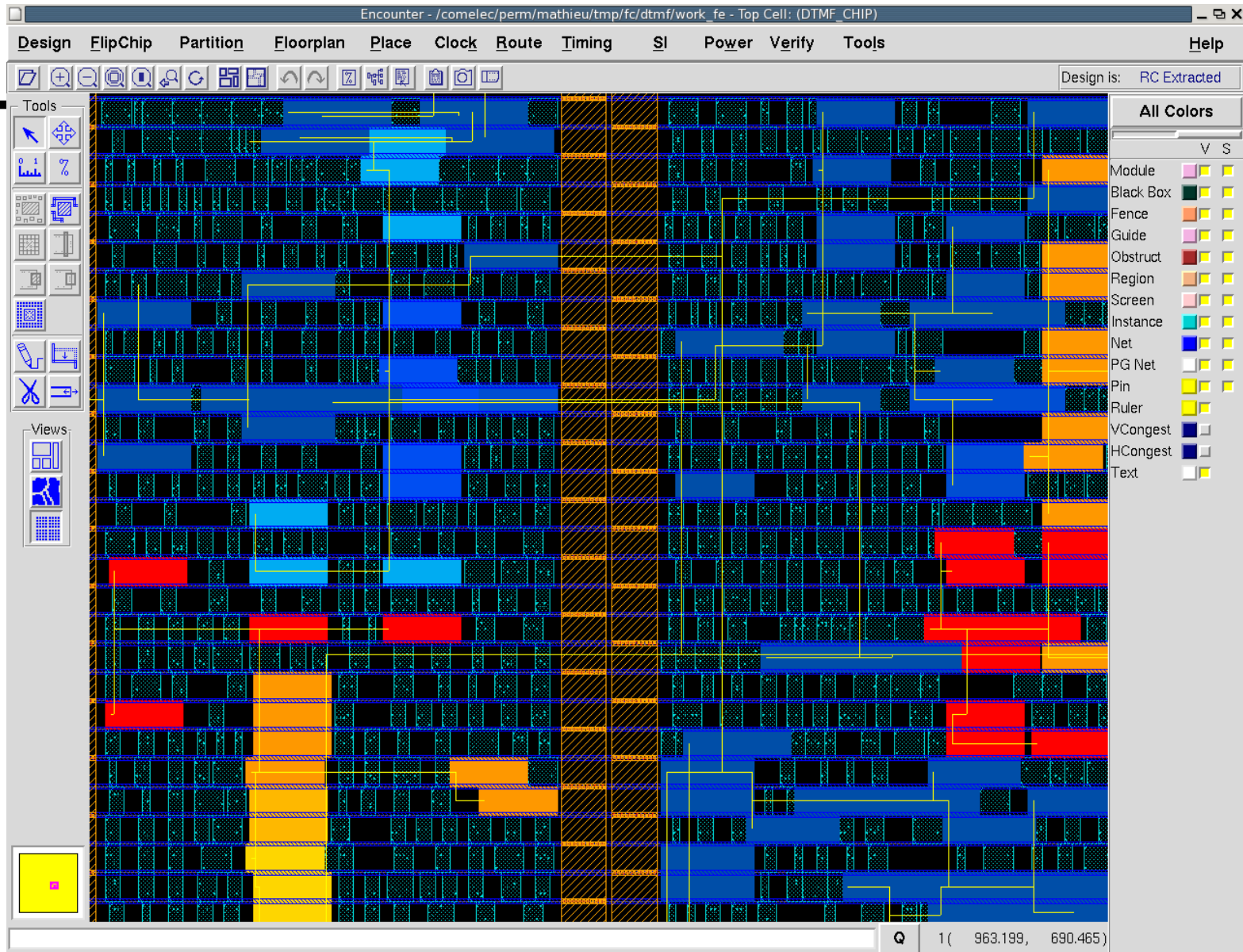
# ROUTAGE : le circuit routé



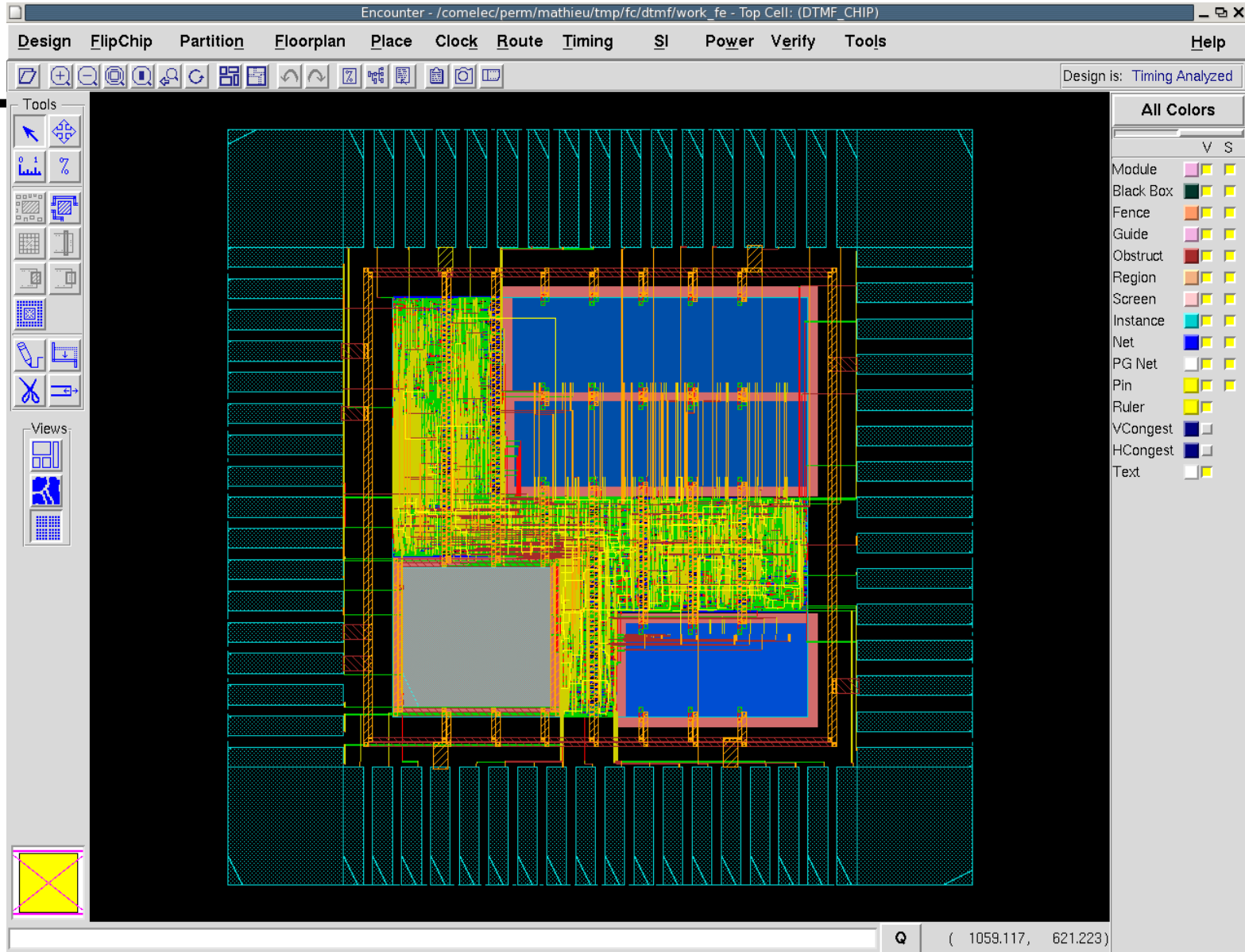
# SYNTHESE DE L'ARBRE D'HORLOGE : Visualisation de l'arbre



# SYNTHESE DE L'ARBRE D'HORLOGE : les délais...



# Routage des éléments ajoutés (IPO + horloge)



- **Les phases du placement/routage**
- *Bibliothèques de cellules*
- **RAM, ROM et DRAM Embarquées**
- **Génération de Chemins de Données**

# Vision du concepteur

---

- ❑ A priori, le concepteur ne connaît pas le contenu de la bibliothèque précaractérisée.
- La transcription d'une description (Shéma, VHDL, Verilog,...) en assemblage de cellules est prise en charge par l'outil de synthèse.
- Mais....



# Vision du concepteur

---

## □ En phase de vérification :

☒ *Les portes « physiques » utilisées sont détaillées dans les chemins critiques*

□ *Le concepteur chevronné peut évaluer le bon choix de portes réalisé par le synthétiseur*

## □ En phase conception :

□ *Le concepteur peut restreindre et guider le choix de portes par le synthétiseur*

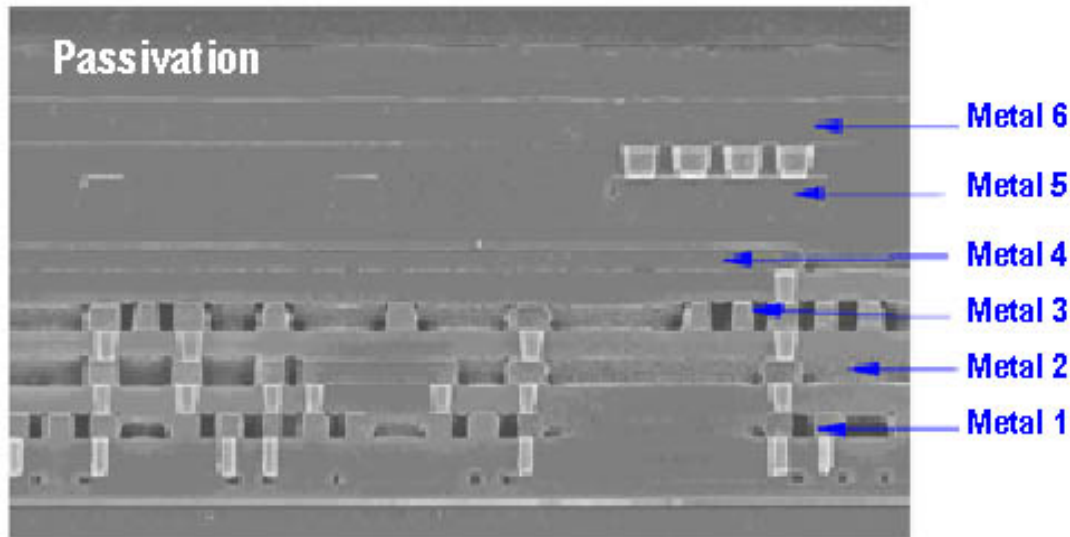
# Caractéristiques des bibliothèques

---

- ❑ **Complexité (nombre de portes disponibles)**
- ❑ **Complexité et précision du modèle temporel des portes**
- ❑ **Qualité du modèle de consommation**
- ❑ **Adaptation aux différents outils de CAO du marché (Cadence, Mentor, Avanti....)**

# Interpréter un document Marketing

## ST introduces new 0.18 $\mu\text{m}$ standard cell library



- *Une bibliothèque est associée à une technologie...*
- *Pas ou peu de bibliothèques virtuelles...*

The CB65000 standard cell series uses a high performance, low-voltage, 0.18 $\mu\text{m}$  drawn, six metal levels, high density and high speed HCMOS8D process.

# Interpréter un document Marketing

- Densité d'intégration
  - Après placement et routage final

*With an average routed gate density of 85,000 gates/mm<sup>2</sup>, the CB65000 family allows the integration of up to 30 million equivalent gates and is ideal for high-complexity or high-performance devices for computer, telecommunication and consumer products.*
- Limitation de la taille des puces

# Interpréter un document Marketing

---

*With a gate delay of 35 ps with High Performance transistor and 60 ps with Low Leakage transistor (for a 2-input NAND gate at fan-out 1), the library meets the most demanding speed requirements in telecommunication and computer application designs today*

- ❑ **On ne peut pas optimiser à la fois vitesse et consommation.**
- ❑ **En réalité il y a plusieurs bibliothèques basées sur des compromis différents**

# Interpréter un document Marketing

## ✓ Consommation moyenne d'une porte

*Optimized for 1.8V operation, the library features a power consumption of less than 35nW/Gate/MHz (High Performance; fan-out=1) and 25 nW/Gate/MHz (Low Leakage; fan-out=1) at 1.8V.*

✗ Bien que l'on dispose d'une plage de tensions d'alimentation utilisables, la bibliothèque est optimisée pour une tension nominale donnée.

# Interpréter un document Marketing

---

- ✓ Il faut communiquer avec l'extérieur du circuit

*The I/O buffers can be fully configured for both 1.8V and 3.3V interface options, with several high speed buffer types available. These include: low voltage differential (LVDS) I/Os, PCI, AGP, USB, LVTTL, LVCMOS and SSTL.*

- La bibliothèque d'Entrée/Sortie est dissociée. de la bibliothèque de cellules Standards.
- Le concepteur intervient directement dans le choix des cellules d'Entrée/Sortie

# Bibliothèques d'entrées/sorties

---

- ▣ **Les cellules E/S sont caractérisées en vitesse et consommation au même titre que les cellules logiques**
- ▣ **Les modèles de vitesse et de consommation sont intégrés par les outils de vérification (simulateurs, analyse temporelle...)**

# Complexité des bibliothèques

---

- ▣ Exemple de bibliothèque :
  - ▣ *293 Cellules combinatoires (nand, nor..)*
  - ▣ *88 Cellules séquentielles (bascules D..)*
  - ▣ *Une cellule pour l'arbre d'horloge*
- ▣ La richesse permet aux synthétiseurs d'optimiser aux mieux les critères de complexité, de vitesse ou de consommation

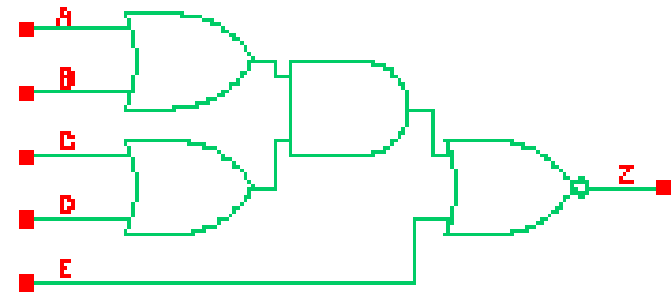
# Complexité : gestion des charges

---

- ▣ 4 Instances du AND à 4 entrées
  - ▣ *AN4, AN4P, AN4X3, AN4X4*
- ▣ La seule différence provient de l'inverseur de sortie de taille croissante
- ▣ Le synthétiseur choisira la version la plus adaptée au contexte d'utilisation...

# Complexité : portes CMOS complexes

- La logique CMOS permet de réaliser de façon efficace n'importe quelle fonction de type Somme(Produits..)
- Le synthétiseur peut exploiter de manière efficace l'existence de telles fonctions.
- Exemple : AO13X05



# Complexité : variantes d'une bascule

---

- ▣ Le synthétiseur ne choisi que la bascule strictement nécessaire...
- ▣ Bascule D
  - ▣ *Avec ou sans « reset »*
  - ▣ *Avec ou sans « set »*
  - ▣ *Avec ou sans sortie « Q »*
  - ▣ *Avec ou sans sortie « QB »*
- ▣ 12 cas possibles....

- **Les phases du placement/routage**
- **Bibliothèques de cellules**
- ***RAM, ROM et DRAM Embarquées***
- **Génération de Chemins de Données**

# Bibliothèque de RAM

---

- ▢ Fournir un bloc RAM ou ROM :
  - ▢ *nombre de mots*
  - ▢ *taille des mots*
  - ▢ *nombre de bus*
  - ▢ *type d'accès*
- ▢ Fournir un modèle simulable pour la vérification
- ▢ Fournir un modèle physique
- ▢ Fournir des « spécifications » (vitesse, conso...)

# Alternatives

---

## ▣ 2 solutions possibles

- ▣ *Bloc paramétrable synthétisable*

*(description VHDL, Verilog, générateur de netlist....)*

- ▣ *Bloc « cousu-main » (dessin dans la technologie cible)*

- ▣ *Dans la pratique le fondeur dispose de générateurs*

- ▣ *Ces générateurs ne sont pas accessibles à l'utilisateur final*

- ▣ *Génération à la demande.*

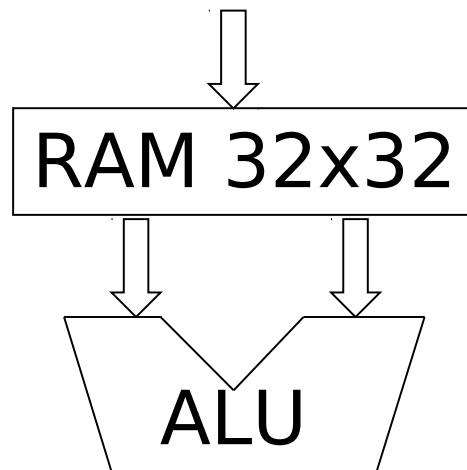
# Bloc paramétrable synthétisable

---

- ▢ Utiliser une bibliothèque précaractérisée (latches, D-flipflops)
- ▢ Générer une netlist contenant le réseau de bacules, les décodeurs d'adresse, les contrôleurs
- ▢ Dans de nombreux cas, le concepteur peut coder lui même le générateur en VHDL ou Verilog ( mais peut être utile pour diminuer le temps de conception (FIFO, bancs multiports)
- ▢ Pas efficace :
  - ▣ *Taille mémoires*
  - ▣ *Vitesse de fonctionnement ??*
- ▢ Vitesse déterminée par le flot de conception classique

# Bibliothèques DesignWare

- ▣ Exemple : Synopsys DesignWare
- ▣ DW\_2r\_w\_a\_lat : RAM asynchrone triple ports
- ▣ Basée sur des latches (Max 256 mots de 256 bits)
- ▣ Pratique pour faire un banc de registres...



# Bloc cousu main

---

- ▢ **Difficulté : fournir un modèle de temporel précis**
  - ▢ *Comportements électriques complexes dans une RAM ou une ROM*
  - ▢ *Il n'y a pas que de la logique CMOS (précharge, amplis différentiels...)*
- ▢ **Solution 1 (pour le concepteur du générateur): Générer tous les cas possibles et se baser sur des simulations électriques...**
- ▢ **Solution 2: Générer des cas pathologiques et se baser sur un modèle analytique des temps de propagation compatibles avec ces cas pathologiques..**

# Bloc paramétrable cousu main

## Le chemin critique dans une ROM

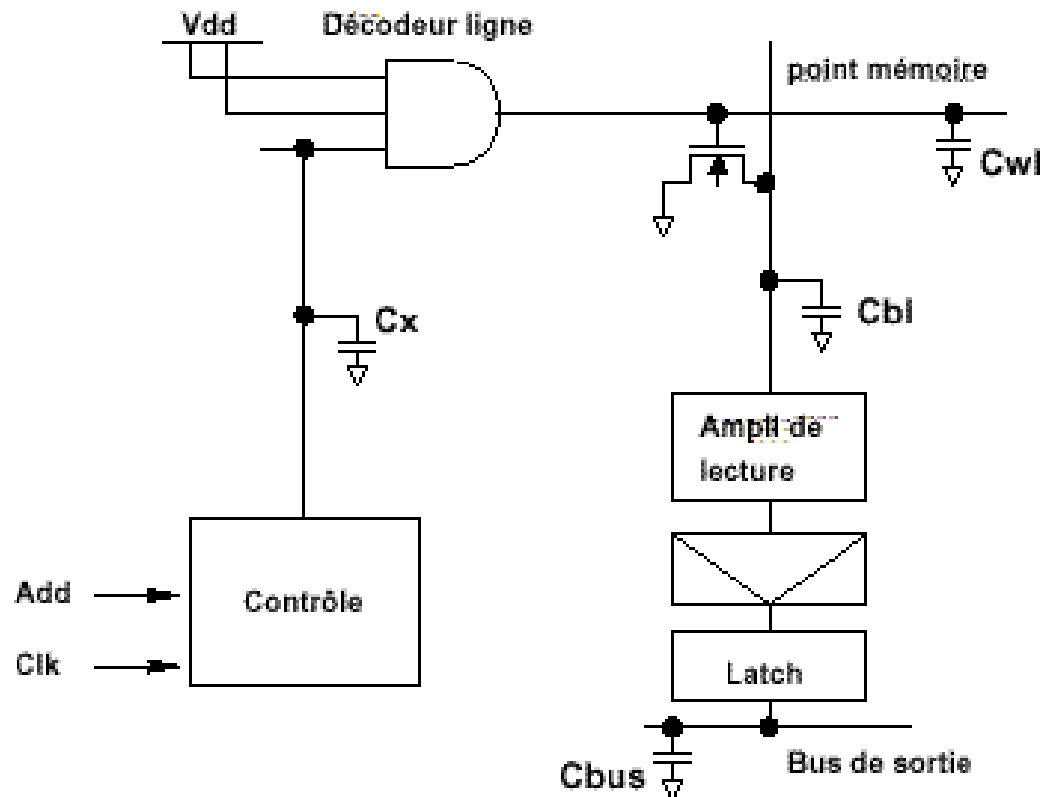


FIG. 4.1 – Schéma simplifié d'un chemin critique pour une ROM

# Circuit de test de générateurs de ROM

## ▮ Quelques instances de ROM sortant d'un générateur

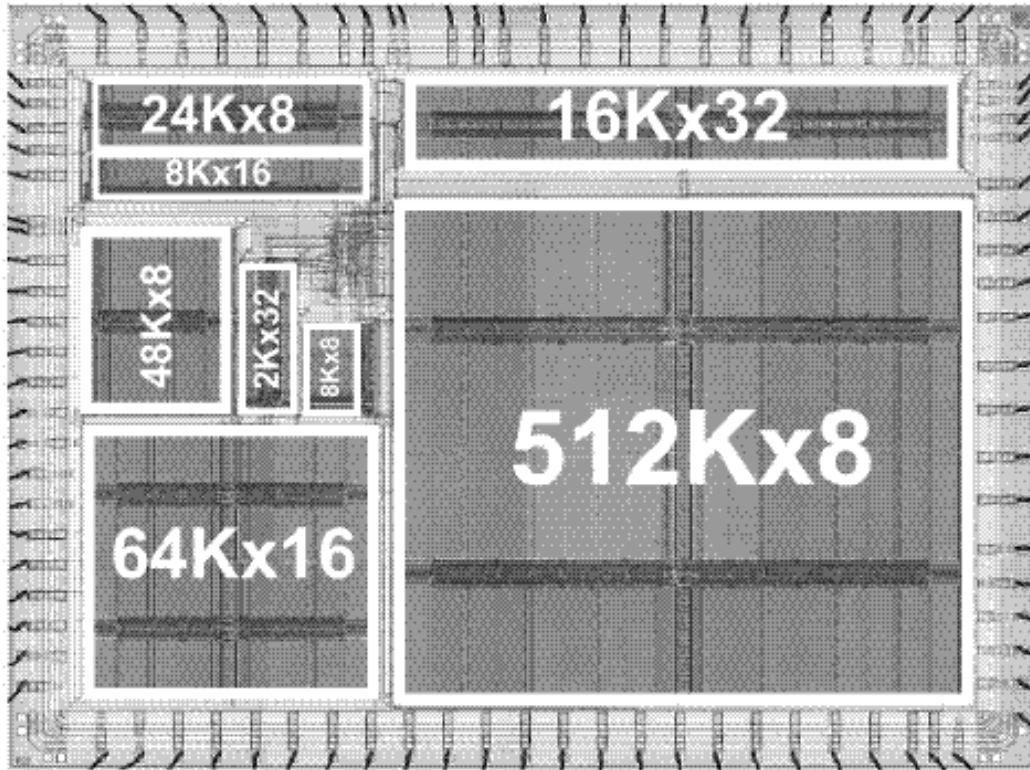


FIG. 2.13 – Testchip avec 8 instances

*Remarques:*

1/ Surfaces très différentes entre

2kx32 et 8kx8

2/ Facteurs de forme très divers

- **Source : Thèse de Mr Arnaud Turier (2001). ([www.lip6.fr/reports](http://www.lip6.fr/reports))**

# Caractéristiques des RAM intégrées

---

## ☒ Ram SYNCHRONES / ASYNCHRONES

☐ Petite taille / Grande Vitesse

☐ Grande taille Petite vitesse

☐ Simple Port

☐ True Dual port

☐ Mode test :

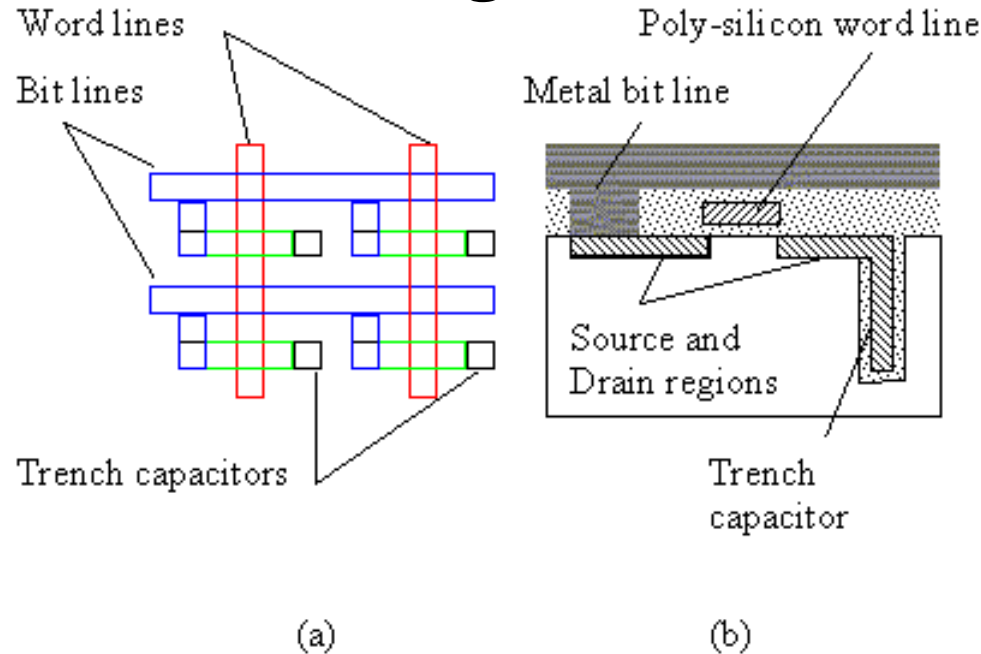
☐ *Bypass logic*

☐ *Built In self Test*

- **DRAM, seule solution pour le stockage de masse**
- **Pas forcément proposée par tous les fondeurs**
- **Problèmes de rendement (redondance,...)**

# Technologie DRAM spécifique

- ▮ point mémoire à 1 transistor
- ▮ Capacité de stockage en structure verticale



- Solution : mélange techno ASIC / techno DRAM
- Coût élevé

# Technologie ASIC CMOS

---

- ▢ point mémoire à 3 transistors
  - ▢ Pas de masque « capacitif » particulier
  - ▢ 1 grille de transistor pour la capa
  - ▢ 1 transistor de lecture
  - ▢ 1 transistor d'écriture
  - ▢ Pas de lecture destructive
- Peu dense
  - Faible coût

# DRAM en techno ASIC

---

- ▣ **Caractéristiques en vitesse proches des « vraies » DRAM**
- ▣ **Taux d'intégration plus faible**
- ▣ **Des méthodes d'accès plus simples que la DRAM standard**

- **Les phases du placement/routage**
- **Bibliothèques de cellules**
- **RAM, ROM et DRAM Embarquées**
- *Génération de Chemins de Données*

- ▢ **Les systèmes numériques se découpent en « partie contrôle » / « partie calcul »**
- ▢ **Les parties « calcul » sont**
  - ▢ *Souvent causes des chemins critiques*
  - ▢ *Souvent organisées autour de transferts de données de tailles fixes (8b, 16b, 32b...)*
- ▢ **Des techniques spécifiques peuvent être utilisées pour optimiser ces parties**

# Solutions non liées à la technologie

---

- ▣ **Les vendeurs d'outils de CAO proposent des compilateurs de DATAPATH :**
  - ▣ *Extraction des chemins de données dans le code VHDL*
  - ▣ *Proposition de solution « globale » pour chaque opérateur arithmétique (choix de multiplieurs, choix d'additionneurs...)*
  - ▣ *Pas de notion de « bit-slice » : l'outil de placement n'exploite pas la structure globale du chemin données*
- ▣ **Exemple :**
  - ▣ *Datapath Compiler (Cadence)*
  - ▣ *DesignWare+Module Compiler (Synopsys)*

# Solutions intermédiaires

---

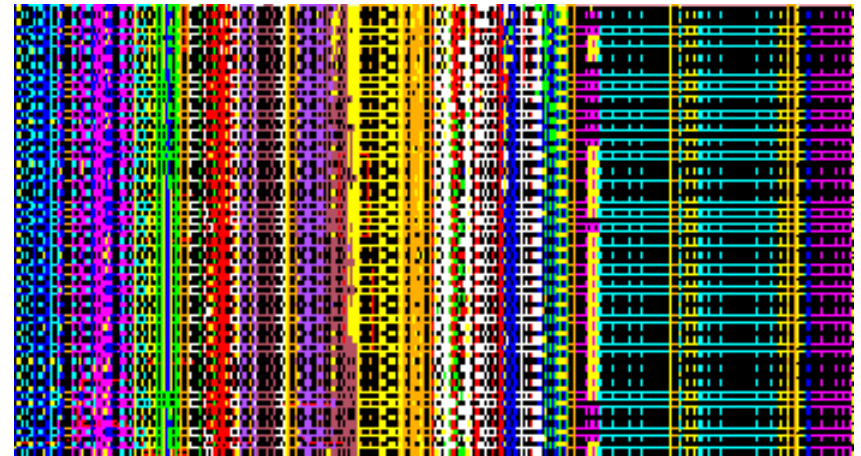
- ▮ Des vendeurs d'outils de CAO proposent des optimiseurs de placement de datapath en cellules standard
- ▮ Exemple Arcadia Design (outil Mustang) (<http://www.arcadiadesign.com>)

# Mustang

Placement « classique »



Placement « datapath »



Les couleurs des cellules correspondent aux fonctions  
Gain en densité  
Gain en vitesse