

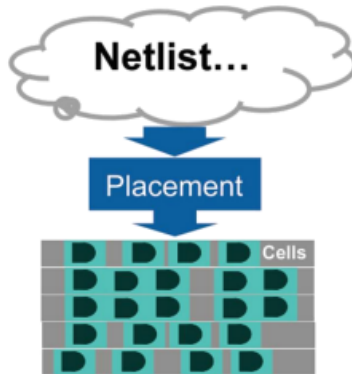
# Placement (V1)

Written by: Fatma Ali

## Content:

- Introduction
- Coarse placement Vs Legalized Placement
- High fanout synthesis (HFS)
- Tie high and Tie low cells
- Scan chain reordering
- Logical restructuring
- Congestion
- Placement blockages, keepout margin and placement bounds
- ICC tool

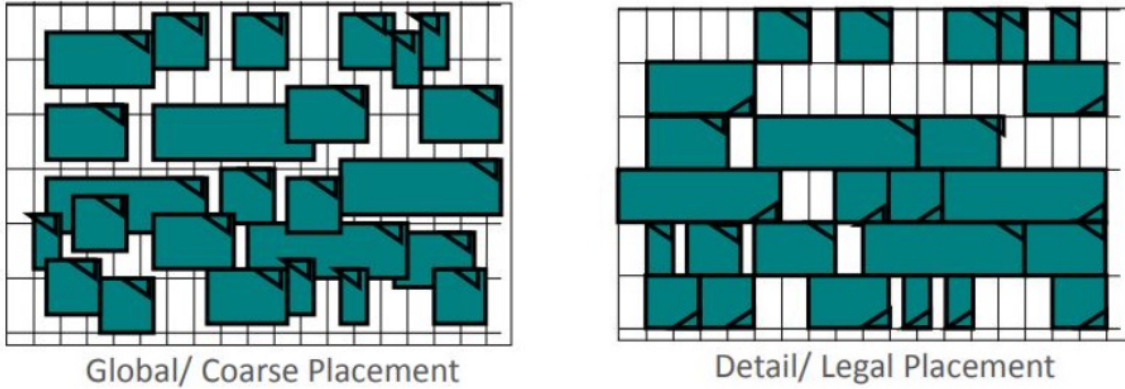
## Introduction:



لحد هنا احنا جهزنا ال floor plan & power plan و خلاص كده كله جاهز والخطوة الجاية ان ال tool تحط ال cells فى اماكنها بالفعل وهى دى خطوة ال placement الى هنعملها دلوقتى فاحنا متوقعين اننا هنخرج من الخطوة و خلاص كل ال cells محطوطة فى اماكنها ونحل اى مشاكل ممكن تحصل نتيجة انهم اتخطوا ... يعنى دلوقتى فيه مساحة فاضية لل tool وعندى site rows محددة الامكن الى ممكن ال cells تاخذها وعندى netlist فيها ال cells بتاعة ال design فالمطلوب من ال tool انها تحط ال cells فى الامكن المتاحة ليها بحيث تضمن انها تحقق ال constraints بتاعتي زى ال area و ال power و ال timing و كمان يكون فى الآخر routable design يعنى تسبيلي اماكن كويسة لل routing ومبيقاش ال design فيه congestion ... يعنى فى ال placement ال tool هتخط ال cells و كمان هتعمل optimizations

ال commands بتاعة ال placement حطها فى نفس ال script عادى الى بدأنا بيه ال PnR كله زى ما قلنا احنا شغالين على انه كله script واحد لل PnR بس بنشرح كل topic بشكل منفصل وحاليا هنشرح ال placement الاول وال concepts الى فيه ثم نبقي نشوف ال commands فى الآخر ... خلىنا الاول نشوف هتخط ال cells ازاي وهل هتخطها مرة واحدة ولا على خطوات ثم نشوف بقى ايه المشاكل الى ممكن تظهر بعد كده ونحتاج نحلها وال optimizations الى نعملها

## Coarse (Global) Placement Vs Detailed (Legalized) Placement



ال placement بيتنم على خطوتين اول واحدة وهى ال **Coarse** او يطلق عليها بردوا **Global** وهنا ال tool بتخط ال cells بشكل مبدئى بمعنى انها بتتعامل مع ال cells من خلال التوصيلات بينهم يعنى كان كل cell هى عبارة عن مجموعة من ال pins بتوعها الى بتوصل ب cells تانية وهكذا فبتحاول تحطهم بحيث ان ال cells الى واصله ببعض تبقى قريبة يعنى كإنها بتعمل virtual connections بين ال cells وبعضها عشان تقدر تحطهم بشكل مناسب بحيث يحصلش congestion ويكون ال routing بينهم سهل بعدين ولكن هنا ال tool مش بتهتم لو فيه cells مثلا overlapped على بعض او مش محطوطين بشكل مضبوط فى ال site rows ويمكن ال orientation مش مضبوط مثلا cell المفروض تتخط مقلوبة (زى ما قلنا فى ال floor plan ان ال rows بنقلب فيها ال cells عشان تعرف تشارك VDD & VSS مع ال rows الى قبلها وبعدها) بس تلاقيها محطوفة معدولة عادى ... فى ال Coarse احنا بس بنخط ال cell بشكل مبدئى فى مساحة معينة من ال core

فى ال **Detailed** او بيطلق عليه بردوا **Legalized** بنخط بقى ال cell مضبوطة يعنى ال tool بتشوف اقرب مكان ليها فى ال site rows مضبوط وتحطه فيها وتضبط ال orientation بحيث نخرج من الخطوة دى وتكون خلاص كل ال cells محطوفة فى اماكن مضبوطة وتحاول طبعا وهى بتختار الاماكن القريبة ليها فى ال site rows انها متعملش congestion بردوا وتقال طول ال wire الى هيوصل بين ال cells على قد ما تقدر يعنى تقريهم من بعض على قد ما تقدر

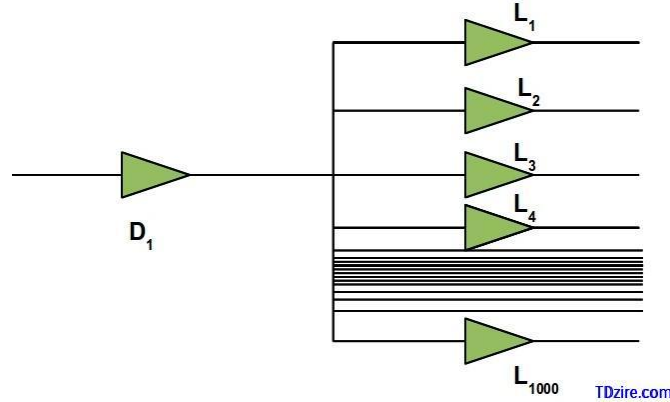
يعنى كذا الاول ال tool بتعمل Coarse placement تحط ال cells فى اماكن مبدئية بغض النظر عن اى rules بتحكم الاماكن دى هى بس بتخطها فى ال core وخلاص بحيث تقلل ال congestion بقدر الامكان وتخلي ال design بيبقى routable ثم تروح للخطوة التانية وهى ال Detailed Placement تحط بقى ال cells فى الاماكن المتاحة ليها المضبوطة حسب ال rules الى اتحطت قبل كده فى ال floor plan ببقى كإن ال Coarse بتسهل الدنيا شوية لل Detailed وتعملها حاجة مبدئية كده تكمل من بعدها

خلاص كده حطينا ال cells نروح بقى نشوف ايه المشاكل الى ممكن تظهر ووال tool بتحلها فى ال step دى وايه ال optimizations الى هى بتعملها اصلا عشان ت meet ال constraints بأفضل شكل

=====

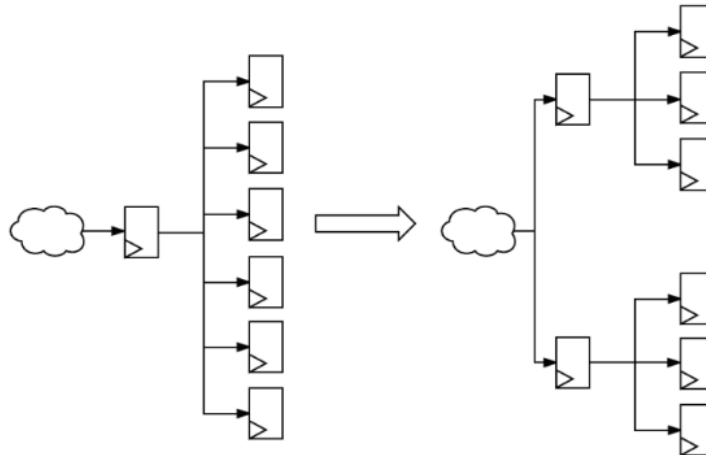
## High Fanout Synthesis (HFS):

دى اول مشكلة هنتكلم عنها ودى ال tool بتحلقها فى Coarse placement ... دلوقتى اى net فى ال design الى هى اى توصيله بين cell والتانية بيكون فيه constraints محطوط لل max fanout فى ال design كله بشكل عام يعنى بقول مش عايز اى net فى ال design ببقى عندها fanout اعلى من كذا ... وده بيمثل عدد cells والى هى بتكون جاية من ال cells الى واصلة على ال output بتاع cell دى ... خيلنا طيب نوضح اكثر حنة ال fanout من الصورة الى تحت دى:



دلوقتى  $D_1$  cell الى هى هنا نوعها buffer واصل على ال output بتاعها cells تانية كتير هنا عددهم 1000 ودول بقى بيمثلوا ال fanout فكدا بنسمى  $D_1$  driver على اساس انها بت drive كل ال cells الى واصلة على ال output بتاعها

فاحنا بقى زى ما قلنا فى ضمن ال constraints بتاعتنا الى بنجهزها من وقت ال synthesis بيكون فيه واحد بيحدد ال max fanout وده لان ال fanout العالى اوى هيسبب delay كبير لان ال transition time بتاع ال  $D_1$  output هنا هيكون عالى لانها بت drive عدد كبير من ال cells .. وكمان ال transition time العالى ده ممكن يسبب crosstalk هنتكلم عنه بعدين بشكل منفصل ... فممكن بعد ال placement الاقى ان فيه net عندها fanout عالى بي violate ال constraint الى حايطينه فالى بيحصل ان ال tool بتبدأ تقسم ال path ده لعدد من ال paths يعنى ال driver cell مثلا بدل ما تبقى بت drive عدد كبير من ال cells وليكن 20 لا نخليها ت drive مثلا 5 بس وكل واحد من الخمسة ي drive عدد بردوا وهكذا فهلاقى ان كل net بقت بت meet ال constraint ... فى الصورة تحت مثلا بدل ما كانت net واحدة عليها 6 cells قسمتهم وقللت ال fanout عليها:

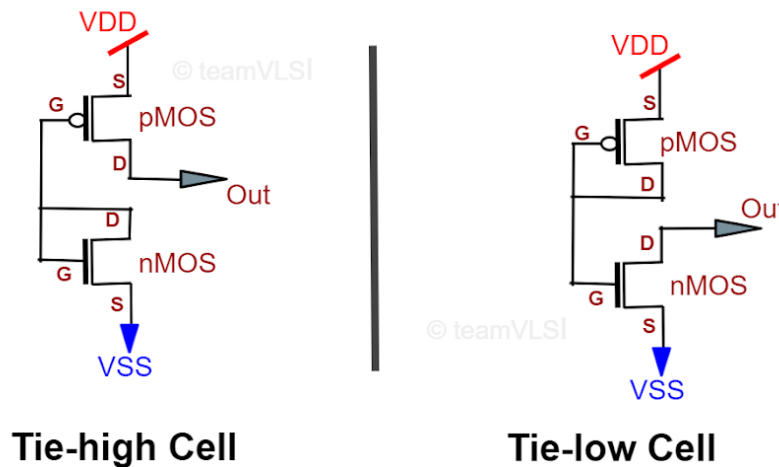


## Tie high and Tie low cells:

فيه cells تانية ال tool هتحتها وهي مش في ال design بتاعى بس لازم تتحط عند ال input بتاع بعض ال cells وهي tie cells طب ايه دي وليه لازم احطها؟؟ دلوقتي احنا عارفين ان فيه cells بيكون ال input بتاعها high يعني VDD او low يعني VSS ... لان مش لازم ال input بيقى جاي من cell قبلها ممكن يكون input ثابت ليها VDD or VSS وفي الحالة دي بنجيبه من ال supply ... فحتى الان الى في خيالنا انه ال wire الى جاي من power mesh وهيوصل لل cell خلاص هاخذه اوصله على input pin على طول ... بس ده مش صح لان مهما حصل وحاولت انك تقلل supply noise هيفضل فيه noise وفيه ومش هتضمن انه pure VDD or pure VSS واحنا عارفين ان ال input بتاع اى cell هو في الاخر واصل على gate of a transistor الى هي polysilicon تحتها مع تقدم ال technology بيكون more thinner و sensitive اكثر فممكن يبيوظ بسبب ال noise دي ... فبنحط cell ملهاش اى input هي ليها VDD & VSS وبنسميها tie cell

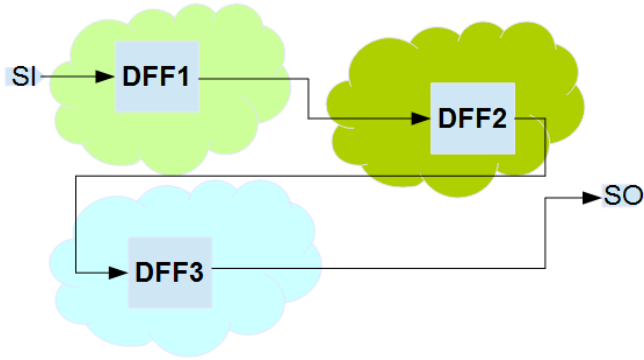


واما ان ال output بتاعها دايمًا high وبنسميها tie high cell وتوصل على gate input الى محتاج يوصل ل VDD او ان ال output بتاعها بيقى low دايمًا وبنسميها tie low cell وتوصل على gate input الى محتاج يوصل ل VSS

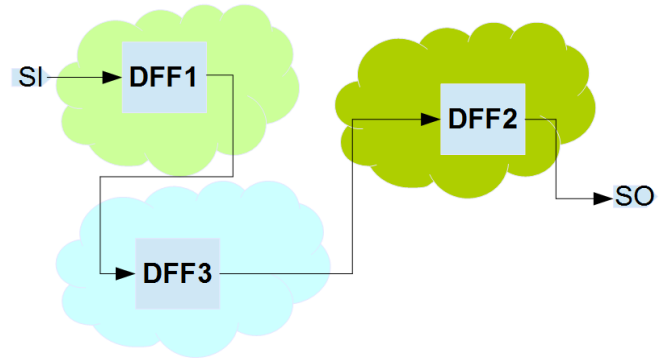


## Scan Chain reordering

ده يخص ال DFT ... في DFT احنا حطينا scan flipflop وبقى عندنا scan chain الى هي ال flops الواصلة على بعض (ممكن ترجع ل DFT PDF تفهم منه ال concept تاني لو متعرضتش ليه قبل كده او مش فاكركه كويس) فالمهم ان ال chain دي كانت واصلة ببعض بطريقة بعد ال DFT compiler ... دلوقتي بقى اما حطيت ال cells بالفعل ممكن الاقي ان التوصيله دي مش احسن حاجة دلوقتي:



Physical Synthesis Netlist: Before Reordering

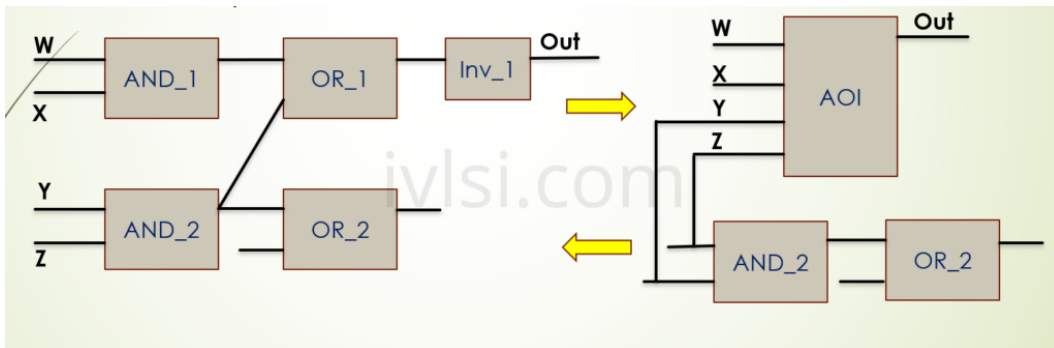


Physical Sythesis Netlist: After Reordering

يعنى مثلا فى المثال الى فى الصورة قبل reordering هلاقى ان فيه path طويل بين DFF2 & DFF3 فده wire طويل اوى فى حين انه عادى اغير ترتيب ال chain هنا واخله زى ما بقى after reordering فالاقى انه wires كده كلها كويسة ... انا غيرت ترتيب ال flops فقط فمأثرتش على ال functionality فى حاجة بس الى هيتأثر عندى هنا هو generated pattern الى كنت مطلعه قبل كده باستخدام ATPG tool زى ما اتكلمنا فى DFT ... لانه طالع على ترتيب معين لل flops بس ده مش مشكلة لإنى اقدر ادى ATPG الترتيب الجديد عن طريق انى ادخلها المرة دى ال netlist الجديدة وتطلعلى pattern ليها عادى

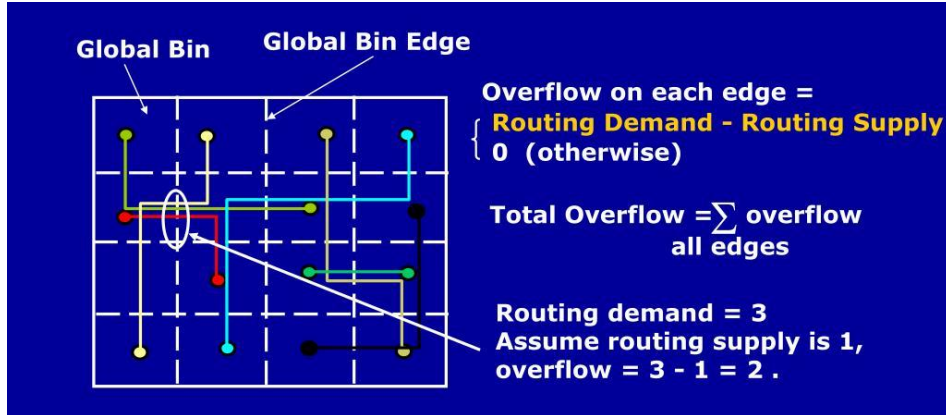
## Logical restructuring

ده معناه ان ال tool بتستبدل logic معين بواحد تانى بس يعمل نفس ال functionality بهدف انه يحسن من power او area او timing وده زى مثلا ان ممكن يكون عندى كذا cell ورا بعض وهما سوا كدا بعملوا function معينة فممكن الاقى ان فيه cell اصلا فى library الى انا شغال عليها بتعمل نفس ال function دى فهتبقى customized يعنى معمولة مخصوص لل function دى وبالتالي ممكن مثلا تبقى هتوفرلى فى area ... حتى لو هى اصلا معمولة من جوا بنفس ال cells الى انا حاططها دى ورا بعض بس هتبقى قريبة من بعض اكثر و routing فيها افضل مثلا على اساس انها معمولة مخصوص لل function دى على عكس ما ال tool بتاعتى شيفاهم مجرد cells فى design كبير مش هتديهم اهتمام خاص يعنى ... من اشهر ال cells مثلا AND OR Inverter AOI فمثلا هتشيل AND OR الى ورا بعض واحط مكانهم ال cell دى او العكس تشيل AOI وتحطهم three cells ورا بعض حسب الانسب ايه يعنى مش لازم دايما يبقى الانسب انى اجمعهم واستخدم cell واحدة تقوم بنفس ال function ممكن يبقى العكس انسب هى ال tool بتشوف الانسب بحيث تحقق ال constraints الى انت حاططها من حيث power, timing, area

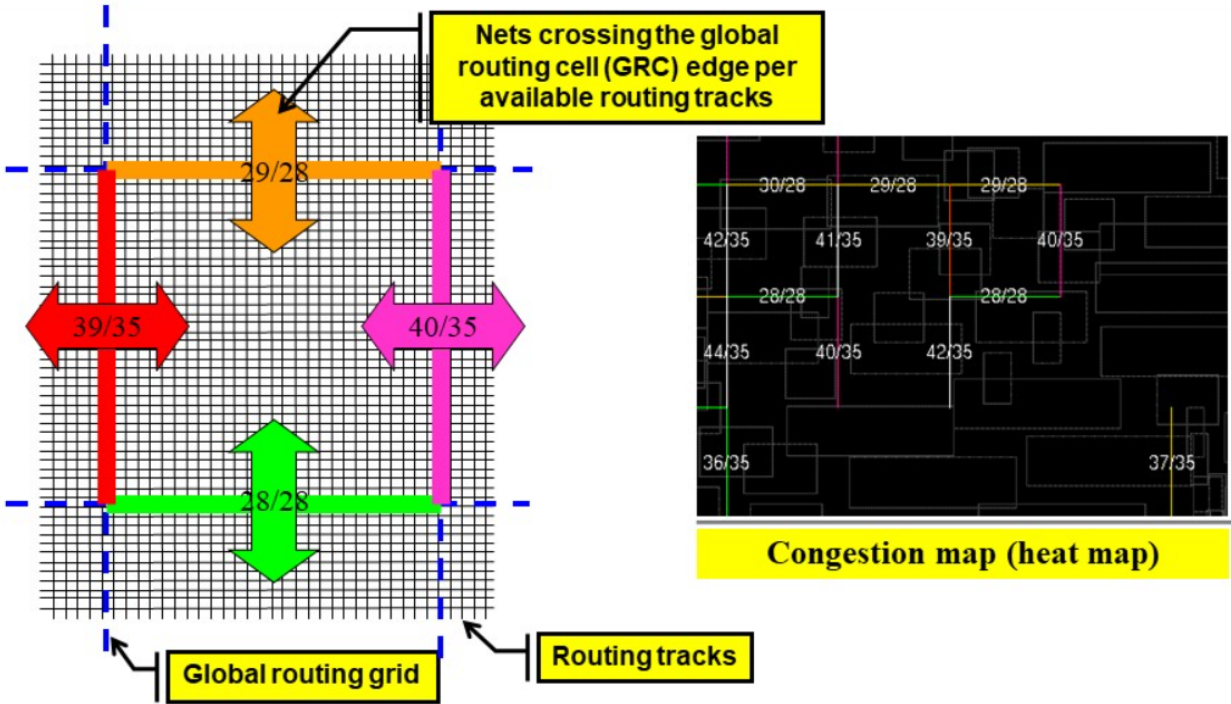


## Congestion:

احنا عاملين نقول من بدرى ال tool بتحط ال cells بحيث متعملش congestion واحنا عارفين ان معناها زحمة يعنى ... بس خاينا هنا نشوف ازاي اصلا ال tool بتبص لل design وتكتشف ان فيه congestion وتحاول تحله ... دلوقتى احنا من ايام ال floorplan كنا محددين ال wire tracks فى كل metal layer فانا دلوقتى لو جيت بصيت على اى مكان عندى فى ال core هتقدر ال tool تحدد فيه tracks قد ايه فى الحتة دى وهى دى كذا ال available routing resources ... وبعد ما حطت ال cells بردوا هى عارفة كل مكان محتاج يكون فيه routing resources قد ايه عشان اوصل ال design زى ما مطلوب فلو فلا مكان محدد لقت انه ال needed اكثر من ال available يبقى كده هنا فى المكان ده فيه congestion زى المثال فى الصورة محتاج three tracks بس ال available واحد بس:



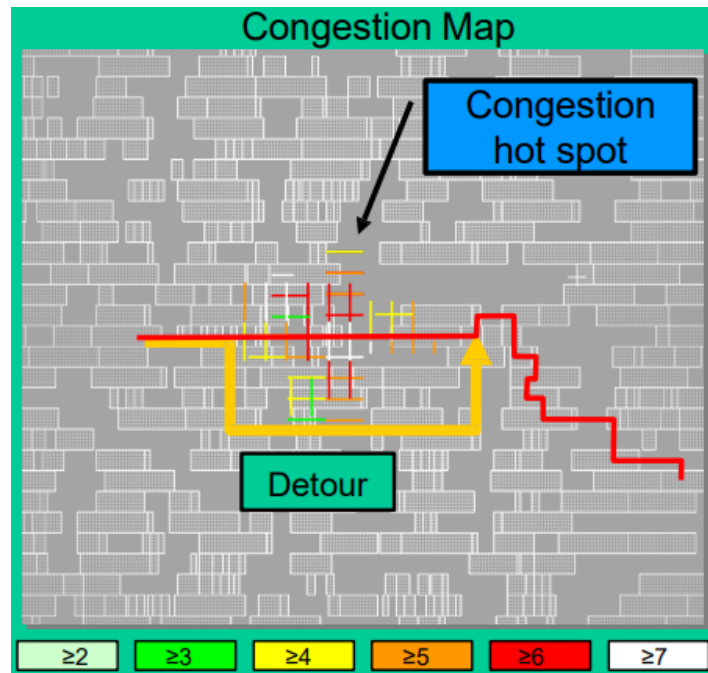
فيه عندنا congestion map بتساعدنا نعرف الاماكن الى فيها congestion وفيها overflow قد ايه .. هتلاقى فيه ارقام مخطوطة على الخطوط المتلوونة تحت هى عاملة زى الصورة الثانية على الشمال بس اول صورة على tool والثانية شكل مبسط عشان نفهم يعنى:





طب هو انا دلوقتى اصلا مش فى خطوة ال routing لسه بس مينفعش اسيب ال design بتاعى وانا عارف ان فيه congestion واكمل لان ببساطة هتقابلنى مشاكل قدام زى:

- ان ال tool اصلا فى ال routing متقدرش تعمل routing يعنى مش هتعرف تتصرف فى ال overflow الموجود ده
- ممكن انها تتصرف ... بانها اما تلاقى الدنيا زحمة فى حته ومش مكفى فيدل ما تمشى فى المكان الى ال placement كان عامله لانها هتلاقى congestion فهتلف من طريق تانى وتستخدم routing tracks مختلفة وعشان لفت وخذت طريق تانى لحد ما وصلت لنفس ال pin الى عايزة توصلها بنطلق عليها detoured net زى فى الصورة كده لاقى الدنيا زحمة ومفيش مكان راح لاف من طريق تانى:



طب ايه المشكلة بقى ما هى اتصرفت خلاص؟؟ المشكلة ان فى placement كان عامل virtual tracks كده بين ال cells الى هى كان فيها congestion وعلى اساسها حسب RC estimated لل wire الى هيتحط فلما انا دلوقتى اجى احط wire بالفعل بقى فى ال routing واخليه ياخذ مسار تانى الى هو هيبقى اطول وكده ال cell الى بت drive ال wire هتشوف R&C اكبر يعنى delay اكبر و transition time اكبر وده هيجلى ال signal تبقى معرضة انها تتأثر بال crosstalk اكثر(هنتكلم عن ال crosstalk فى CTS) وكمان ال transition time الكبير ده هيكوّن input لل cell الى بعدى وده هيجلى ال delay بتاعها كبير ... وبالتالي الفرق بين estimated & actual performance هيجلى ال performance هيفتلف وممكن الاقى timing violations مكنتش موجودة وظهرت

يبقى لازم ال tool تحل ال congestion ده قبل ما تكمل باقى ال flow ممكن تحاول تحله بانها تزود ال utilization بحيث توزع ال cells على اماكن اكثر او انها تعمل placement بهدف تقليل ال congestion بدل ما يكون بهدف تقليل ال wire length بين ال cells ودول two approaches مختلفين لل tool فى ال placement اول واحد وهو **timing driven placement** وهنا بيبقى هدفها تقليل ال wire length بين ال cells بحيث تقلل RC وتقدر ت meet setup timing والثانى هو **congestion driven placement** وهنا الهدف بيبكون تقليل ال congestion

الحاجات الى احنا عملناها فى floorplan المفروض ان هى تضمن ميحصلش congestion زى & keepout margin & blockages placement of Macros وبالتالي انا ممكن ارجع اعمل تعديلات فى floorplan ممكن تعديلات فى core بمعنى مثلا ازود utilization

او انى ازود blockages فى اماكن تانية او اغير اماكن ال macros لو لاقيتها مش مناسبة ... ويمكن اعدل فى core aspect ratio  
يعنى كده هغير ابعاد ال core بتاعى ازوده مثلا ويردوا فى ال power plan اعمل تعديلات لو مثلا لاقيت انه واخذ routing resources  
كثير اوى عملت مشكلة congestion بعدين اما حظيت ال cells وجيت ابص على ال expected routing بتاعهم ... يعنى كل الحاجات  
الى انا ظبطها قبل كده فى floor plan & power plan كانت بتجهز ال core بتاعى بشكل كويس لل placement فانا لو لاقيت مشكلة  
ممکن جدا تكون جاية من انى مجهزتوش كويس فى الخطوات السابقة للى الخطوة الى انا واقف فيها فبرجع اعدل تانى فى الى فات ثم اكمل  
واشوف لسه المشاكل دى موجودة ولا لا

عشان كده احنا ذكرنا حاجات كثير نعملها فى floorplan كانت عشان نتفادى ال congestion فى ال placement زى انه نعمل  
blockages حوالين ال macro عشان ميتحطش cells حواليه وانه متحطش ال macro pins عند ال core boundary وشوفنا ال  
orientations المختلفة لل macro وايه الصح وايه الغلط وايه ال not preferred

### Placement blockages, Keepout margin and bounds:

احنا اتكلمنا عن ال placement blockages بالتفصيل فى floorplan وشوفنا كل انواعه وال commands بتاعته ... فلو انا مثلا  
لاقيت congestion عالى لسه بعد ال placement ممكن ارجع اغير فى ال commands مثلا ازود blockage فى مكان معين  
... ونفس الكلام فى ال keep out margin اتعرفنا عليه ويردوا فى ال placement لو حسيت انى محتاج اعمله لحاجة تانى ارجع واعمله

فيه حاجة كمان نقدر نعملها لو انا بعد ال placement لاقيت ال timing عندى critical ومحتاج احسنه وهى ال placement bound ..  
فكرته انى بحدد cells معينة واقول انا عايزهم يتحطوا مع بعض فى مساحة تساوى كذا وهى المساحة دى الى بطلق عليها bound وده مفيد  
فى ال critical paths بحيث اضمن ان ال tool هتجطهم جمب بعض اكيد ... زى ما ال blockage له انواع فبردوا ال bound له:

- **Hard Bounds:** هنا لازم المساحة الى هقوله عليها يحط فيها ال cells الى انا محدداها ومينفعش ابدأ يبقى فى منهم برا المساحة دى
- **Soft Bounds:** هنا مش لازم المساحة الى هقوله عليها يحط فيها ال cells يعنى انا مش هبقى ضامن ان فعلا ال cells محطوة  
فيها لو استخدمت soft bounds فهو كإنى بقول لل tool مجرد suggestion

### **:NOTES**

- فى ال timing checks انت مش مهتم بال hold وهى اصلا ال tool هنا مهتمة تحل setup violations لان ال hold هتخله بعدين  
بعد ال CTS الى هو الخطوة الجاية اما نوصلها فى ال PDF الجاى هنتكلم عنها ونشوف اشمعنا بعد ال CTS لازم يبقى مفيش ال hold  
violations وقبل كده عادى نلاقى ونعديها
- بما انه لحد الخطوة دى احنا معتبرين clock لسه ideal لانه لسه هنبدأ نخط clock network فى الخطوة الجاية فى ال CTS زى ما  
تكلمنا فى ال introduction to PnR يبقى لازم بردوا ال constraints الى هستخدمها هنا يفضل فيها ال commands الى بتخص ال  
clock الى كانت فى ال synthesis الى هى uncertainty, latency, transitions كل دول افضل محافظ على وجودهم فى ال  
placement



## ICC Tool

### 1. Checks before placement:

#### Command:

`check_physical_design -stage pre_place_opt`

ال command ده بنستخدمه من اول ال placement بحيث انه يتأكد من ال design لحد الخطوة دي تمام من خلال checks بيعملها حسب كل stage واقف فيها ... هنا بيعمل checks قبل ما نبدأ placement بعد ما يخلص بيطلع على ال error & warnings

- **stage:** هنا قبل placement فكتبنا **pre\_place\_opt** وعندنا two possible values غيرهما وهما **pre\_clock\_opt** ودي checks قبل ال CTS وفيه **pre\_route\_opt** هنعملها بردوا قبل ما نبدأ ال routing

=====

### 2. Placement Bounds:

#### Command:

`create_bound -name bound_name -type hard/soft -boundary {llx1 lly1 urx1 ury1 ...} object_list`

- **llx & lly:** lower left corner coordinates , **urx & ury:** upper right corner coordinates

هنعمل ال placement bounds الى شرحناها ... فنفس فكرة ال blockages هحط name لل bound نفسه ثم اختار hard or soft ثم هحط ال boundaries بتاعه ال bound ده وبعدين احط اسماء ال instances الى عايز احطها فيهم بحيث كل llx,lly,urx,ury دول مساحة بسمح لل tool انها تحطهم فيها عشان كده ممكن تلاقيهم overlapped عادى

=====

### 3. Start Placement:

#### Command:

`Place_opt -effort (high/medium/low) -congestion -area_recovery -continue_on_missing_scandefl -optimize_dft -cts`

ده بيعمل coarse placement, high fanout synthesis, physical optimization and legalized placement ... ودي بعض ال options المهمة فيه خلينا نفسرهم:

- **effort:** ده بيقول لل tool درجة المجهود الى تبذله في placement وطبعا كل ما كان اعلى كل ما عمل placement بشكل احسن و optimized اكثر بس في نفس الوقت هياخد وقت اكبر عشان يعملته وال default هو medium
- **congestion:** هنا بقول لل tool انها تعمل placement بهدف تقليل ال congestion على قد ما تقدر يعني كده بعمل congestion driven placement الى اتكلمنا عنه
- **area\_recovery:** هنا بقول لل tool انها تقلل ال area اكثر بمعنى انها هتسبدل ال cells ب cells اصغر ... بس كده ال delay بتاع ال path الى فيه ال cells دي هيزيد لانه زى ما شرحنا في files تانية قبل كده انه اما cell area بتقل يبقى delay بتاعها زاد وبالتالي ممكن يحصل timing violations في ال path ده فمحتاج ان تكون ال paths ليها positive setup slack كويس

- **continue\_on\_missing\_scandef**: بعد DFT سيكون عندى scan chain file ده بيكون مكتوب فيه scan chains الى فى ال design وايه ال cells الى فيهم ... ده لو مش موجود place\_opt هيطلع error message فلازم احطله ال option ده الى هو بيقوله كمل حتى لو ملقتش scan chain file ... بس هو ليه مش بيكمل الا بيه؟؟ لأنه مش هيعرف يكمل خطوة HFS لأن ال inputs & outputs الى احنا بنضيفهم زى testmode مثلا دى بتوصل على عدد كبير من flops لأنها بتوصل لل chain كلها فهو عشان يعرف يعمل HFS ليها محتاج ال file ده والا مش هيعرف وبالتالي مش هتنجح خطوة HFS
- **optimize\_dft**: عشان يعمل scan chain reordering بس محتاج ابقى قريت ال scan chain information باستخدام read\_def command
- **cts**: لو ضيقته يبقى ال command ده كمان هيعمل CTS ... هيحط ال clock tree ويعملها optimizations و routing الى هو هنعمله بشكل منفصل فى ال topic الجاى CTS ... لكن لو هستخدم ال option ده يبقى قبلها لازم اكون مجهز كل الى محتاجه ال tool قبل ما تعمل CTS زى ال constraints & targets بناعة ال clock tree وهنشوف ايه الى بتحتاجه وندخله ازاى ليها فى file بعد ده الى هو CTS file

=====

### Command:

**create\_placement -effort (high/medium/low) -congestion -congestion\_effort (high/medium/low) \**  
**-timing\_driven -quick -continue\_on\_missing\_scandef -check\_only**

- ال command ده بيعمل coarse placement فقط ... مش هيعمل legalized placement وبعض ال options بتاعته:
- **effort**: ده بيقول لل tool درجة المجهود الى تبذله فى placement وطبعا كل ما كان اعلى كل ما عمل placement بشكل احسن و optimized اكثر بس فى نفس الوقت هياخد وقت اكبر عشان يعمل وال default هو medium
  - **congestion**: هنا بقول لل tool انها تعمل placement بهدف تقليل ال congestion على قد ما تقدر يعنى كده بعمل congestion driven placement الى اتكلمنا عنه
  - **congestion\_effort**: هستخدمه لو استخدمت option ال congestion طبعا الاول ... فهنا بقول درجة المجهود الى تبذله ال tool عشان تحسن ال congestion وال default بتاعه medium
  - **timing\_driven**: هنا بقول لل tool انها تعمل placement بهدف تحسين ال timing يعنى بعمل timing driven placement الى اتكلمنا عنه فى الشرح
  - **quick**: بسرعه ال tool وهى بتعمل coarse placement
  - **continue\_on\_missing\_scandef**: بقول لل tool تكمل حتى لو ملقتش scan chain file ودى شرحناها بالتفصيل فى place\_opt command
  - **check\_only**: ده بستخدمه عشان اعمل check هل كل الى ال command محتاجه موجود ولا لا ... وبالتالي مقدرش استخدمه مع اى option تانى لأنه انا لسه هنا بعمل check مش هعمل placement

=====

## **Command:**

**legalize\_placement -effort (high/medium/low) -incremental -timing -check\_only**

ده بيعمل legalized placement يعنى هيتأكد ان ال cells فى اماكنها المتاحة ليها مضبوطة وكل ال constraints الى على ال cell placements مضبوطة يعنى ال legalized الى شرحناه فوق وعشان كده لو انت عملته بعد place\_opt هتلاقى هيا design الموجود وكل حاجة بس هيلقى كل حاجة مكانها غالبا فبيقولك No cells needs to be legalized ويطلعك reports كده زى الى طلعت فى place\_opt عادى فيها summarize ... انما لازم تستخدمه لو عملت create\_placement لإنك هتبقى عملت coarse بس ولسه هتعمل legalized بده ... ده بعض ال options بتاعته هيا:

- **effort**: ده بيقول لل tool درجة المجهود الى تبذله فى placement وطبعا كل ما كان اعلى كل ما عمل placement بشكل احسن و optimized اكثر بس فى نفس الوقت هياخد وقت اكبر عشان يعملته وال default هو medium
- **incremental**: يعنى هيعمل changes بسيطة ثم ي check لو لسه مش تمام يعمل غيرها وهكذا يعنى هيا بتحرك عدد صغير من ال cells بقدر الامكان بس ممكن تحركهم لاماكن بعيدة عن الى موجودة فتأثر على ال timing ... ده بنستخدمه بنسبة كبيرة بعد ما نكون عملنا legalized placement اصلا قبل كده بس عايزين نغير فى الموجود شوية عشان نعمل optimizations لو استخدمتها مع option ال timing مش هتتحرك ال cells فى timing critical paths لأماكن بعيدة بحيث مياثرش على ال timing فيهم ... لإ العادى لل option ده هيعمل incremental placement على اساس حجم ال cells ويختار الامكان المناسبة من غير ما يهتم بال critical paths
- **timing**: بيقلل حركة ال cells فى critical timing paths او بمعنى اصح متتحركش بعيد اوى عن مكانها الحالى عشان ميحصلش timing violations ... يعنى هنا بخلى ال tools على اساس ال worst pin slack مش على اساس ال cell area
- **check\_only**: ده بنستخدمه عشان اعمل check هل كل الى ال command محتاجه موجود ولا لا

=====

## **4. Placement optimizations**

### **Command:**

**refine\_placement [-congestion\_effort (high | medium | low)] [-continue\_on\_missing\_scandef] \ [-perturbation\_level min | medium | high | max]**

ده بيعمل congestion optimization فقط ... يعنى بعد ما اخلص legalized placement & coarse لو لاقيت لسه فيه congestion عالى فاقدر استخدم ال command ده اعمل بيه optimization .. بعض ال options هيا:

- **congestion\_effort**: هنا بقول درجة المجهود الى تبذله ال tool عشان تحسن ال congestion وال default بتاعه medium
- **perturbation\_level**: ده درجة الاختلاف الى تعملها ال tool عن ال initial placement الى معمول دلوقتى يعنى مثلا لو قولتلها min فهيا هتبذل كل مجهودها انها تحسن ال congestion بس باقل تغيير ممكن فى اماكن ال cells دلوقتى يعنى تعمل fine tuning وكل ما زودت فى level هيحصل تغيير اكبر فى الامكان عن ال initial placement ... وال default هو medium
- **continue\_on\_missing\_scandef**: بيقول لل tool تكمل حتى لو ملقتش scan chain file .. ودى شرحناها فى place\_opt

### Command:

`set placer_enable_enhanced_router true | false`

ده variable اسمه `placer_enabled_enhanced_router` بياخد `true` or `false` فيستخد `set command` الى في TCL عشان اديله `value` بس ده اصلا بيعمل ايه؟؟ بعمل `optimization` لل `congestion` منه ... زي ما احنا اتكلمنا في الشرح انه من خلال `routing estimations` او يعني `virtual routing` بتقدر ال `tool` انها تعمل `course placement` وزى ما هنشوف في ال `routing estimation` لل `global routing` وهنفهم ايه ال `global` وال `detailed` وقتها فالمهم ان ال `variable` ده لما اعمله `true` بخلى ال `coarse` يستخدم ال `global route` الحقيقي الى في ال `tool` بدل ال `estimator` فده بيخليه يقدر يحسن ال `congestion` ... يعني هو بيخلي ال `coarse placement` بدل ما يعتمد على حاجة `estimated` يعتمد على حاجة واقعية اكثر

=====

### Command:

`psynopt [-area_recovery] [-power] [-congestion] [no_design_rule | -only_design_rule]\`  
`[-size_only | -in_palce_only] [-only_area_recovery] [-only_power] [-continue_on_missing_scandef]`

ده بيعمل `optimizations` ومش لازم استخدمه هنا في ال `placement` بس لا عادى انا ممكن كمان بعد ال `routing` اخليه يعمل `final optimizations` فهو بي `optimize` ال `design` بشكل عام .... واقدر اتحكم في ال `optimizations` الى تتعمل من ال `options` .. بعض ال `options` هي:

- **area\_recovery**: هنا بقول لل `tool` انها تقلل ال `area` اكثر بمعنى انها هتسبدل ال `cells` ب `cells` اصغر ... بس كده ال `delay` بتاع ال `path` الى فيه ال `cells` دي هيزيد لانه زي ما شرحنا في `files` تانية قبل كده انه اما `cell area` بتقل يبقى `delay` بتاعها زاد وبالتالي ممكن يحصل `timing violations` في ال `path` ده فمحتاج ان تكون ال `paths` ليها `positive setup slack` كويس
- **congestion**: هنا بقول لل `tool` انها تعمل `optimizations` لتحسين ال `congestion`
- **power**: هنا بقول لل `tool` انها تعمل `optimizations` لتحسين ال `power`
- **no\_design\_rule | only\_design\_rule**: بختار واحدة منهم وهنا ببقى عايز اقول هل تعمل `fixing` لل `design rules` `violations` بس يعني متعملش `fixing` لل `timing violations` فاستخدم `only_design_rule` او انى اقول لل `tool` متعملش `fixing` لل `design rules violations` اصلا فاستخدم `no_design_rule` ... لو مستخدمتش ولا واحدة فهتمشى على ال `default` وهى انها تعمل `fixing` للالتنين ال `design rules violations & timing violations`
- **size\_only | in\_palce\_only**: الالنتين دول بيخلوا ال `optimizations` تكون بتغيير حجم ال `cells` وبس يعني يحصل `remove` او `add` ل `cells` ... الفرق بينهم ان `in_place_only` بتحط `constraints on ECO placement change` ... طيب ال `ECO` دى اختصار ل `Engineering change order` فهنا المقصود هو التغير الى يحصل في ال `physical placement` الموجود دلوقتى فيعنى دى مش هتغير في امكانهم الموجودة ... لكن `size_only` ممكن تغير عادى مش بتحط `constraints on ECO` فاحنا بنستخدم واحد من ال `options` دى بس ... ويكون `recommended` انك تستخدم `in_place_only` بعد ما تخلص `routing` يعني في تقفيل ال `design` خلاص ويكون `legalized design` بحيث انك تعمل `optimizations` من غير تغييرات في ال `design` بشكل كبير فمحتاج `option` يحط `constraints on ECO`

- **only\_area\_recovery**: يعني تعمل optimization لل area recovery لكن متعملش لل design rules او ال timing وبالتالي ده مقدرش استخدمه مع وجود only\_design\_rule & no\_design\_rule options
- **only\_power**: يعني تعمل optimization لل power فقط
- **continue\_on\_missing\_scandef**: بقول لل tool تكمل حتى لو ملقتش scan chain file ودي شرحناها في place\_opt

=====

## 5. power/ground connection

### Command:

`derive_pg_connection -power_net VDD -power_pin VDD -ground_net VSS -ground_pin VSS`

هنعمل ال connection لل power & ground nets وده شرحناها بالتفصيل في floorplanning وقولنا هنكررها كل ما نخط cells جديدة وده الى حصل هنا اننا عاملناها بعد ال placement

=====

## 6. Tie cells insertion

### Command:

`connect_tie_cells -objects {object_coll} -obj_type port_inst | cell_inst | lib_cell\  
[-tie_high_lib_cell lib_cell] [-tie_low_lib_cell lib_cell] [-tie_highlow_lib_cell lib_cell]\  
[-tie_high_port_name port] [-tie_low_port_name port] [-max_fanout number]\  
[-max_wirelength number] [-incremental true | false]`

هنا هخط ال tie high & tie low cells الى اتكلمنا عنهم في الشرح

- **objects**: ده option لازم يبقى موجود ... هخط فيه الحاجة الى عايز اوصلها على ال tie high & tie low cells مثلا ممكن اخط اسم cells عندى فيبقى اى input ports لل cells دى واصلة على VDD & VSS هتوصل على tie high & tie low cells ... ويمكن اخط port معين يعني هحدد input port لل cell واقول ده الى هيوصل عليهم على حسب هو واصل على VDD ولا VSS هيوصل على tie high او tie low

- **obj\_type**: ده option لازم يبقى موجود ... هخط فيه نوع ال elements الى انا حطيتها في object هل حطيت ports فاستخدم هنا port\_inst ولا حطيت اسماء instant of cells فاستخدم cell\_inst ولا حطيت اسم cell بشكل عام من ال library فاستخدم lib\_cell ... ناخذ بالنسبة لاسم ال cell في library واحد بس انا بستخدمها كتير في ال design فكل مرة استخدمها فيها بتبقى instant عندى وله اسم يميزه عن باقي ال instants الى جاين من نفس ال library cell

- **tie\_high\_lib\_cell**: هخط فيه اسم ال cell الى هستخدمها ك tie high cell فلزام استخدم معاه option tie\_low\_lib\_cell

- **tie\_low\_lib\_cell**: هخط فيه اسم ال cell الى هستخدمها ك tie low cell فلزام استخدم معاه option tie\_high\_lib\_cell

- **tie\_highlow\_lib\_cell**: هحط فيه اسم ال cell الى هستخدمها ك tie low cell & tie high cell يعنى هي cell واحدة ببقى فيه port فيها يمثل logic high و port تانى يمثل logic low يعنى مش output واحد وبالتالي لازم استخدم معاها options تانى وهما tie\_high\_port\_name & tie\_low\_port\_name
- **tie\_high\_port\_name**: هكتب فيه اسم ال port الى بيمثل logic high فى ال cell الى اختارتها فى tie\_highlow\_lib\_cell
- **tie\_low\_port\_name**: هكتب فيه اسم ال port الى بيمثل logic low فى ال cell الى اختارتها فى tie\_highlow\_lib\_cell
- **max\_fanout**: هحط اكبر عدد من ال ports يتوصل على ال tie high \ tie low cell ال cell الواحدة ال output بتاعها يوصل على كام port بمعنى اخر ت drive كام cell فى ال design
- **max\_wirelength**: هحط ال max length لطول ال wire الى هيخرج من tie high \ tie low cell يوصل لل input port الى هي المفروض توصل عليه ... لو محطتش هياخد قيمة default
- **incremental**: فى ال mode ده بخلى ال tool تزود على الموجود بمعنى انه لو موجود tie cells اصلا قبل كده محققة ال max\_fanout & max\_wirelength constraints فهتسيبهم ... لكن ال default انه يكون false بحيث ان ال tool بتشيل اى tie cells موجودة وتحط من جديد

**EX:** three examples are provided from the ICC manual:

#### EXAMPLE

In the following example, all input ports of the BONUS\_SET\_\*/ cells that are tied off to power or ground are connected to new tie-high and tie-low cells.

```
prompt> connect_tie_cells -objects [get_cells "BONUS_SET_*/"] \
  -obj_type cell_inst \
  -tie_high_lib_cell TIEH -tie_low_lib_cell TIEL
```

In the following example, all input ports named A of the BONUS\_SET\_\*/ cells that are tied off to power or ground are connected to new or existing tie-high and tie-low cells that are at most 64 microns of Manhattan distance away.

```
prompt> connect_tie_cells -objects {"BONUS_SET_*/A"} \
  -obj_type port_inst \
  -tie_high_lib_cell TIEH -tie_low_lib_cell TIEL \
  -max_wirelength 64 -incremental true
```

In the following example, all input ports of the BONUS\_SET\_\*/ cells that are tied off to power or ground are connected to new tie-highlow cells. Ports that are tied off to power are connected to the OUT1 port on the TIEHL cell. Ports that are connected to ground are connected to the OUT0 port on the TIEHL cell.

```
prompt> connect_tie_cells -objects [get_cells "BONUS_SET_*/"] \
  -obj_type cell_inst \
  -tie_highlow_lib_cell TIEHL \
  -tie_high_port_name OUT1 \
  -tie_low_port_name OUT0
```



- لما كتب في get\_cells اسم cell حظ بعدها \*/\* فدى معناها انه يقصد اى cell بببدأ اسمها ب BOUNS\_SET يعنى ممكن مثلا في واحدة اسمها BOUNS\_SET\_A وواحدة تانية BOUNS\_SET\_B وهكذا يعنى مقصود بيها انه بي match اى pattern اوله BOUNS\_SET

- لما بدل cell حب يتعامل مع port فراح كتب نفس الشكل بتاع ال cell وزود "/" ثم اسم ال port وهنا A ... ال/ عامة بتعبر عن path فانا بقوله اى cells عندك اولها BOUNS\_SET انا عايز ال port الى اسمه A فيها

=====

### Command:

`derive_pg_connection -power_net VDD -ground_net VSS -tie`

ال command احنا عارفينه وعارفين انه بنعمله اما يبقى فيه cells جديدة بحيث يعمل ال power connections الفرق هنا انى بدل ما هحط option power\_pins فهو اصلا عنده option اسمه tie معناه انه يعمل ال power connections لل tie cells فلازم اكون حاطط ال tie cells اصلا قبلها

=====

## 7. Save MW cell:

### Command:

`save_mw_cel -as cell_name`

زى ما احنا عارفين اما اتكلمنا عن MW library وشرحناها بالتفصيل في floorplanning ... يبقى لازم بردوا بعد ال placement اعمل save ل MW cell جديدة الى هي ال design بتاعى لحد الخطوة دى وادبها اى اسم وليكن design\_place

=====

## 8. Check after placement:

### Command:

`check_legality -verbose`

هنا بعد ما اخلص placement بيعمل check ان ال design بتاعى legal بمعنى انه مفيش overlapped cells او cell مش في مكانها مضبوط في ال cite row او overlapping بين cell & placement blockage او بين cells & power straps ... من ال description هتلاقى لخص ال checks في دول:

```
DESCRIPTION
The check legality command checks the legality of current placement,
and prints out a report of statistics about violations. The violations
can be classified as:

1. Cells that are not on rows.
2. Cells overlapping each other.
3. Cells overlapping with blockages in the design.
4. Cells with orientation that is not allowed for the row on which a cell is placed.
5. Cells with a core site type not equal to that of the row on which cells are placed.
6. Cells overlapping with power straps in the design.
```

- **verbose:** لو انا حطيت ال option ده هيطلعلى detailed report عن ال violations الموجودة كلها قدامى في ال terminal لكن لو محطتهوش هيطلعلى message بال overall violations فممكن مثلا لو انا كتبت ال command من غيرها ولاقيت violations ابقى اعمل run تانى ب verbose اشوف التفاصيل وياه هي ال violations بالظبط وفين

هنا مثلا عملت run من غير verbose فبعد ما هيعمل load ال design هلاقى فى الاخر جايلى reports كده وهتلاقى كاتبتك اصلا استخدم verbose عشان تعرف details لأن ده كاتبتك total number of violations فى كل حاجة من ال six cases الى بيت check عليهم وذكرهم فى ال description ... هنا فى الصورة مفيش violations فكلهم ب 0:

```
*****
Check legality: Report for Fixed Placement Cells
Information: Use the -verbose option to get details about the legality violations. (PSYN-054)
*****
(fixed placement) Cells Not on Row          : 0
(fixed placement) Cell Overlaps             : 0
(fixed placement) Cells overlapping blockages : 0
(fixed placement) Orientation Violations     : 0
(fixed placement) Site Violations           : 0
(fixed placement) Power Strap Violations     : 0
*****

*****
Check legality: Report for Non-fixed Placement Cells
Information: Use the -verbose option to get details about the legality violations. (PSYN-054)
*****
Number of Cells Not on Row          : 0
Number of Cell Overlaps             : 0
Number of Cells overlapping blockages : 0
Number of Orientation Violations     : 0
Number of Site Violations           : 0
Number of Power Strap Violations     : 0
*****

Total moveable cell area: 36528.7
Total fixed cell area: 0.0
Total physical cell area: 36528.7
Core area: (20000 20000 3364000 3352000)
1
icc_shell>
```

---

### **Contact info:**

Linkedin profile: [linkedin.com/in/fatma-ali-57b1a6200](https://www.linkedin.com/in/fatma-ali-57b1a6200)

E-mail: [fatma.ali.2028@gmail.com](mailto:fatma.ali.2028@gmail.com)

All ASIC files are provided on VLSI - ASU Community:

[https://drive.google.com/drive/folders/1aLwCLZj0YG9KJhln1D60\\_nWM7p-L3q9a?hl=ar](https://drive.google.com/drive/folders/1aLwCLZj0YG9KJhln1D60_nWM7p-L3q9a?hl=ar)

---