Routing a Transmission Line

Tannu Sharma (tannu.sharma@utah.edu)
Electrical and Computer Engineering 6740, Spring 2014
University of Utah

Abstract

We present here a routing tool developed to route transmission lines. The main purpose of the project is to identify the candidate nets to route as transmission line and report the effects of this selection on the overall quality of the routing. To validate this our tool performs a planar routing for our transmission line by avoiding cross-over nets and follows heuristics to minimize bends while maintaining the optimum usage of area in the design. We report the changes seen in the quality of route based on various heuristics applied. These results aptly demonstrate the trade offs that we see in a routing problem. We try to tune different strategies and see their effects. In future, the extension of this project may include a methodology to minimize Z-shapes and using L-shapes in routing the transmission line. We expect to improve the tool to add multi layer routing.

1 Introduction

Power dissipation in the interconnect network has become a bottleneck in modern System-on-Chip (SoC) design. This is especially true in the case of global wires, which do not scale with technology. Shrinking wire dimensions increase the resistance of the wires, which subsequently [x.] increases the IR drop along the wires. To compensate for this signal degradation, repeaters are added at regular intervals, thereby increasing the silicon area and energy. High data rates necessitate pipeline stages on a long wire to compensate for wire delay while maintaining desired throughput. This also contributes to the added energy cost of data communication on long wires. Designers have begun to look for viable alternatives to pipelined RC wires for use as global interconnects. Transmission lines are an interesting medium that provide near speed-of-light signaling for long distances without the use of repeaters.

A digital circuit designer might want to predefine the set of net which needs to be routed as a transmission line. However, for an analog circuit, a designer might want to change a simple wire to a transmission line after analyzing the signal interference or for drift compensation. *In this project we are trying to evaluate if long wires and high power signals can be replaced with a transmission line on an IC.*

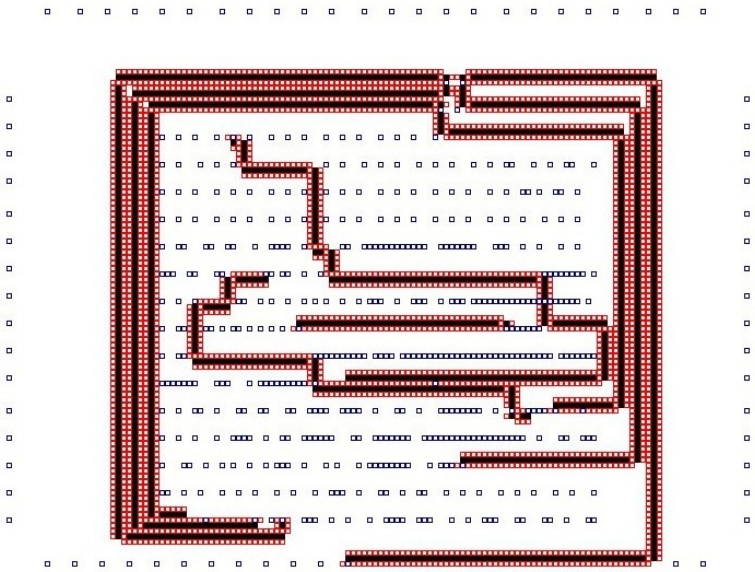

Figure 1: Planar routing for Transmission Lines on a SOC, with minimum bends and pin drops coupled with GND traces.

Transmission Line or TL is an interconnect used to guide electro-magnetic waves from one point to another with least possible loss and distortions. It is implemented by placing two co-planar wires next to each other. TL is mainly used in RF circuits or in high speed digital transmission. A normal electric wire creates too high a loss and distortion to the signal whereas, if a TL is used the signal will be transmitted with almost zero loss.

To exploit the benefits of a TL, special care is needed during its routing. Signal line isolation is desired to prevent unintended coupling of a TL.

a. ***RF transmission line*** shouldn't be routed in close proximity to each other for extended distance. Coupling between micro-strip lines increases with decreasing separation and increasing parallel routing distance. So, co-planar ground waveguides are used for isolation between lines or a ground layer between parallel running layers with transmission lines.

b. ***High-speed digital signal lines*** should be routed on a different layer than a RF signal lines to prevent coupling.

c. ***VCC and power lines*** need to be routed on a dedicated layer. These lines should also be separated from any RF lines that will transmit large amount of RF power.

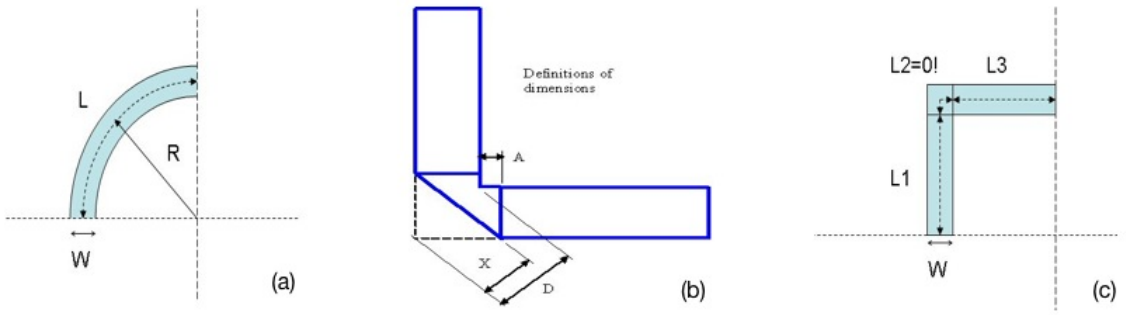

Figure 2: (a) When curving radius greater than the three times the TL line width is used, no significant changes in impedance characteristics are observed. (b) Mitered bends restore the original characteristic impedance by removing the capacitance introduced by orthogonal bends in Fig (c). [ix.]

Since, we want our transmission line to be routed only on a single layer, so affect of layer changes in a TL will not be discussed. To optimize the single layer TL routing, the project suggests a methodology for bend compensation in planar routing.

Bends and corner compensation: When a transmission line is required to bend or change direction due to routing constraints (refer Figure 2), we need to use the *bend radius that is at least three times the center conductor width.* This is required to avoid the characteristic impedance change moving through the bend. In an IC, due to manufacturing limitations, curved bends or angled miter are not feasible solutions. So, we will assume that TL will undergo a right-angled bend adding impedance discontinuity caused by local increase in effective line width going through the bend. To avoid that, we will work on a problem to minimize bends while routing a transmission line.

Here, we will introduce a methodology to select those nets which could be probable candidates to route as transmission line over an integrated circuit. The focus of our selection will remain on the longer nets. So, our problem is divided into following parts:

i. Identify the nets which can be replaced with a transmission line.

ii. Design a router which can route planar nets as a transmission line, where three nets will run parallel to each other, and the outer GND wires shield the TL in the center from coupling effects.

iii. We will do the cost-benefit comparison for adding a transmission line with the area occupied by a TL and the throughput improved and energy saved.

2 Overview

Almost all global routing algorithms look to minimize wirelength and via count. Techniques to optimize secondary objectives usually build on this initial solution. There have been efforts to introduce a pareto-front based framework [xii.]to minimize signal power by re-routing highly congested nets to increase wire spacing, with an acceptable wirelength increase. An extension to this work introduces routing congestion as an added optimizable parameter[xiii.].

In this project, I investigated some strategies to route a transmission line on an IC. I also studied the cost-benefit trade-off of adding transmission lines on a design. We will obtain the placement of our modules using Capo MetaPlacer in the bookshelf formal, which is an input for our router. At this moment, Bookshelf format is the only recognizable format for our router.

2.1 Global Selection

In this step, we calculate the manhattan length of each net to find the longest nets in our design. For multi pin nets a round trip length is calculated in an arbitrary order. We have to select longer nets with less number of pin drops. Post selection all such nets will be marked as TL.

2.2 Net Ordering

Routing multiple nets by greedily optimizing one net at a time may produce inferior configurations with unnecessarily large numbers of routing failures (refer Figure 3) and total wire length (refer Figure 4). Also, multi-pin nets increase the complexity of net ordering when decomposed into two-pin sub-nets. So, it is important to define the routing priority of a net so as to minimize the number of bends in a transmission line, and have shorter routes for a design.

2.3 Crossings

Two transmission lines cannot cross each other as they carry signals at different frequencies. They are properly shielded so as to avoid the coupling effects to other signals in the design. So, our algorithm for planar routing will ensure that no two nets TL or non-TL wire cross each other.

2.4 Planar Routing

Planar routing for a TL net is done using our custom router. At this moment, all the nets are routed as planar routes, however, in future we do see the capability to incorporate

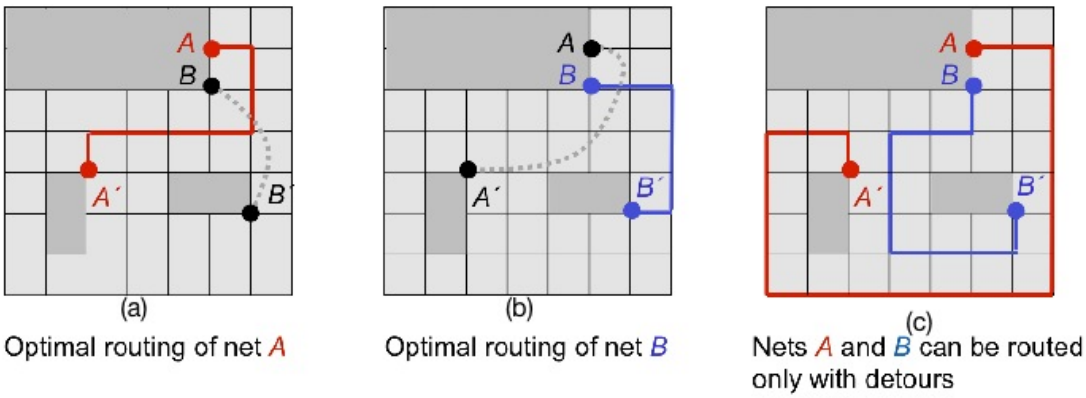

Figure 3: Effect of net-ordering on rout-ability of a design. (a) Optimal routing of net A prevents net B from being routed. (b) optimal routing of net B, prevents net A from being routed. (c) Nets A and B can only be routed if each uses more than minimum wire length.

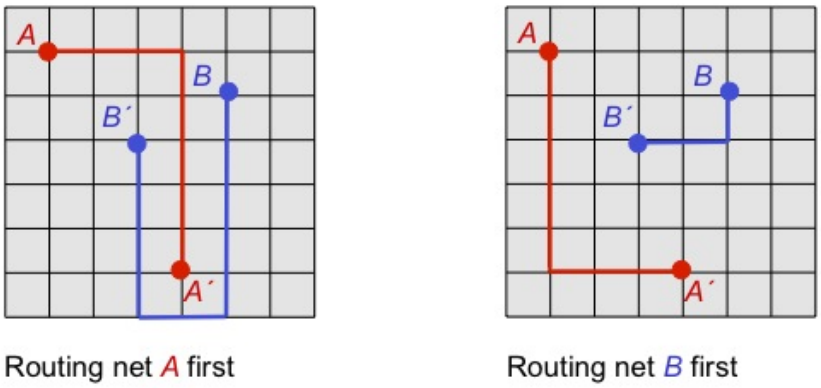

Figure 4: Effect of net-ordering on Total wire length. Total wire length will increase if we route net A before net B.

the changes in the router to consider planar routing only for transmission lines and layer changes for other non-TL nets.

2.5 Cost-Benefit Trade-off

i. A planar transmission line is sandwiched between GND traces for shielding purpose. This increases the area consumption by a TL in comparison to a non-TL trace. However, routing it as a TL removes the need of buffers or repeaters in the design which saves the placement area. The routing area occupied by a TL is a cost to a design.
At this moment, my test case include an analysis for replacing a simple trace with a TL. However, as a long term scope of the project, we might want to study a design where we replace bus with TL, which might translate the routing cost into an added benefit introduced.

ii. A transmission line, supports high data rate, leveraged as bus properties over a TL with low power consumption. i.e. TL improves the throughput for a design with the low energy consumption.

In this project, we will analyze the area consumed by a TL as a cost to the system using our custom router.

3 Proposed Methodology

At this moment, we have two different techniques which could be helpful to route a transmission line. We implement one of the methodologies. Let us first understand the reason for choosing planar routing for a transmission line.

i. Transmission lines are the co-planar lines running between GND parallel shields on its sides.

ii. TL are replaced for long traces so as to avoid the need of repeaters for transmitting the signal.

iii. Any orthogonal bends, vias or layer changes will make the signal prone to reflections. (To avoid orthogonal bends in a TL we might want to study the feasibility of miters or curved traces in an IC.)

iv. More the congestion in the routing area, more will be the bends in a TL, since they are routed for long routes.

v. Sometimes, taking a route to a different layer helps in minimizing the number of bends because of increased routing area

While proposing a routing solution for a TL, we have worked on the above constraints.

3.1 Custom Router for Planar Routing

a. Maze router is implemented using Dijkstra's algorithm to find the shortest path between two nodes. We will break a multi-pin net into a two pins and then search for the closest path between two nodes. We have saved the estimated wire length (EWL) in *.ewl and routed wire length (RWL) in *.rwl file for each net.

b. The values in .ewl and .rwl will help us to make a selection for the probable TL nets based on the wire length and pin drops for each net.

c. Higher weight is added on a net which will be routed as a TL.

d. The net-ordering algorithm is used to prioritize the routing of a TL. All the nets to be routed as a transmission line will be ordered first, followed by other nets.

e. TL routing will occupy three grids (one for a TL and two for parallel gnd traces) whereas a non-TL route will pass through a single grid. Hence, more routing area is occupied by a TL, shoving-off the non-TL nets from there path in non-TL routing.
This might result in un-routable nets because of the unavailability of the routing area.

4 Experimental Apparatus

There are several policies in the experimental setup provided with the implementation. So, in the following sub-sections we will discuss the implementation strategy for my project.

4.1 Format of Experimental Design

My test design is in the Bookshelf format which is default format for Capo - the MetaPlacer. The TESTS folder under Capo installation has demo designs in Bookshelf format. Run the example design in Capo, to obtain the optimal placement and then route the design using Custom router.
Note: Save the Capo placer results in bookshelf format.
Apart from output files from Capo, we need two more files *.tl and *.conf.

a. ***.tl:*** file has a list of nets which needs to be routed as a transmission line. Example -
net_name1
net_name2

b. ***.conf:*** is a configuration file which has the scaling details for the Pygame Window. User can configure the window by modifying the file as per the design need. Example -
MapSize_x=80 # x dimension of the chip area
MapSize_y=80 # y dimension of the chip area
scaling=10 # scaling 10 means pixel is 10 for display.

Wstraight = 2 # cost of an edge
spaceScale=5 # space scaled the design with this value to insert white spaces for routing area
WsameNet=1 # if there is a bend, we add bend penalty and reduce WsameNet
WBendPenalty=1 # to calculate bend penalty, added in Wstraight.
inversion =0
screenX=1000 # horizontal viewer size of pygame.
screenY=800 # vertical viewer size of pygame.
negCompX=0 # if we have negative x value for object location.
negCompY=0 # if we have negative y value for object location.
resolutionFact=1 # resolution of the design.

4.2 Global Placement

To start with, we have a netlist along with the randomized placement of objects in and around the placement grid. We need to place the objects optimally so as to have minimum net length for our placement.
Cost function: For a graph G(V, E), where V is the set of vertices which are modules or objects and E is the set of edges defined by connectivity of various objects i.e. nets. Then, to have optimum placement, we need to minimize the total net length and the length for each net.
Subject to constraints: For a graph G(V, E), the placement of the module should obey its object orientation and fixed status while maintaining the minimum distance between the placed objects. The routing should be done in such a way it should minimize the total number of bends in the design
Legalization: Constraints added to avoid overlapping between objects:

i. No overlap exists between modules, legalized by Capo.

ii. No two nets (TL or non-TL) can be routed from the same grid location.

iii. Minimum spacing constraints between two nets (TL or non-TL) is obeyed everywhere.

4.2.1 Capo

For placing cells across an IC, I have used Capo, the MetaPlacer, developed by the University of Michigan. Capo is just one of the feature included in the MetaPlacer. It does placement in two steps:-

a. ***Global Placement:*** where it does cell placement using recursive min-cut based bi-partitioning scheme of hmetis tool (which is a multi-level Fidducia Mattheyses partitioner).

b. ***Detailed Placement:*** is done using branch and bound partitioner.

The size of the partition is adjusted to accommodate excess circuit modules in one partition. There are three different partitioners implemented within Capo:

i. *Optimal:* partitioner is implemented using branch and bound method. Bins with seven or less cells are processed with this one.

ii. *Middle range:* using Fidducia Mattheyses.

iii. *Large scale:* using multi-level Fidducia Mattheyses.

4.2.2 Bookshelf Format

Bookshelf has slots which contains entries. It contains:

i. *Reference Implementations* - to ensure progress in algorithm research, especially when heuristics are involved.

ii. *Standardized data representations* - to enable easy comparisons among referenced implementations.

iii. *Standard experimentation evaluation techniques* - to support collaborative research.

Below is the brief information on various files for placement instance and then reused for routing:

a. ***.aux:*** It contains a set of input files referenced together. Example -
RowBasedPLacement : demo1.nodes demo1.nets demo1.wts demo1.pl demo1.scl

b. ***.nodes:*** It has the width height information for each object along with the keyword 'terminal' if the modules are fixed. Example -
object_name width height terminal

c. ***.nets:*** It has a list of nets, with input and output pin connectivity for each net. It also has pin offset information, by default it is (0, 0). Example -
NetDegree : number net_name
object_name I : x_offset y_offset
object_name O : x_offset y_offset
Pin offset is measured from the center of corresponding object.

d. ***.pl:*** This file specifies the placement of the object. It contains (x, y) location, orientation (N, S, E, W, FN, FS, FE, FW) and fixed status (by default - unfixed) for each object. Example -
object_name1 x_location y_location : orientation /FIXED
object_name2 x_location y_location : orientation
lines are comments with /FIXED is added for fixed objects.

e. ***.scl:*** It has the circuit row information. If it is horizontal, y-coordinate is mentioned, if vertical, and x-coordinate is specified. Example -
CoreRow Horizontal
Coordinate : y_coordinate # row starts at coordinate _coordinate
 # (y-coordinate if horizontal, x-coordinate if vertical row)
Height : y_value # row height is y_value
Sitewidth : 1
Sitespacing : 1 # distance between the left edges of two adjacent sites
Siteorient : 1
Sitesymmetry : 1

f. ***.wts:*** This file contains weights assigned on objects and nets, useful during placement and routing. Example -
object_name weight

The Bookshelf output from Capo will be used as an input for our Custom Router.

4.3 Custom Router

We have designed a custom planar route solution using Dijkstra's Algorithm. The custom router is based on Maze router, additionally, it implements net ordering, transmission line routing and bend minimization. Router reads the Capo results and scales (space scale parameters in *design-name.conf file*)them to insert white spaces for routing between the modules. The overall functioning of the router can be explained in different routing stages discussed below:

4.3.1 Global Selection

TL selection algorithm is executed on the results obtained from the Capo. Our custom router lets the layout designer to select the probable nets which can be routed as a transmission line. To help make this choice we provide a list of nets with estimated length and number of pins *design-name.ewl*, sorted by length. Concisely, following steps are done in Global selection:

i. We obtain a set of nets with long trace length in the *design-name.ewl* file.

ii. Designer will make a choice by listing the nets to be routed as TL in the *design-name.tl* file.

iii. If no net is specified, all the nets will be routed as a simple trace on a single layer.

Once the TL nets are specified, these nets will be assigned high net weight for differentiating them with a non-TL net.

4.3.2 Net Ordering

The choice of net and pin ordering is dependent on the type of routing algorithm. Pin ordering can be optimized using -

i. Steiner tree based algorithm where we optimize multi-pin nets into two pin nets.

ii. Geometric criteria where ordering is done based on the (x, y) locations of a pin.

There three rules which need to be obeyed:

Rule 1: For two nets A and B, if aspect ratio of net A is greater than the aspect ratio of net B, then, net A is routed first (refer Figure 5).

Rule 2: For two nets A and B, if pins of net A are contained with the bounding box of net B, then net a is routed first. As shown in Figure 6 There are two potential net orderings D-A-C-B or D-C-A-B.

Rule 3: Let X(net) be the number of pins within bounding box of net *net*. For two nets A and B, if X(A) is less than X(B), then A is routed before B. For each net, consider the pins of other nets within its bounding box, then the net with the smallest number of such pins is routed first. Ties are broken based on the number of pins that are contained within the bounding box and on its edge. As shown in Figure 7, net D is routed first because it contains no pins within its bounding box, then net C is routed because it contains one pin, next net E and later net B and net A.

4.3.3 Global Routing : Bend Minimization, length minimization

As stated earlier we implement the path finding through a variation of Djikstras. We have particularly changed the heuristics for giving edge wrights for a graph in the context of Djikstras. The Algorithm views the grid as an unexplored graph. At each node (grid cell), there are only 4 possible edges, in 4 cardinal directions. We assign weight to each such edge based on the context. At this point we assign higher weight to bends, lower weights to occupied grid cell of the same net. This forces djikstras to avoid bends unless absolutely necessary.

In case of multi pin nets , this heuristic makes our algorithm chose a shortest path along the already routed path of the same net. This heuristic is relaxed for Transmission lines. Further for multi pin net, one by one in pair 2 pins are connected through this shortest path algorithm. Prior to this a smart pin ordering can help reduce wire length.

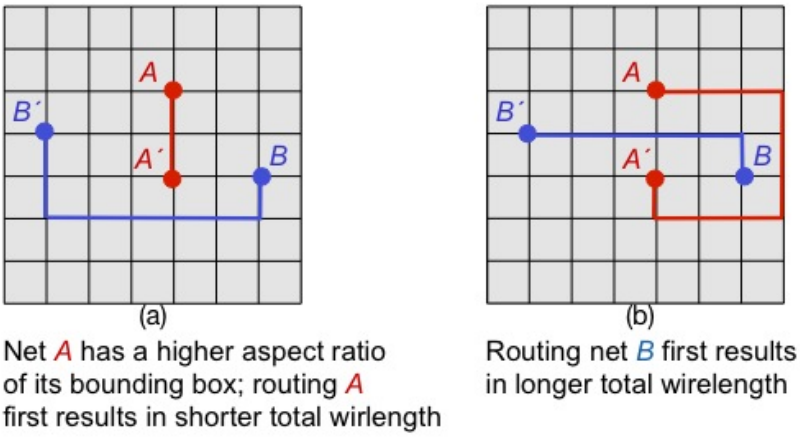

Figure 5: Net ordering based on the aspect ratio of the net bounding boxes. (a) Net A has higher aspect ratio and results in shorter total wire length, (b) whereas routing Net B results in greater total wire length.

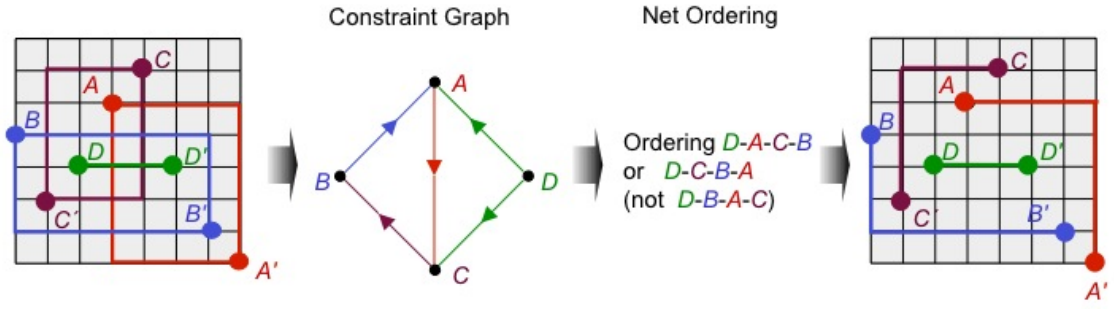

Figure 6: Net ordering based on the pin locations inside the bounding boxes of the nets.

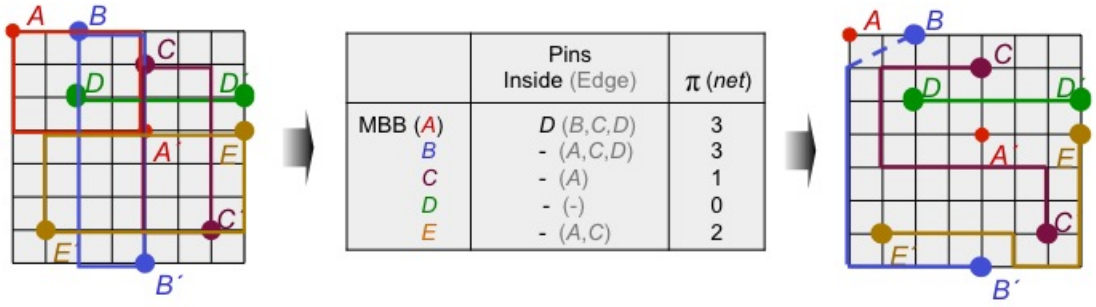

Figure 7: Finding the net ordering scheme based on the number of pins of nets within the bounding box.

5 Results

While selecting a net as Transmission Lines, we have ensured that they have long nets and less number of pin drops. The information can be obtained from *design-name.ewl* file generated by the router.

5.1 Placement

Capo does the placement and the updated results saved in *demo1.pl* file are used. Our code just insert the white spaces for routing area. Figure 8 shows the gnuplot of the placement obtained from Capo for *design1*.

5.2 Global Routing using Custom Router

Routing results (in Figure 9) when no transmission lines are placed for the design (**design2** is used).
Post - routing results in *demo2.rwl* file:
Total Length of Nets after Routing = 3737
Total Area under Routing = 3737
Total Transmission line Length of Nets after Routing = 0
Total Area under Transmission line after Routing = 0
Total Bends after Routing = 779
Routing complete by: 19.2904656319%

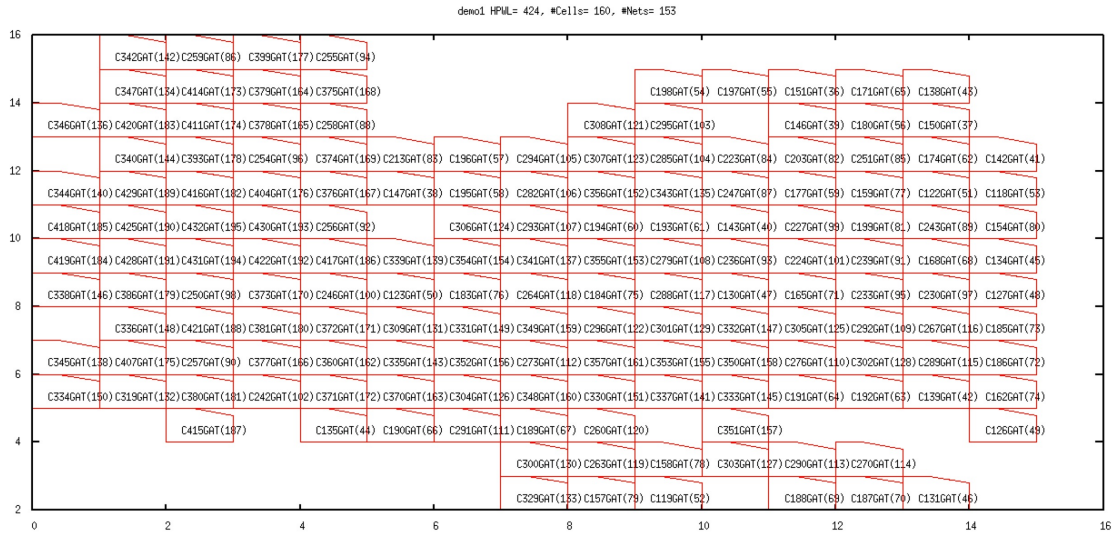

Figure 8: Gnuplot for the placed objects for design1 obtained from Capo.

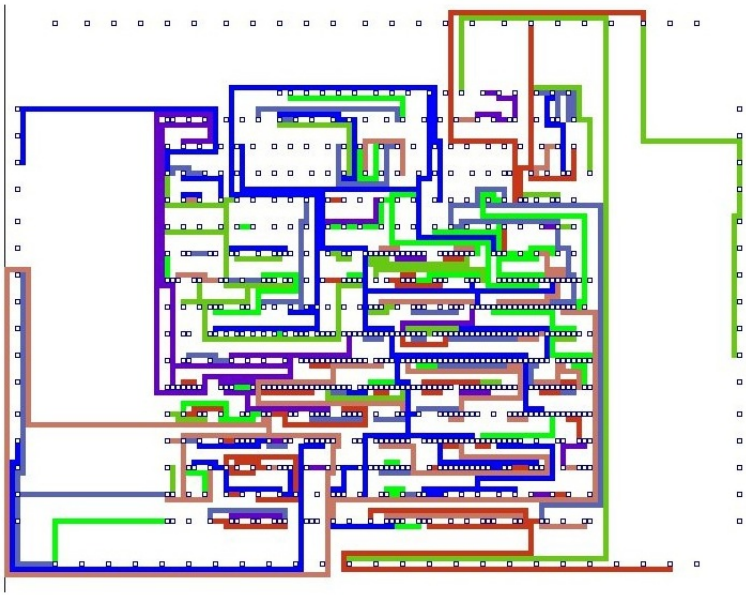

Figure 9: Routing results when no transmission lines are present for design2.

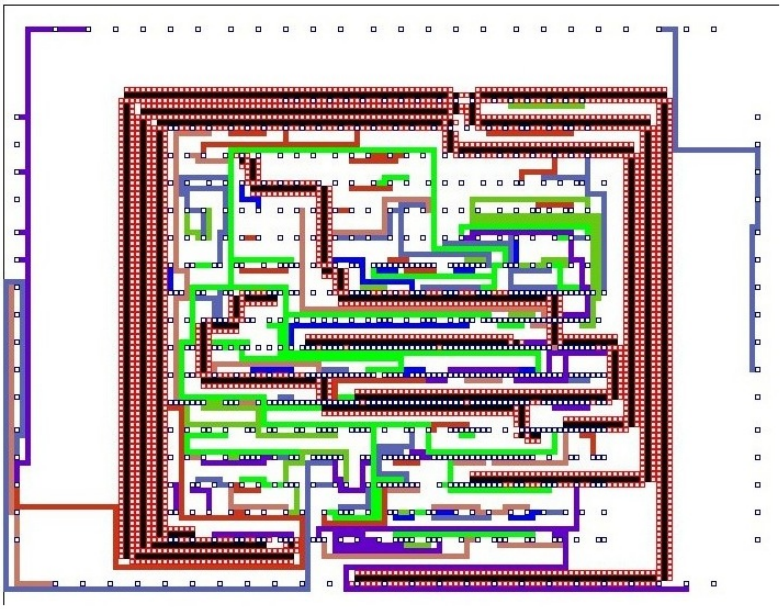

Figure 10: Routing results when transmission lines are present for design2.

5.3 TL routing using Custom Router

Routing results (in Figure 10) when the nets with long trace length and less number of pin drops are routed as a transmission line (**design2** is used). Routing results from *demo2.rwl* file:
Total Length of Nets after Routing = 6144
Total Area under Routing = 6144
Total Transmission line Length of Nets after Routing = 1179
Total Area under Transmission line after Routing = 1179
Total Bends after Routing 720
Routing complete by: 14.6341463415 %

6 Conclusion

The number of un-routed nets is increased after routing only six nets as transmission line. Also, the total wire length for non-TL nets is increased post-TL routing. So, whether it is acceptable to route longer nets as transmission lines will depend on the throughput benefit obtained from TL routing on a design.

7 References [14]

i. Yue Xu, Yanheng Zhang and Chris Chu, Iowa State University, FastRoute 4.0: Global Router with Efficient Via Minimization, IEEE

ii. Ali M. Farhangi and Asim J. Al-Khalili, Concordia University, On the bend minimization on Clock tree networks, IEEE

iii. R.J.P. Douville and D.S. James, Experimental Study of Symmetric Microstrip Bends and Their Compensationf, IEEE

iv. Majid Sarrafzadeh, Kuo-Feng Liao, and C. K. Wong, Single Layer Global Routing, IEEE

v. Jarrod A. Roy, Saurabh N. Adya, David A. Papa and Igor L. Markov, University of Michigan, Min Cut Floor Placement, IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems

vi. Christopher Condrat, Priyank Kalla and Steve Blair, University of Utah, Channel Routing for Integrated Optics, IEEE

vii. Aaron Carpenter, Jianyun Hu, Ovunc Kocabas, Michael Huang, and Hui Wu, Enhancing Effective Throughput for Transmission Line-Based Bus, IEEE

viii. Huang-Yu Chen, and Yao-Wen Chang, Global and detailed routing, National Taiwan University

ix. Bends in a transmission line, Microwave101.com

x. www.itrs.net

xi. Aaron Carpenter, Jianyun Hu, Ovunc Kocabus, Michael Huang, and Hui Wu, Enhancing Effective Throughput for Transmission Line based bus, IEEE

xii. Hamid Shojaei, Azadeh Davoodi, Twan Basten: Collaborative Multiobjective Global Routing. IEEE

xiii. Hamid Shojaei, Tai-Hsuan Wu, Azadeh Davoodi, Twan Basten: A pareto-algebraic framework for signal power optimization in global routing. ISLPED

xiv. Andrew B. Kahng, Jens Lienig, Igor L. Markov and Jin Hu, VLSI Physical Design from Graph Partitioning to Timing Closure