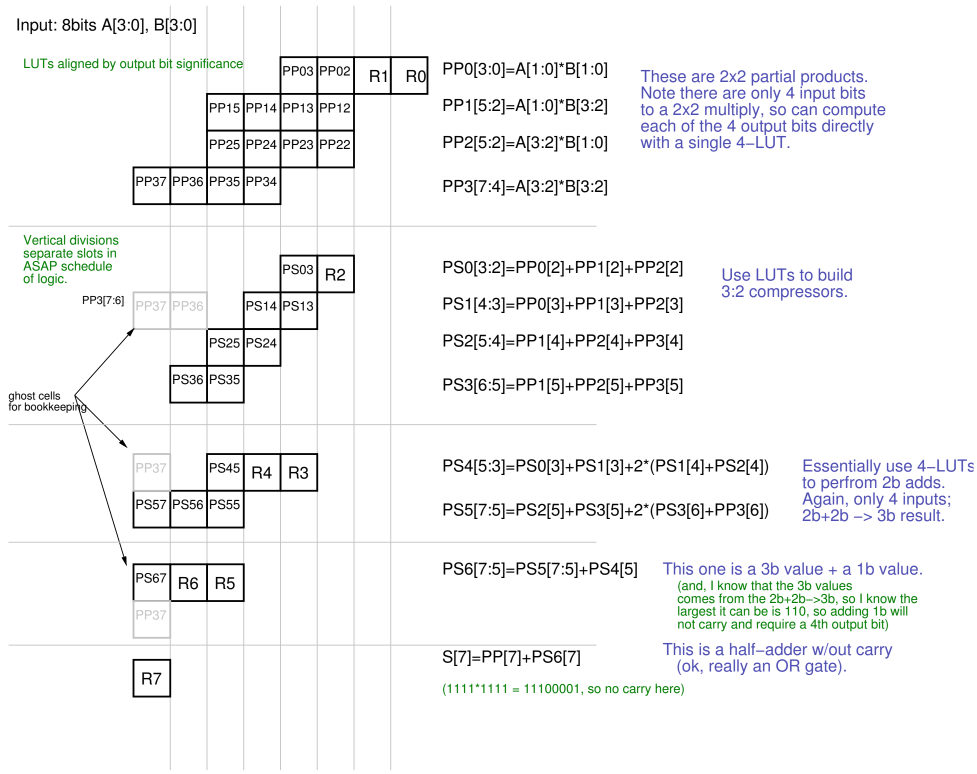
University of Pennsylvania
Department of Electrical and System Engineering
Computer Organization

ESE680-002, Spring 2007 **Assignment 5: Wire Requirements** Monday, February 26

Due: Wednesday, March 14, 12:00PM

For this assignment, we will compute wiring requirements for the 4-LUT based 4×4 multiplier shown below. The total design has 34 4-LUTs (shaded boxes just shown to help illustrate design flow...they might be pipeline registers in a fully pipelined design).



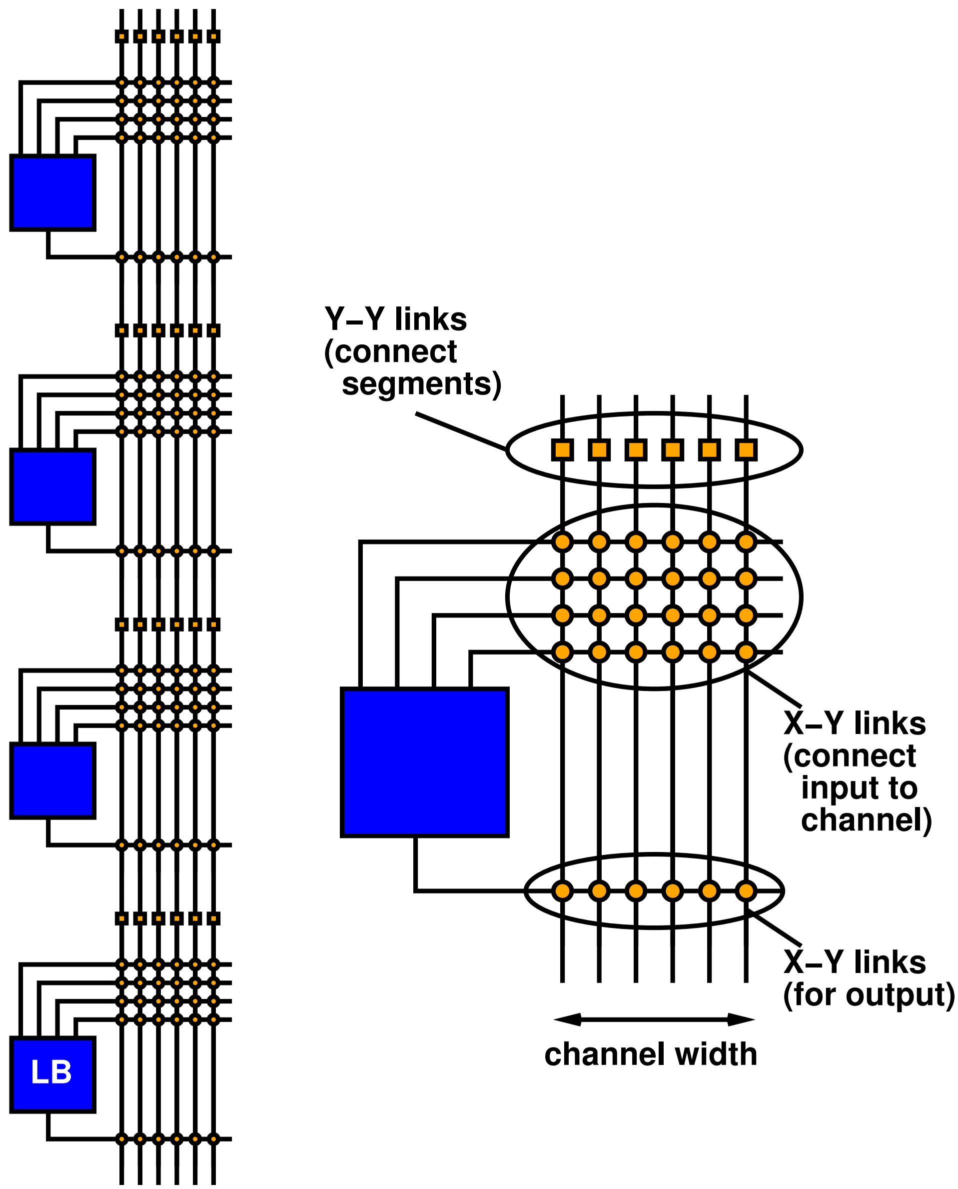
1. Consider using a crossbar for the interconnection of your 4-LUT logic blocks. Assume your primitive substrate has exactly the right number of 4-LUT logic blocks, inputs, and outputs to support your design.

   - The inputs to the crossbar will include the outputs from the LUTs and the primary inputs to the application.
   - The outputs from the crossbar will include the inputs to the LUTs and the primary inputs to the application.

   (a) Identify the size of the crossbar (total inputs × outputs, justify your numbers).

   (b) How many total switches does the crossbar require?

2. Consider a 1D-segmented bus configuration. Again, assume the primitive substrate has exactly the right number of logic blocks to support your design. The external input stream enters on one end of the bus (let's say the top), and your output should exit the opposite end (bottom).



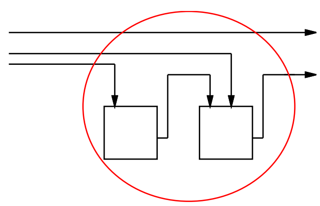
(a) Layout your components in 1D to minimize the number of channels needed. (show your layout and routing in an appropriate, unambiguous manner; it should be possible to easily see the span of each wire segment in your diagram.)

(b) What is the minimum channel width needed to support your design?

(c) How many total switches does this minimum size segmented bus require? (*n.b.* please count both X-Y (C-box) connections and Y-Y (segmentation) links).

(d) Based on your 1D layout, recursively bisect the design into successively smaller groups of contiguous logic blocks and note the total number of IOs into each of the groups.

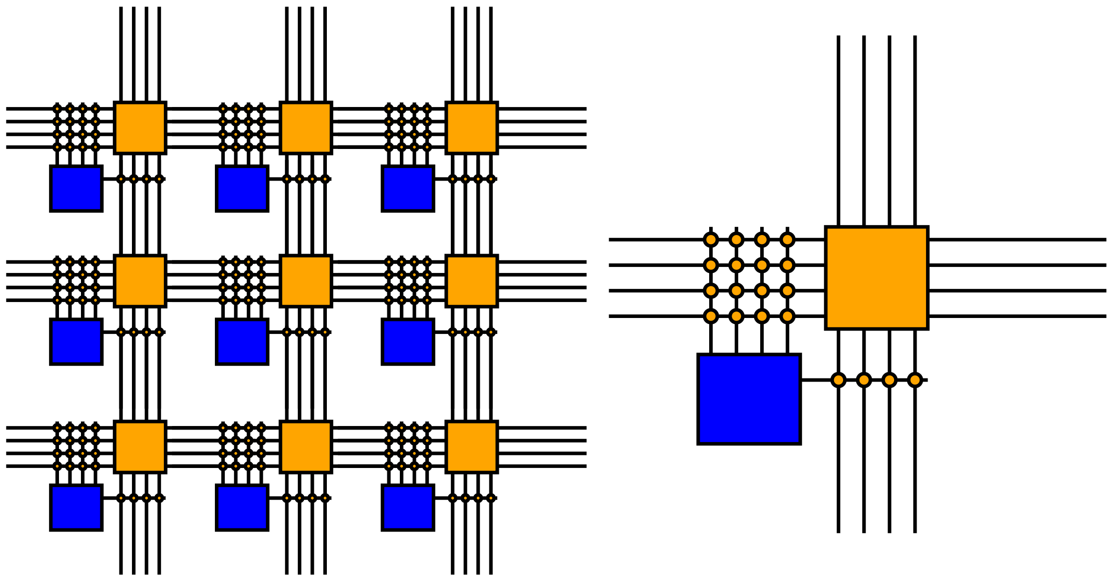
- *i.e.* Since the design has 34 logic blocks, then you have one cluster of size 34 with the IO from the overall design. You will have two clusters of size 17, then 4 clusters of size 8 or 9 (2 of each), then 8 clusters of size 4 or 5, then 16 clusters of size 2 or 3, bisecting the 3's gives another 2 of size 2, and finally 34 clusters of size 1.
- Count the IO of the cluster as the total signals produced outside of the cluster which must be used as inputs to the blocks in the cluster and all the outputs produced in the cluster which must be used outside of the cluster. So, the circled cluster below has 3 IOs. The one block output produced and consumed in the cluster is not included in the cluster IO, nor is the routed wire which passes these blocks but is neither produced not consumed by blocks in the cluster.



Collect the full set of partion size and IO data. Based on this, calculate Rent's rule parameters for your design. (based on the maximum number of IOs for a given partition size? based on the average?)

3. Consider a 2D-array configuration.
   The array is the smallest rectangular region into which your design will fit (*i.e.* 6×6).

   Assume the switchbox at the intersection of channels provides full connectivity to the adjacent channels. That is, it is made up of 4 crossbars each with $3w$ inputs and w outputs ($6w^2$ switches if each switch is bidirectional); where w is the channel width.



   (a) Place and Route your design to minimize channel width. (I want a nice picture here which clearly denotes the assignment of each block and shows each routed link.)

   (b) What is the maximum number of routes which share a channel in your routed design? (This is the minimum channel width for which your design is routable.)

   (c) At this minimum channel width, how many switches does your design require?

   (d) Consider depopulating the switchbox so that it only holds $6w$ switches (each wire entering the switchbox may go straight, turn left, or turn right...but only connects to a single channel on each of the three adjacent sides). You are **not** required to reroute your design.

      i. Assuming that the design could still be routed in the channel width identified above, how many switches does the design require?

      ii. How much of a channel expansion could your design experience before the depopulated design required more switches than the fully populated design? (Answer here is a factor. *e.g.* Perhaps my original fit into 4 channels and it would take over 19 channels for the linear population switch count to exceed the full population count. Then I could tolerate a channel expansion factor of 4.75 and still have the linear population scheme require less switches than the full population scheme.)