This is the approach we implemented:

* 1. Firstly we will assign 1 MAC unit to each of the layers which will set P1, P2 and P3 to 1.
  2. The input values for each layer are identified and factorized. The factorized values are stored in respective arrays in increasing order. The array contains the number (input size M3) also.
  3. Starting from layer3 we compared values of ‘L - 3’ with each of the factors from highest to lowest.
  4. If (L-3) is greater than any one of the factors (say M3\_3), that ‘factor – 1’ is added to P3 and the loop exits. The last step before exiting the loop is decreasing (L-3) with (M3\_3 – 1).
  5. The above 2 steps are repeated for layer2 and layer1 as well.

This is the approach we wanted to implement:

1. Firstly we will assign 1 MAC unit to each of the layers which will set P1, P2 and P3 to 1.
2. The input values for each layer are identified and factorized. The factorized values are stored in respective arrays in increasing order. The array contains the number (input size M3) also.
3. Then, the layer with most number of inputs (say M1) is considered. If ‘L-3’ is greater than the inputs to that layer, ‘M1-1’ MAC units are assigned to that layer. If L-3 is not greater than M1, ‘L-3’ is compared to the 2nd greatest input size (say M2). If L-3 is greater than M2, M2-1 MACs from L-3 are assigned to that layer. Otherwise the next greatest input size (M3) is checked.
4. If ‘L-3’ is lesser than all input sizes, then highest factor corresponding to each input value is compared to L-3 from factors of M1 to M3. If ’L-3’ is lesser than all 3 ‘highest factors’ then do the same for 2nd highest factors. If for any of the values of the factors, L-3 is greater, assign the number of MACs equal to the value of the factors, to that layer. And consider only the other two layers from now on. Remove the value from L-3.
5. Do the above step until all the factors for the input sizes >= 1 are examined.

The drawback for both the approaches is that there can be a MAC that is left out from being used. In that case we might end up wasting MAC units.

1. There are 4 signals that control the operation of every layer. S\_valid, m\_ready, s\_ready, m\_valid

To design a network of multiple layers, every pair of consecutive layers act as master – slave system.

* 1. The master sends data using m\_valid as a control signal which tells the slave that there is input data available. If the slave is ready it will assert an s\_ready signal and receive data.
  2. Once data is processed ‘slave’ in the above situation acts as master to the next layer in the network and asserts an s\_valid signal and waits for m\_ready which act as m\_valid and s\_ready signals for the adjacent layer.
  3. Data\_out one layers becomes data\_in for the next layer in the network.
  4. In our design, we implement the same logic but only for three layers. This logic has to be implemented for M layers.

The challenges that we might face are

1. The data out of the network has to wait until m\_valid for the last layer is asserted. Until then there won’t be any valid output.
2. The output size of each layer has to be equal to the number of columns in the W vector of the next layer.