Ömer Faruk SERT

&

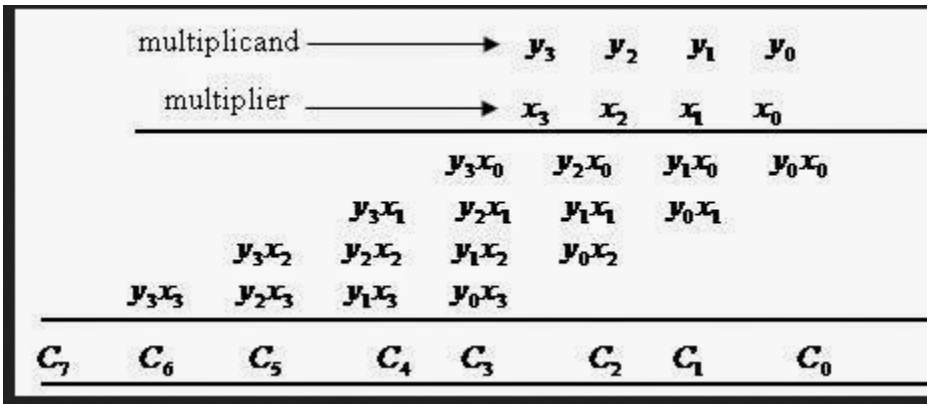
İbrahim Samed YALÇIN

040170244

040180055

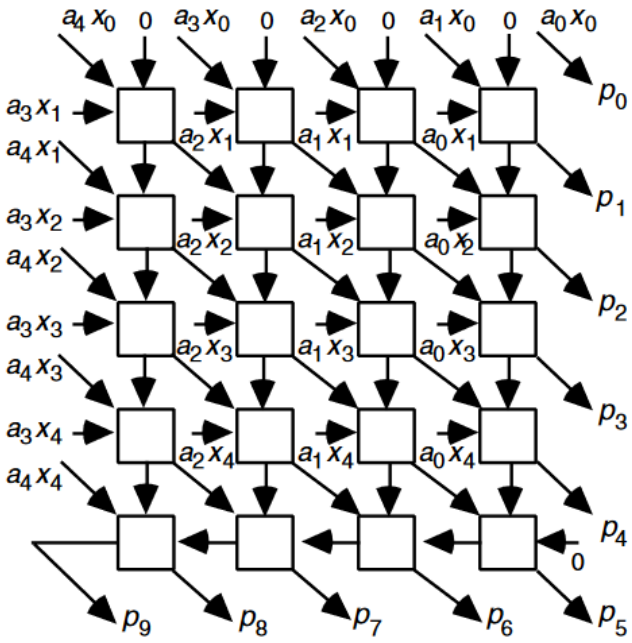
Array Multiplier

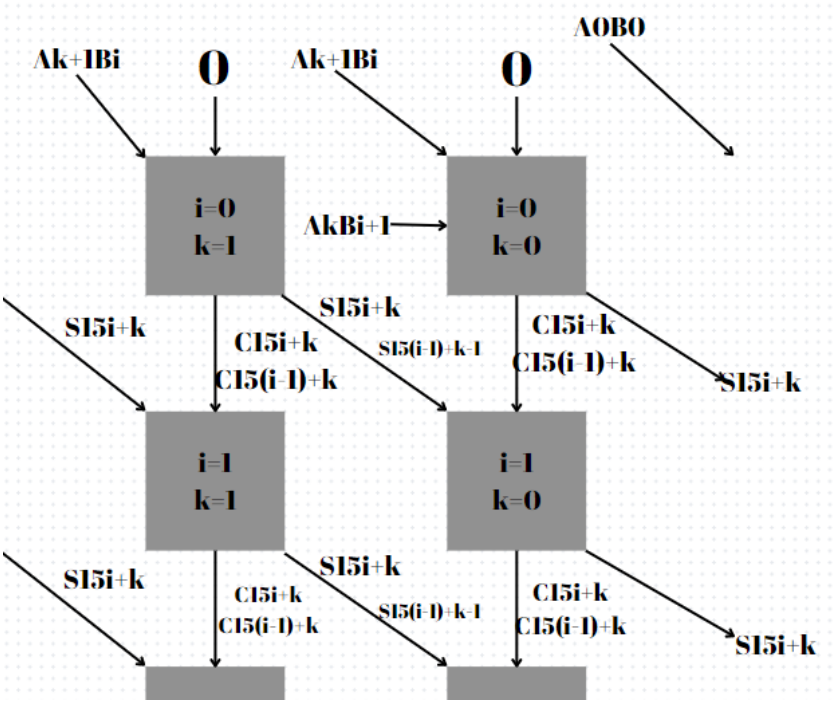
How Algorithm Works



As shown in the figure multiplication contains and gates and addition that's why I used carry save adder which do not pass the carry to next FA block, so we can connect carry to any block that is convenient.

Design Steps





As shown in the figure I gave row and column numbers to the FAs.

One of the inputs of the FA is i=i-1 and k=k+1th FA's sum output,

Other input is the carry output of the block i=i-1 and k=k, final input is connecting to and gate the kth bit of A and the i+1th bit of B connecting the output to the FA. There are special cases at the boundaries.

For i=0th row one of the FAs one of the inputs is 0 because there is no prior FA to product carry. Initial inputs are determined in this row and formula of them is connecting k+1th bit of A and ith bit of B to an and gate and connecting output of the and to the input of the FA, this formula exists also for k=14 and i is larger then 0 and less then 16 (16 is not included) but only one difference which is there is no zero input because there is prior FAs so carry of the i-1th and kth FA is connected to the FA instead of zero input.

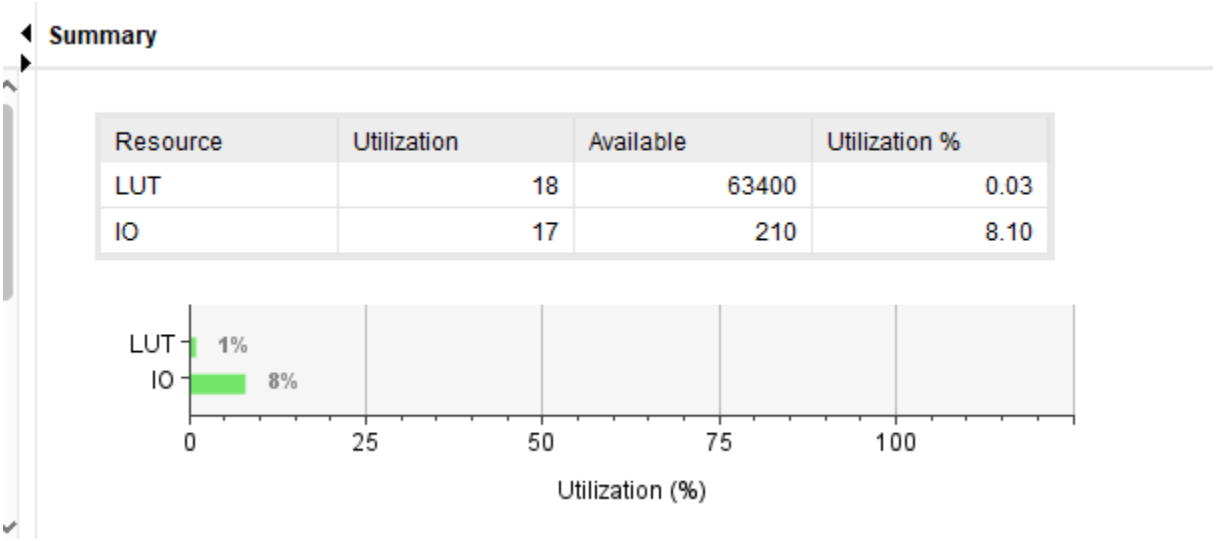
For i=15 row there is also a special case which is i=15 and k=0 FA has zero input and carry output connected to i=15 and k=k+1th FA because there is no more FA for i=i+1 and carry affects k=k+1th FA as in the adding algorithm final carry affects the bit after the present bit.

For the result part formula is P[i+k+1] = S[15xi+k] because we need end of the diagonals and our matrix like FAs has 15 column for every line, that's why there is 15xi in the formula to skip whole column.

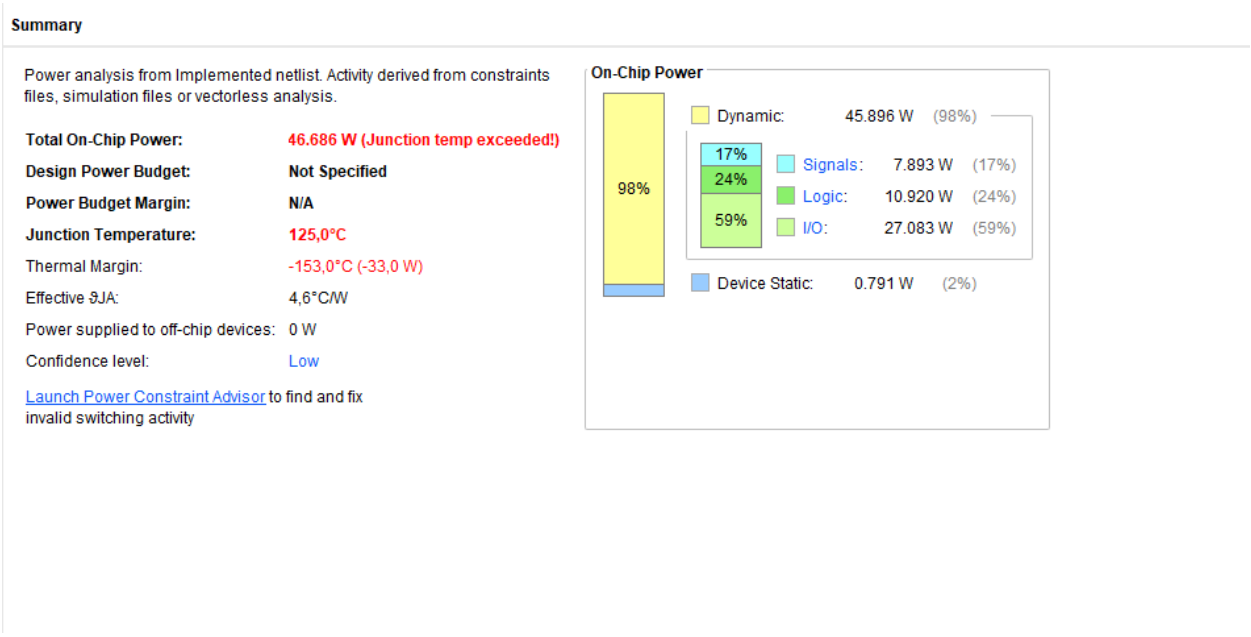
There is two special case first one is P[0] can not produced from formula because i and k starts from zero that's why P[0] can not be produced and also p[0] do not come from any FAs, it is coming from connecting 0^{th} bit of A and B to an and gate and output of the gate is P[0], second special case is for P[31] because it is not coming from sum output of any FA, it does come from last FAs carry output because there is no more FA to connect the carry, that's why P[31]=Cs[239]

Arrays naming comes from Verilog code that I wrote.

Total LUT Usage



Average Power Consumption



Maximum Clock Frequency

B[0]	P[31]	44.160	SLOW	5.766	FAST
A[14]	P[31]	44.056	SLOW	3.669	FAST
A[13]	P[31]	43.649	SLOW	3.547	FAST
B[1]	P[31]	43.643	SLOW	6.047	FAST

Maximum combinational delay is 44.16 ns and maximum clock frequency is 1/T, so it is roughly 27.73MHz

My Intuition about maximum delay

As shown in the second figure left most and upper most FAs inputs to bottom right output must have the maximum combinational delay, which is B[0] to P[31] and A[14] to P[31], the reason both have the almost same maximum delay is this two is going to an and gate and the output f this gate goes the maximum path.