High-Level Model of c5315

Statistics: 178 inputs; 123 outputs; 2406 gates

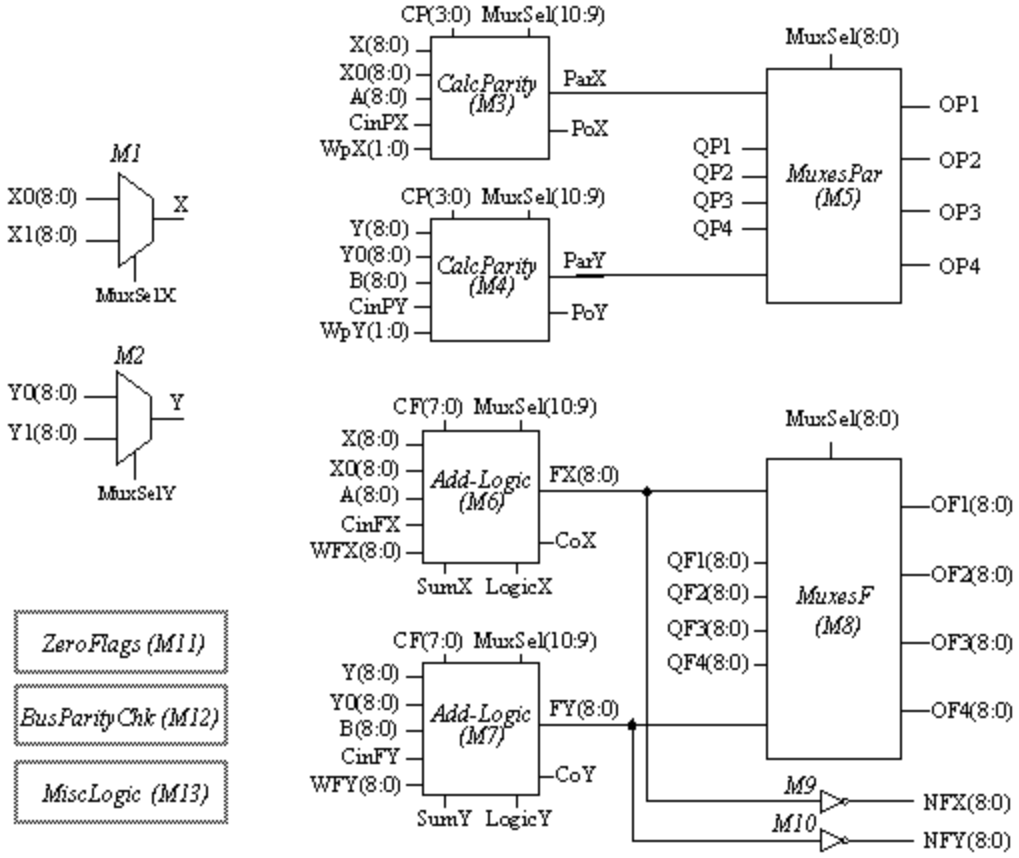
Function: 9-bit ALU
This benchmark is an ALU that performs arithmetic and logic operations simultaneously on two 9-bit input data words, and also computes the parity of the results. Modules M6 and M7 each compute an arithmetic or logic operation specified by the control input bus CF[7:0]. Module M5 consists of multiplexers that route the results of M6 and M7 and four input buses to its four outputs. Output buses OF1 and OF2 can also be set to **logic 0** by MuxSel[8]. Modules M3 and M4 compute the parity of the result of the operation given by CP=CF[7:4]. Module M5 contains four multiplexers which direct the parity results and an additional set of four inputs to its outputs. The adders in M6 and M7 as well as the parity logic for the arithmetic operations in M3 and M4 use a carry-select scheme with 4-bit (low-order) and 5-bit (high-order) blocks. The circuit also includes logic for calculating various zero and parity flags of the input buses.

Inputs/Outputs vs. Netlist numbers

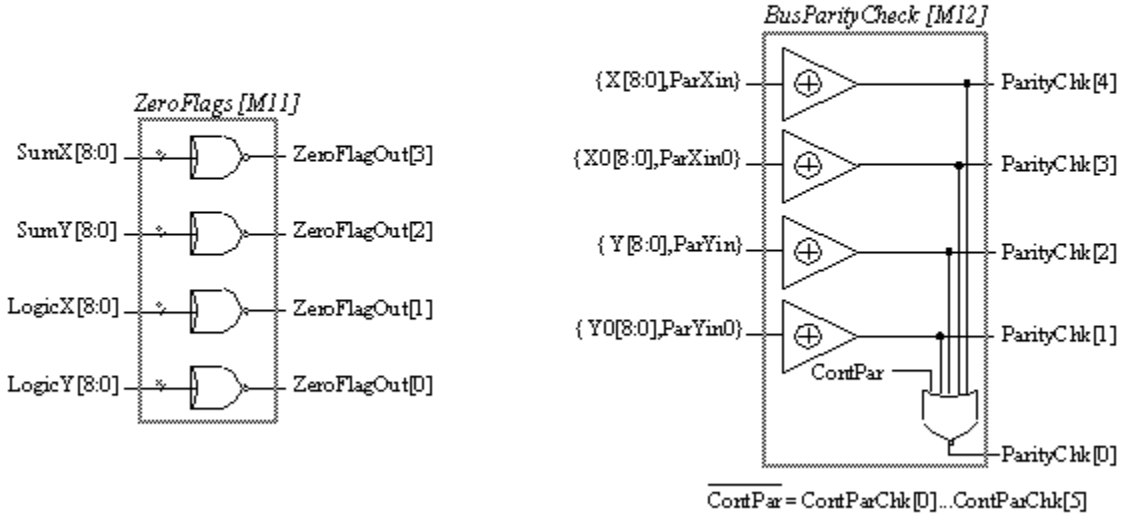
Models:

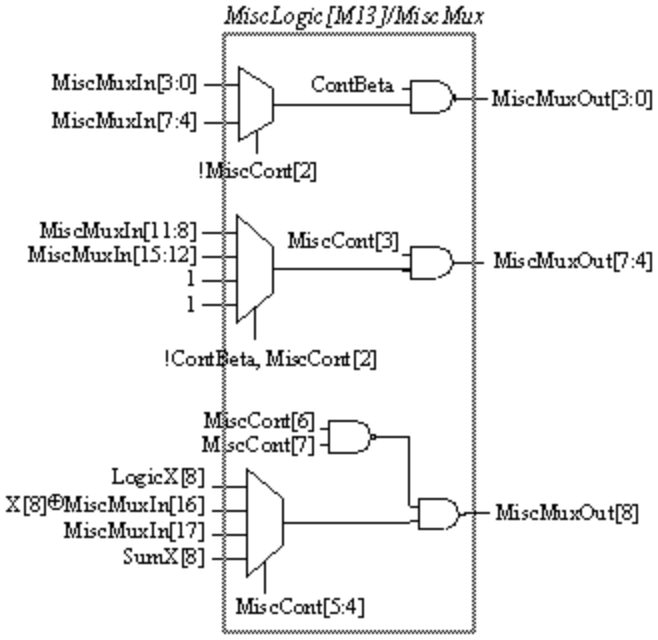
- I. Original ISCAS gate-level netlist
  - in ISCAS-89 format
  - in Verilog
- II. Verilog hierarchical netlist (functionally equivalent to I)
- III. Verilog flat netlist (flat version of II; functionally equivalent to I, but with minor structural differences)

High-Level Model of c5315

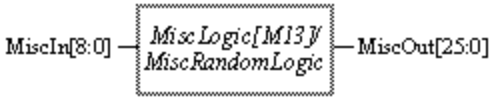


High-Level Model of c5315 [con't]

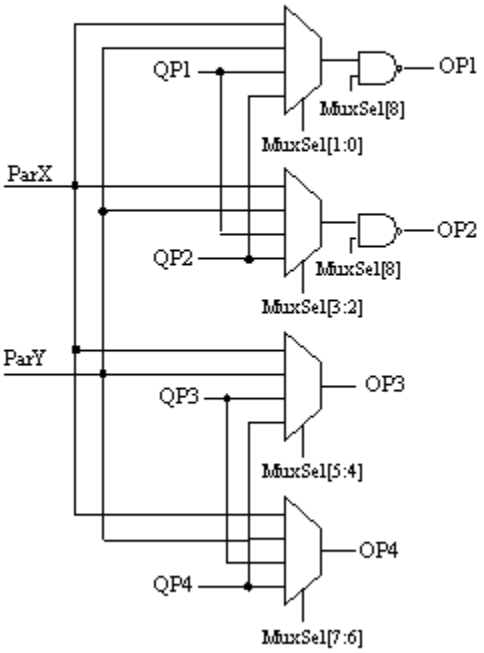




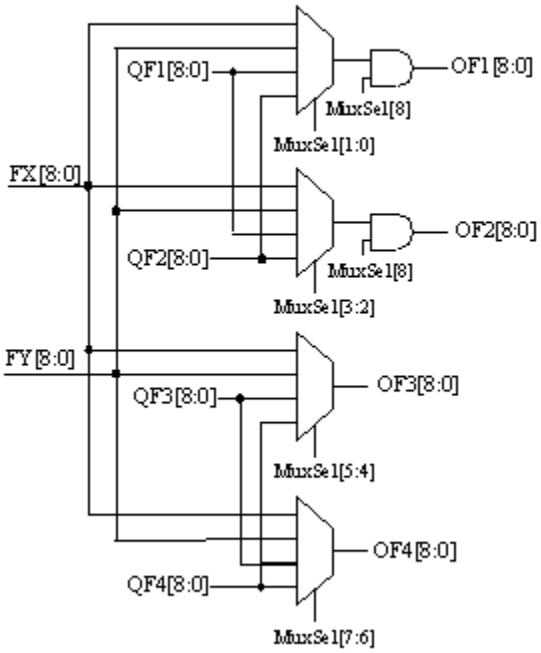
ContBeta = MiscCont[0].MiscCont[1]



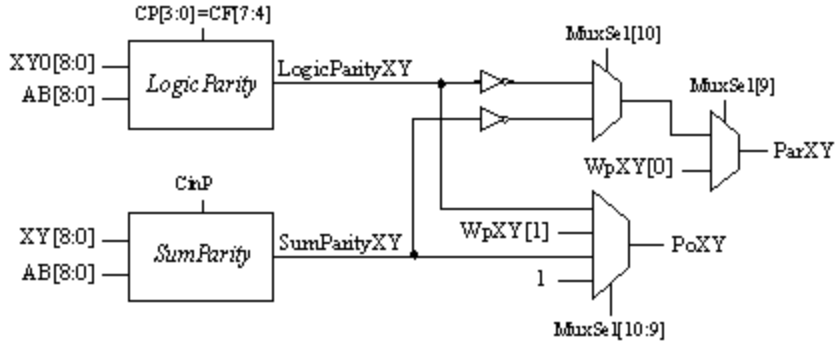
MuxesPar [M51]



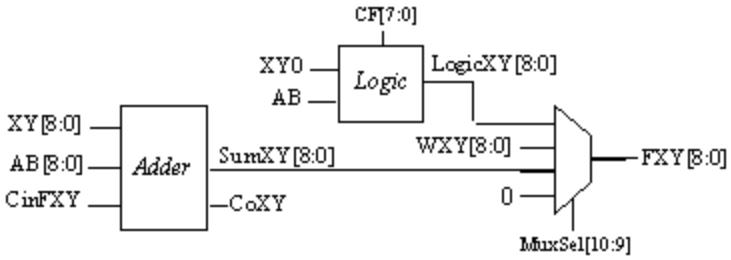
MuxesF [M81]



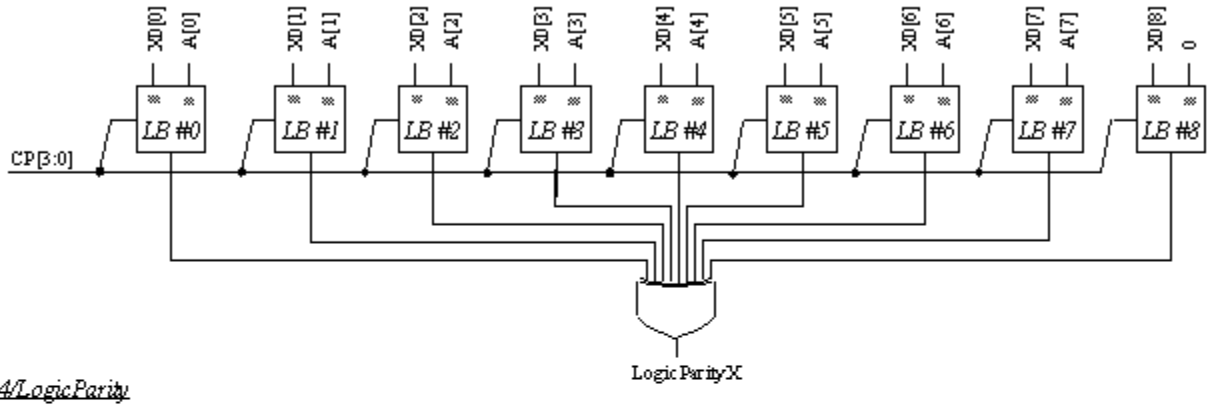
Parity [M3-M4]



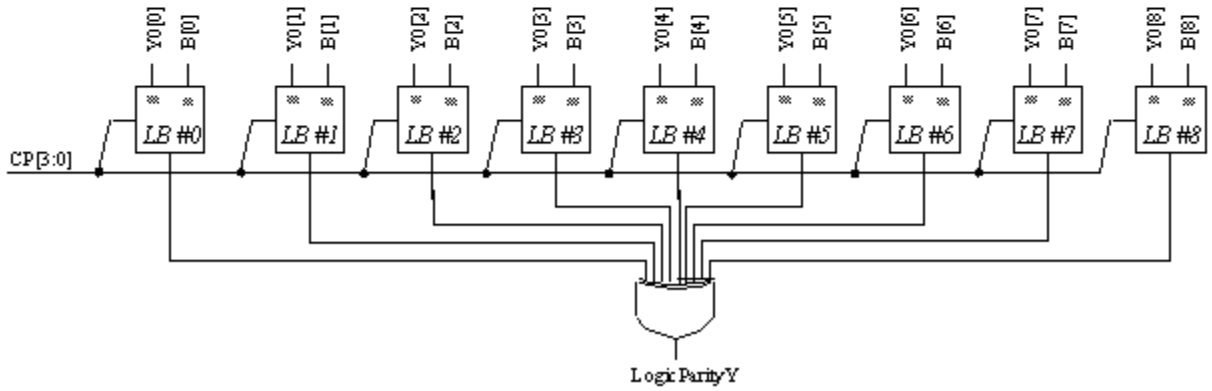
Add-Logic [M6-M7]



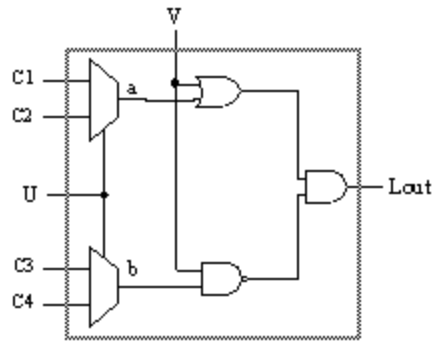
M3/LogicParity



M4/LogicParity



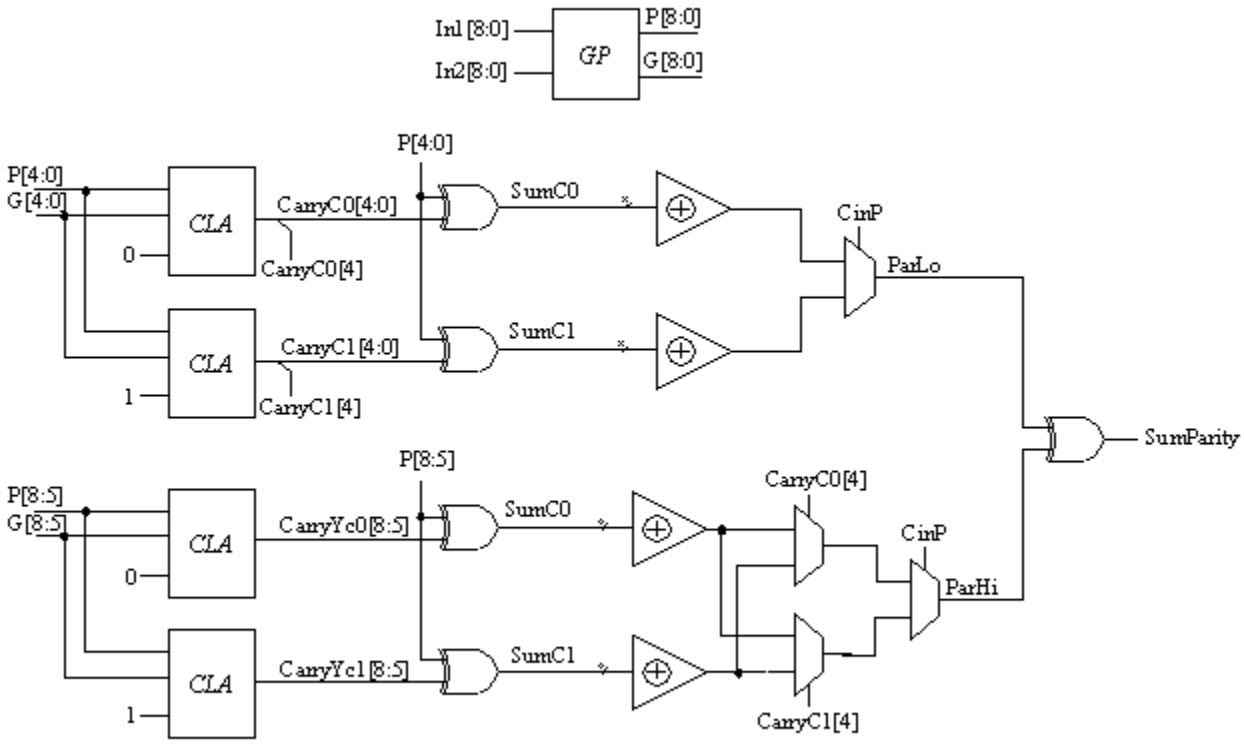
Logic Block [LB]



C1	C2	C3	C4	a	b	Lout
0	0	0	0	0	0	V
0	0	0	1	0	U	U'.V
0	0	1	0	0	U'	U.V
0	0	1	1	0	1	0
0	1	0	0	U	0	U+V
0	1	0	1	U	U	U⊕V
0	1	1	0	U	U'	U
0	1	1	1	U	1	U.V'
1	0	0	0	U'	0	U'+V
1	0	0	1	U'	U	U'
1	0	1	0	U'	U'	[U⊕V]'
1	0	1	1	U'	1	U'.V'
1	1	0	0	1	0	1
1	1	0	1	1	U	U'+V'
1	1	1	0	1	U'	U+V'
1	1	1	1	1	1	V'

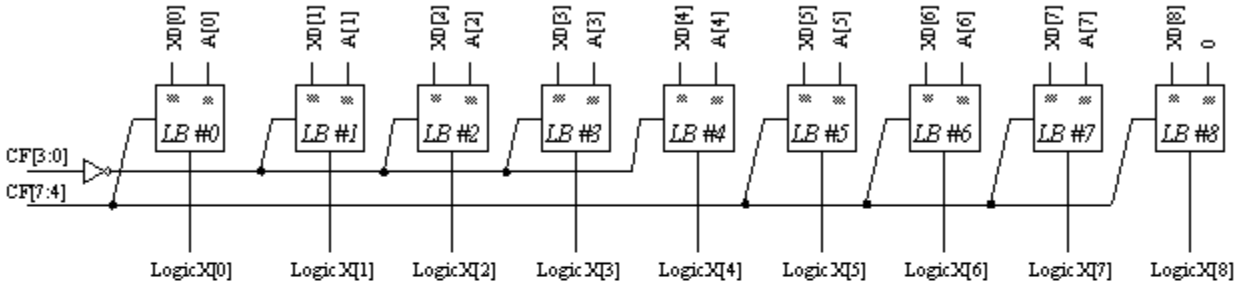
* Lout: all the 16 functions of two variables (U,V)

SumParity

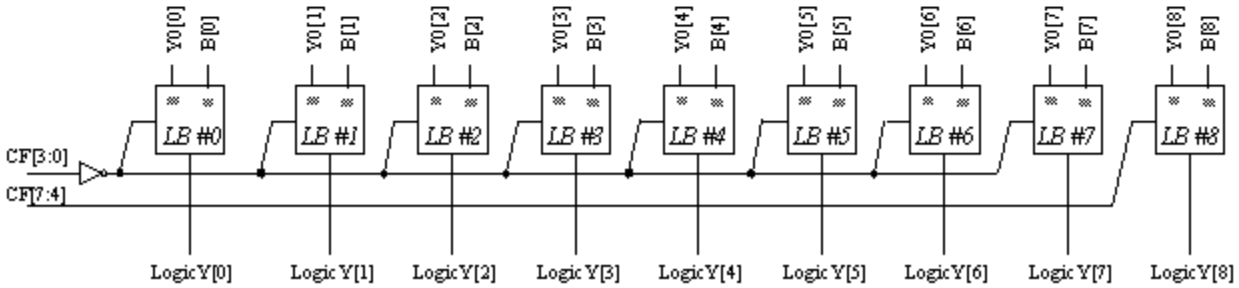


Note: Carry[i] = carry from bit position i to i+1.

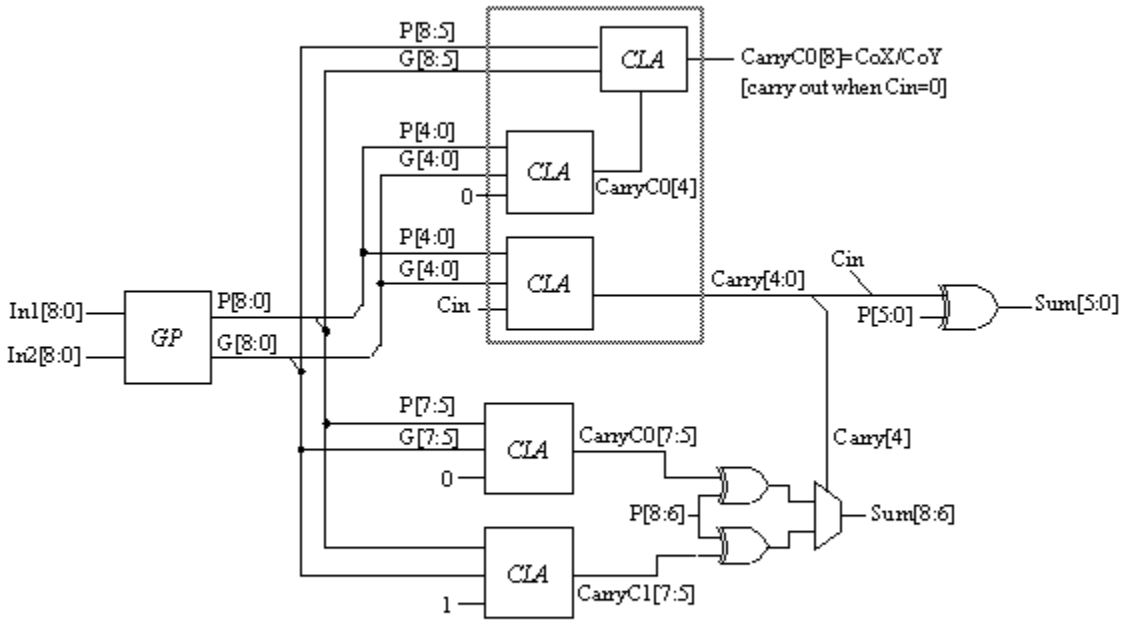
M6/Logic



M7/Logic



Adder



Note: Carry[i] = carry from bit position i to i+1.

Input	Line number
X0[8:0]	293, 302, 308, 316, 324, **logic 1**, 341, 351, 361
X1[8:0]	299, 307, 315, 323, 331, 338, 348, 358, 366
MuxSelX	332
A[8:0]	**logic 1**, **logic 1**, 479, 490, 503, 514, 523, 534, **logic 1**
Y0[8:0]	206, 210, 218, 226, 234, 257, 265, 273, 281
Y1[8:0]	209, 217, 225, 233, 241, 264, 272, 280, 288
MuxSelY	335
B[8:0]	446, 457, 468, 422, 435, 389, 400, 411, 374
CinFX, CinFY	54, 4
CinPX,CinPY	2174, 1497
WpX[1:0]	120, 94
WpY[1:0]	118, 97
QP1,QP2,QP3,QP4	176, 179, 14, 64
Q1[8:0]	191, 194, 197, 203, 200, 149, 155, 188, 182
Q2[8:0]	161, 164, 167, 173, 170, 146, 152, 158, 185
Q3[8:0]	109, 46, 100, 91, 43, 76, 73, 67, 11
Q4[8:0]	106, 49, 103, 40, 37, 20, 17, 70, 61
WFX[8:0]	123, 121, 116, 112, 52, 130, 119, 129, 131
WFY[8:0]	115, 114, 53, 113, 122, 128, 127, 126, 117
MuxSel[10:0]	4091, 4092, 137, 4090, 4089, 4087, 4088, 1694, 1691, 1690, 1689
CF[7:0]	248, 251, 242, 254, 3552, 3550, 3546, 3548
CP[3:0]=CF[7:4]	248, 251, 242, 254
ParYin= MuxSelY ? ParYin0 : ParYin1 (ParYin0, ParYin1)	289, 292
ParXin= MuxSelX ? ParXin0 : ParXin1 (ParXin0, ParXin1)	369, 372
ContParChk[5:0]	562, 245, 552, 556, 559, 386
MiscMuxIn[17:0]	123 (=WFX[8]), 132, 23, 80, 25, 81, 79, 82, 24, 26, 86, 83, 88, 88, 87, 83, 34, 34
MiscContIn[7:0]	4115, 135, 3717, 3724, 141, 2358, 31, 27
MiscIn[8:0]	545, 549, 3173, 136, 1, 373, 145, 2824, 140

Output	Line number
OP1,OP2,OP3,OP4	658, 690, 767, 807
OF1[8:0]	654, 642, 651, 648, 645, 670, 667, 664, 661
OF2[8:0]	688, 676, 685, 682, 679, 702, 699, 696, 693
OF3[8:0]	727, 747, 732, 737, 742, 752, 757, 762, 722

OF4[8:0]	712, 787, 772, 777, 782, 792, 797, 802, 859
NXF[8:0]	824, 826, 828, 830, 832, 834, 836, 838, 822
NYF[8:0]	863, 865, 867, 869, 871, 873, 875, 877, 861
CoX,CoY	(629, 618) * , (591, 621) *
PoX,PoY	843, 882
ParityChk[4:0]	998, 1002, 1000, 1004, 854
ZeroFlagOut[3:0]	585, 575, 598, 610
MiscMuxOut[10:0]	623, 813, 818, 707, 715, 639, 673, 636, 820, 717, 704
MiscOut[25:0]	593, 594, 602, 809, 611, 599, 612, 600, 850, 848, 849, 851, 887, 298, 926, 892, 973, 993, 144, 601, 847, 815, 634, 810, 845, 656

*. (a,b): a,b are identical outputs.

Go to top of this file

Go back to the Benchmark List