TPAC1.1 FDR: Supporting documentation: v0.9

preShape pixel changes

The table below summarises the changes made to the preShape pixel:

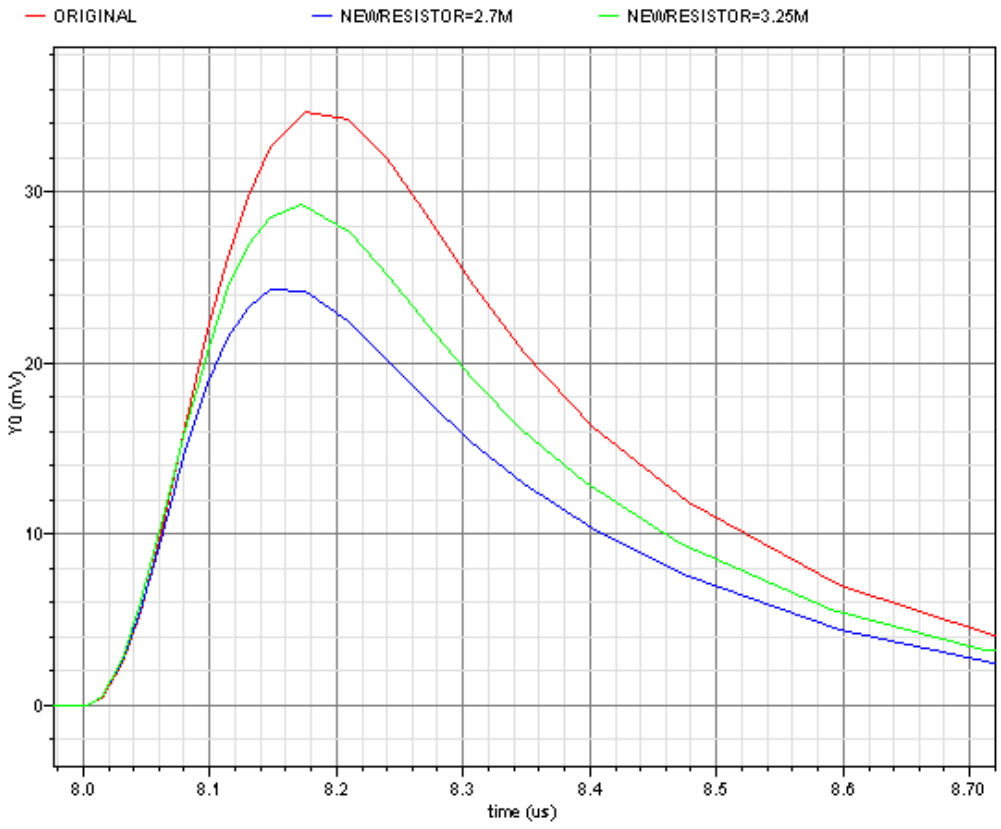
Item	Description	Schematic	Layout
1	Configuration SRAM shift register increased from 5 bits to 7 bits serial data	✓	✓
2	Comparator trims updated from 4 to 6 bit trim adjustment resolution. • Weighted current mirror previously had mirror transistors in multiples 2,4,8,16 • Added mirror transistor multiples 1,32	✓	✓
3	Comparator and monostable power supply "dco" separated such that monostable uses the now redundant "mso" power net to reduce potential for coupling between the two.	✓	✓
4	Adjusted path to diodes, and guard ring, to allow the extra sram and trim bits to be added to the pixel layout more easily. Minor increase in parasitic capacitance on diode node.	-	✓
5	Compacted layout of monostable	-	✓
6	Converted VDD1V8sram to mesh	-	✓
7	Converted VDD1V8mso to mesh	-	✓
8	Changes to resistors (See below)	✓	✓
9	Deep P-well layout (To do)	-	×

preShape resistor values

After trial verification with the calibre LVS tool it appears the original design was manufactured with the incorrect values of hi-resistance poly resistors. The schematic and layout "pcell", (which automatically generates the width/length of the resistor from the resistance value entered), created resistors that were physically too short. The assura rules used at the time did not report this error, but new calibre rules report the parameter mismatch, and the spice documentation lists the unit resistance for hi-poly resistors such that is agrees with the calibre result.

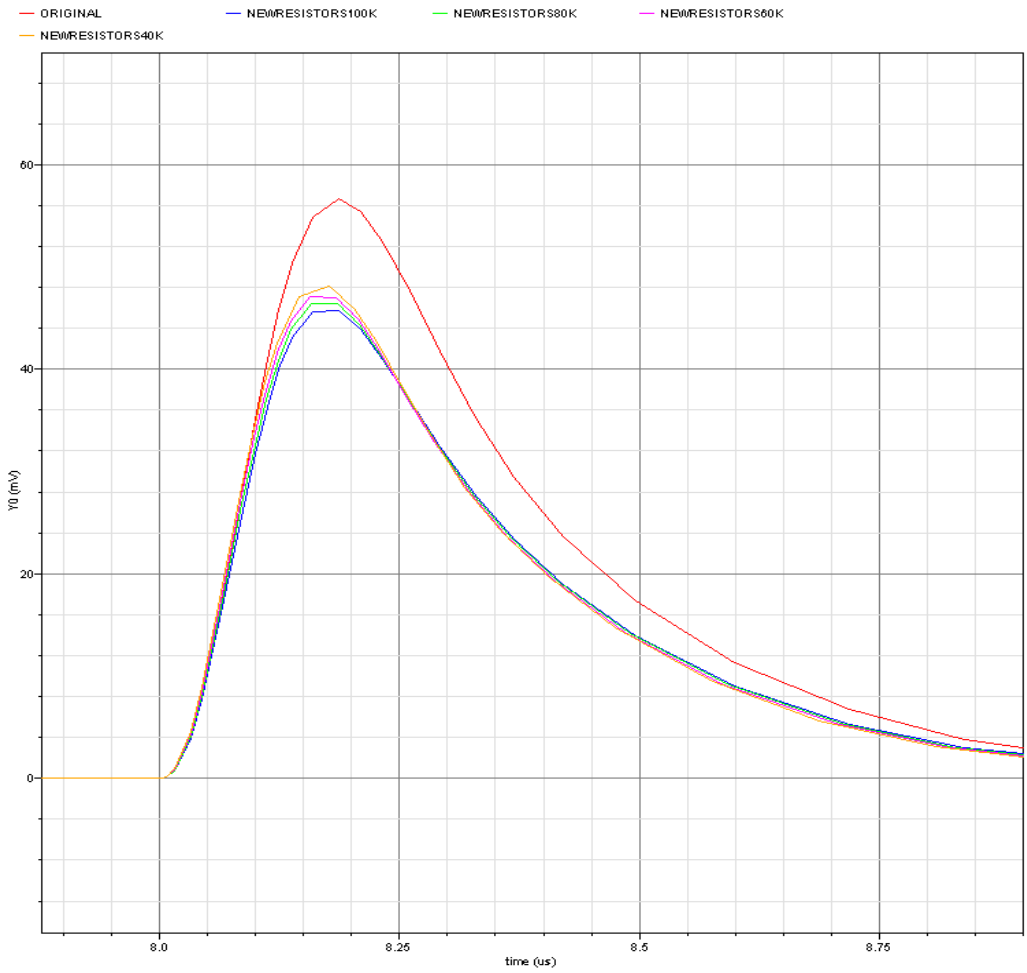
(awaiting comment from foundry)

Calculating the new value for the 4MΩ feedback resistor gives a value of 2.7MΩ. Typical shaper output is shown for the original and newly calculated are shown below in red & blue respectively.



There was not room to enlarge the resistor to restore the full 4MΩ value, but an additional 550KΩ could be inserted in the remaining available space, hence the green trace in the plot above demonstrates the realistically possible improvement on the 2.7MΩ case.

The reduction in resistance from designed value also affects a 100KΩ resistor at the input to the shaper, although the effect of this resistor on the circuit performance is negligible.



The plot above shows the minor improvements possible by adjusting the small resistor. I propose to leave this resistor un-touched in the new design, at ~60KΩ, (pink trace).

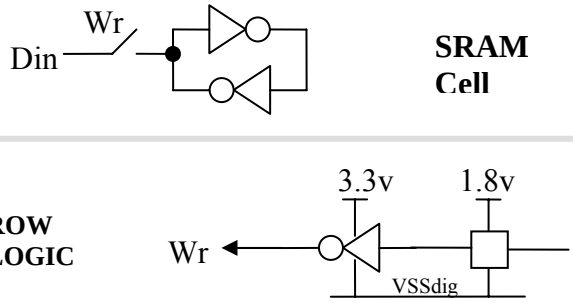
Row logic changes

The table below summarises the changes made to the row control logic:

Item	Description	Schematic	Layout
1	Level shifting NAND gate inserted to correctly up-rate the SRAM write voltage *(more detail follows this table)*	✓	✓
2	Changed strength of inverters in cell mux6x7 to adjust for new loading conditions caused by new nand cell	✓	✓
3	Compacting of layout to make sufficient space for new NAND gate, in cells • mux6x7 • master_controller42.nls	-	✓
4	Layout changes to interfacing/connecting blocks • logic_rpt_buffers • logic_rpt_buffers_ringconnect • sram_rpt_buffers_DN • lgcpwr_tapv1 • lgcpwr_tapv2 • lgcpwr_tapvtop	-	✓

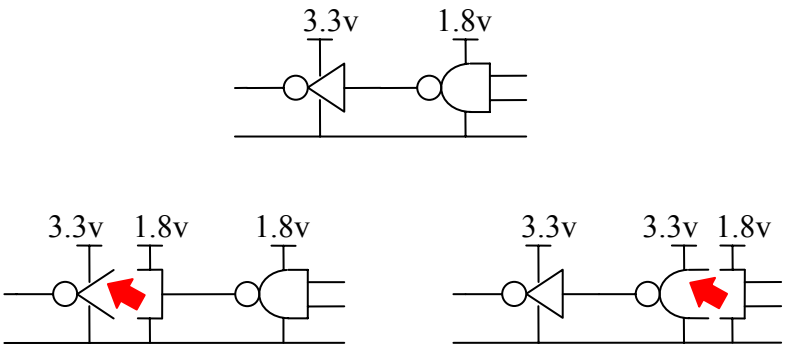
Logic modification: Level shifting NAND gate

In order to overpower the SRAM cells the write signal must be driven >1.8v to successfully write a '1'. In the ASIC1 design the write signal is generated in the row logic and then buffered with an inverter powered with VDD2V5dig, but shares the VSSdig ground net with the rest of the digital row logic. The VDD2V5dig net was up-rated late in the design process from 2.5v to 3.3v to avoid bit errors seen in monte-carlo simulations (the net name remained)



The problem with this circuit is that a 1.8v logic high from the row logic sits in the switching point for the 3.3v inverter, turning both transistors on, thus drawing large static currents such that it cannot be used. ASIC1 was successfully operated using ~2.6v for this inverter's power supply, which successfully wrote data into the SRAMs but still had a moderate static current flow.

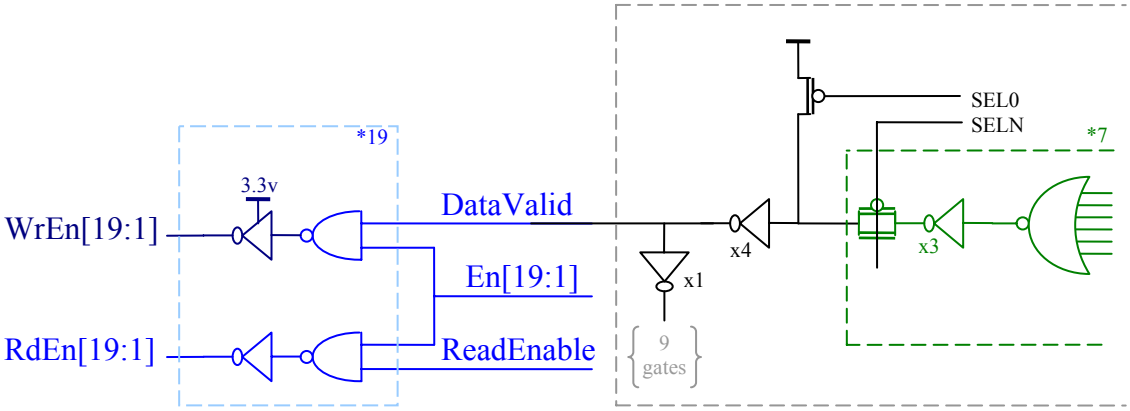
The solution for ASIC2 is to replace the inverter or NAND gate with a level-shifting circuit that correctly up-scales the digital signal to 3.3v without excessive static current. The existing and possible new circuits are illustrated below:



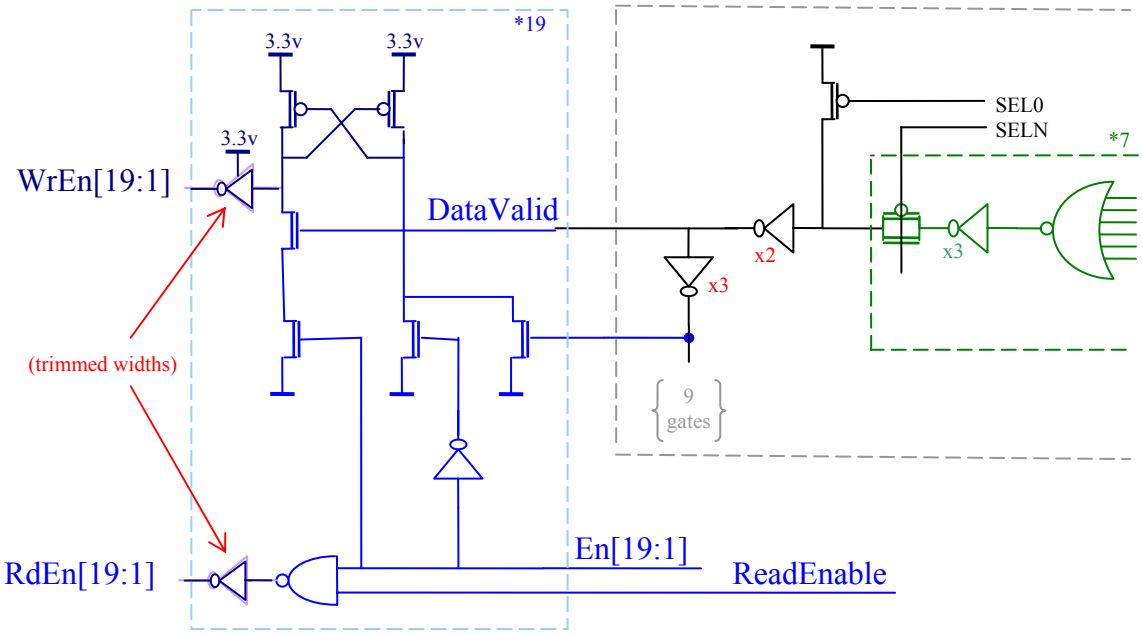
The number of transistors used can be reduced by choosing to implement the level shifting NAND gate.

Practical implementation

Original circuit (in context)

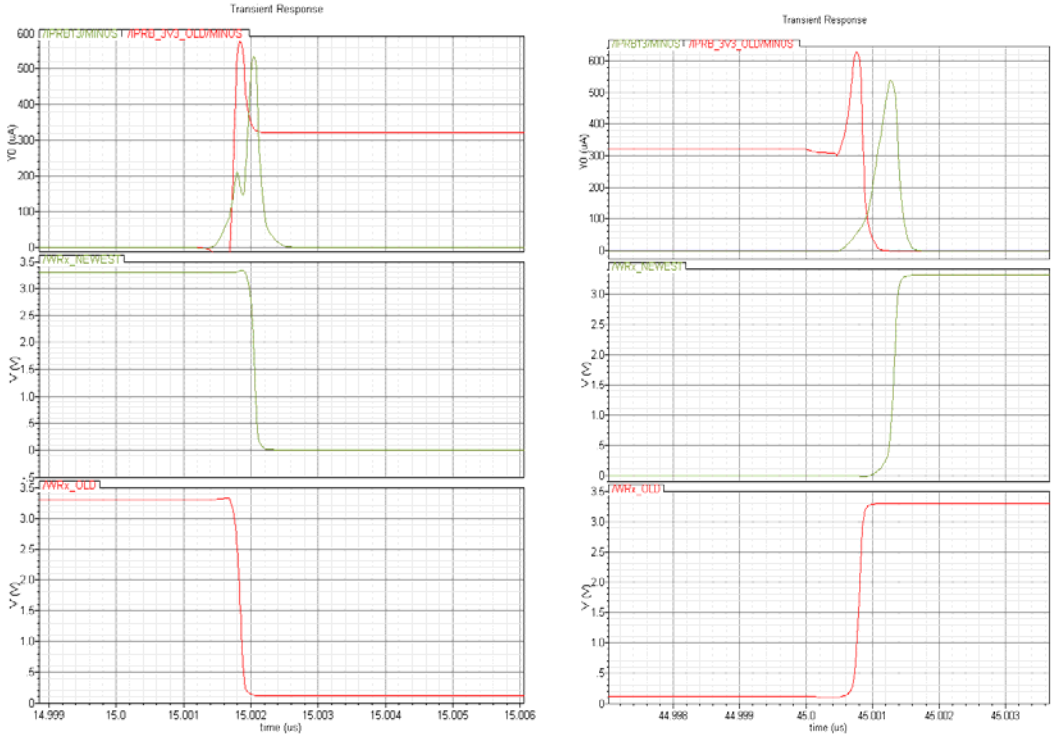


New circuit (in context)



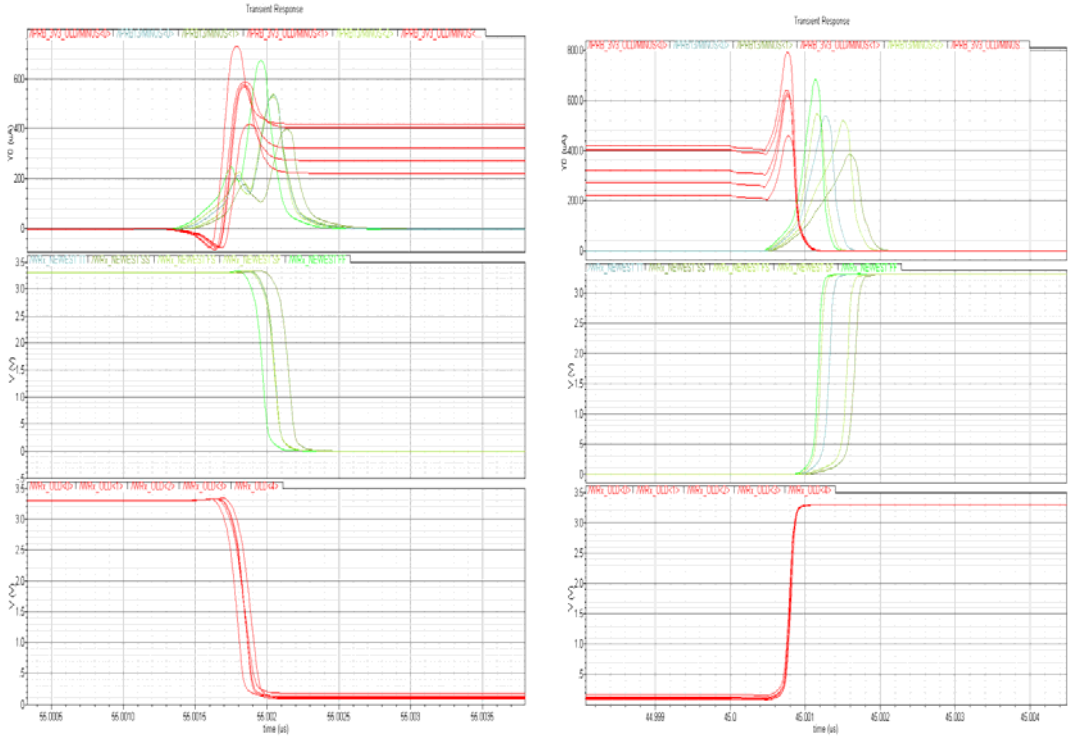
Logic simulation: New level shifter compared with original

The original circuit was modified for simulation by adding an ideal level-shifting device (vcvs) between the NAND gate and the inverter. Simulating this and comparing with the new circuit shows similar performance driving a 250fF load, and equivalent functionality with the same input signals.



Above: The previous circuit (red) and new circuit (green) performance clearly shows the design flaw in the original design, where a WrEn=0 signal draws a static 300uA from the power rails. In the new case, the transient switching current remains (as the cell is driving a moderate load) but the static current for both states of WrEn is zero. In both rising and falling cases, there is an additional time delay of ~1ns with the new circuit due to the extra complexity in the circuit. This is an acceptable performance.

The same comparison is made in the five process corners:

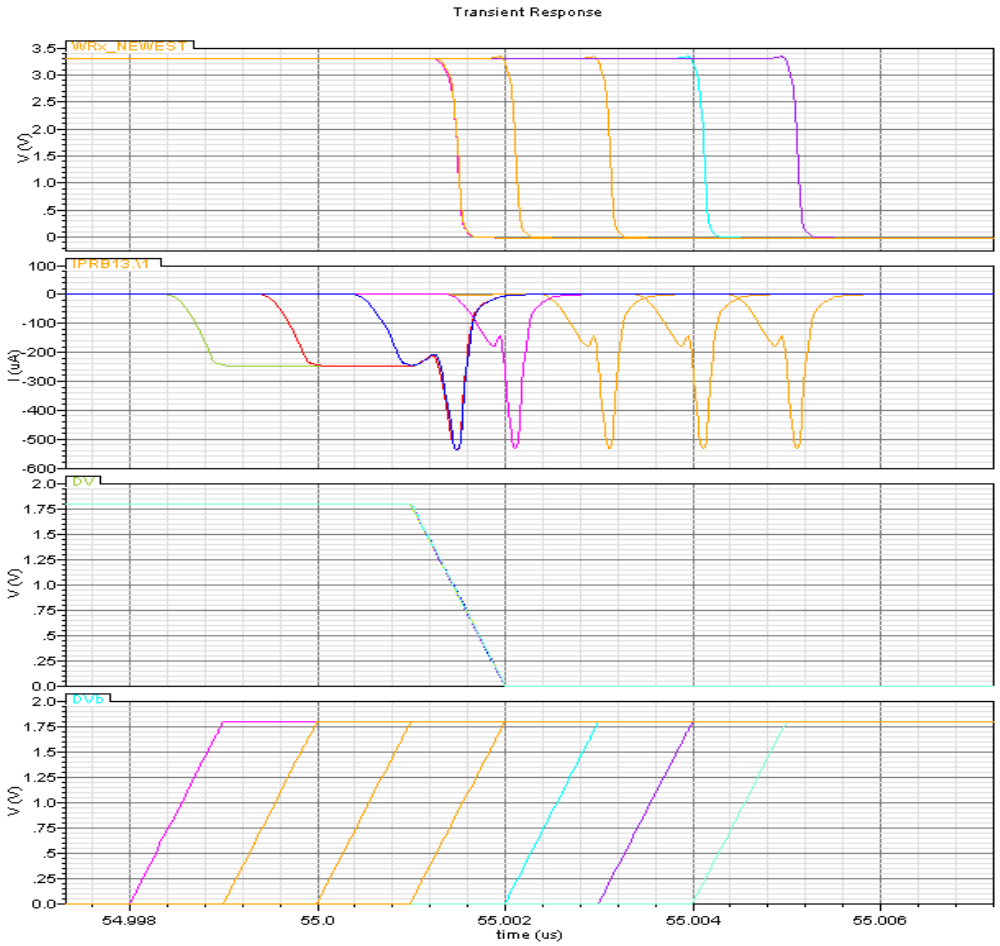


Operation in all process corners is acceptable.

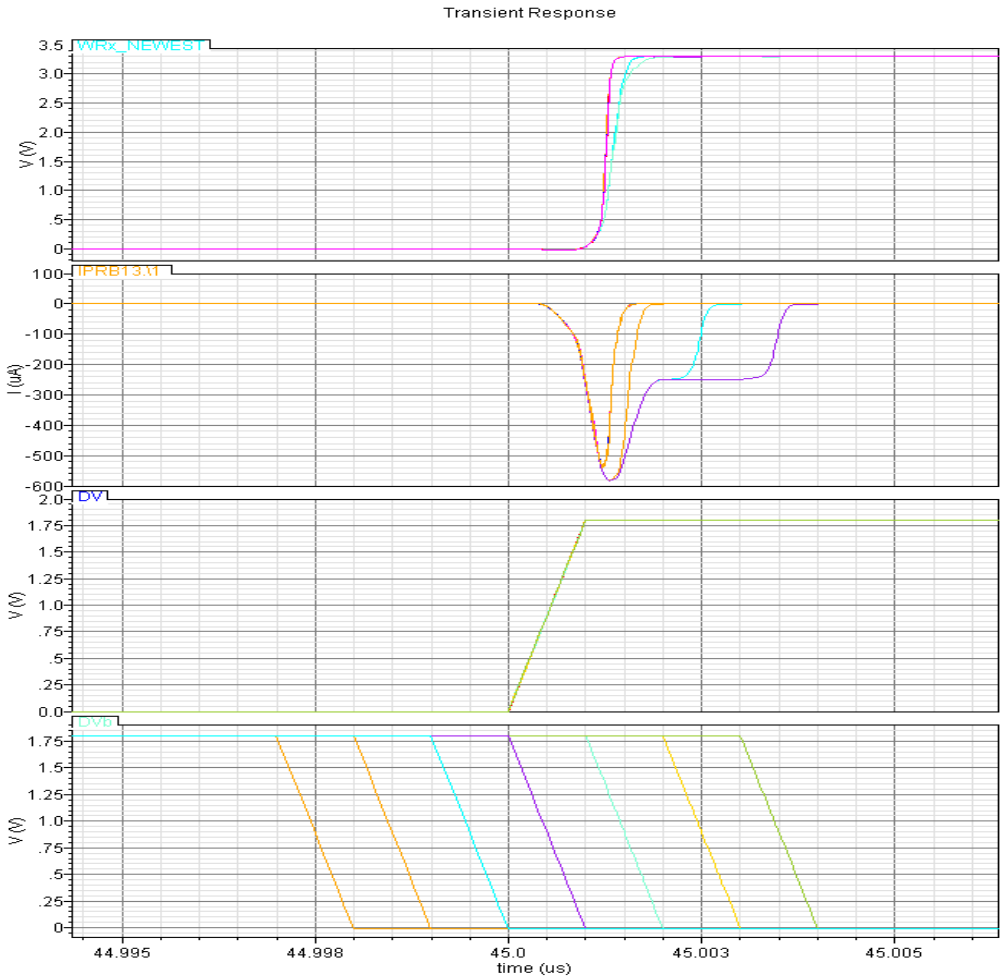
Logic simulation: New level shifter input signals

The new level shifter circuit requires complimentary dataValid inputs, and so it is possible for the two inverse inputs to lead/lag each other. It is important to check whether this adversely effects the performance of the circuit, as they may be generated elsewhere in the logic to save layout space.

Applying an artificial 3ns lead/lag in a simulation yields the following results :



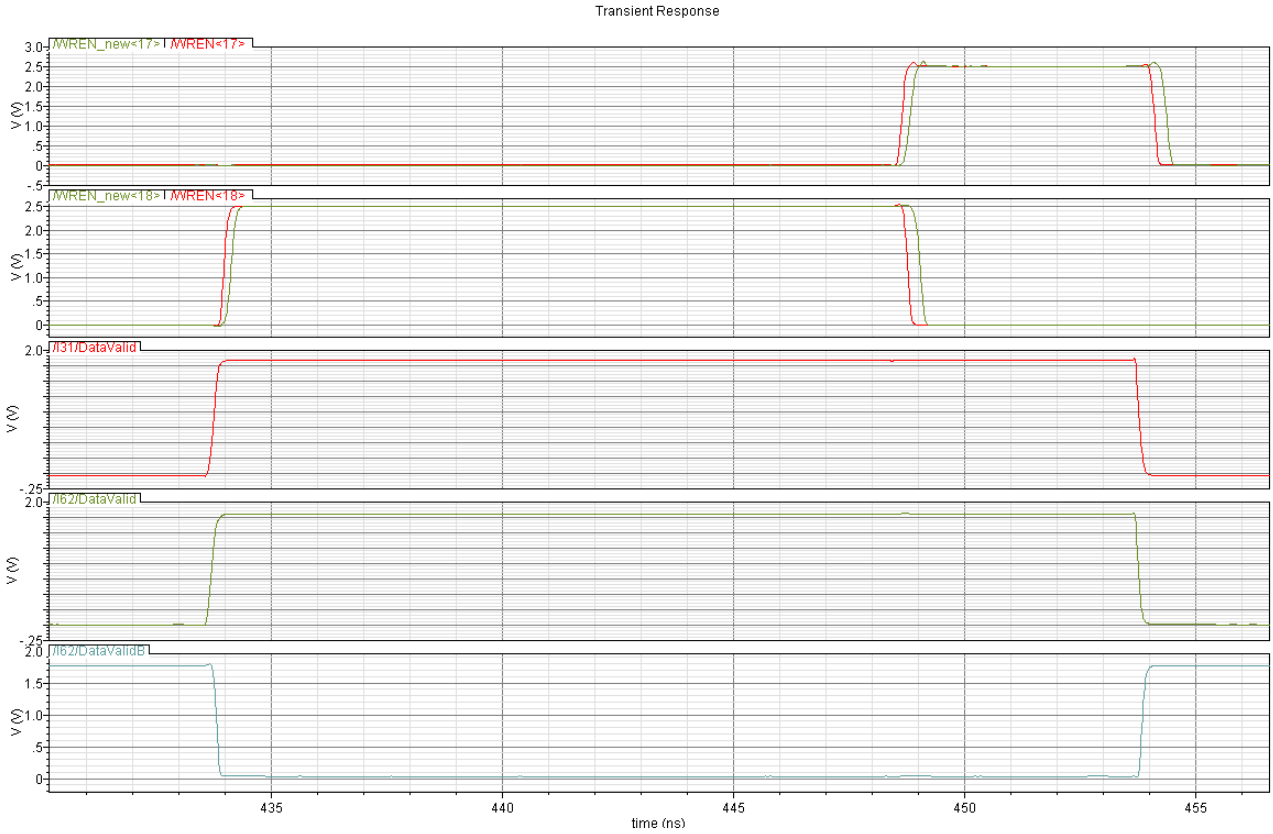
Above: The effect of dataValid lead/lag on the falling edge of write-enable. This shows that any delay to DVb will be applied to the WE output signal, but there is no concerning circuit behaviour in any of the scenarios.



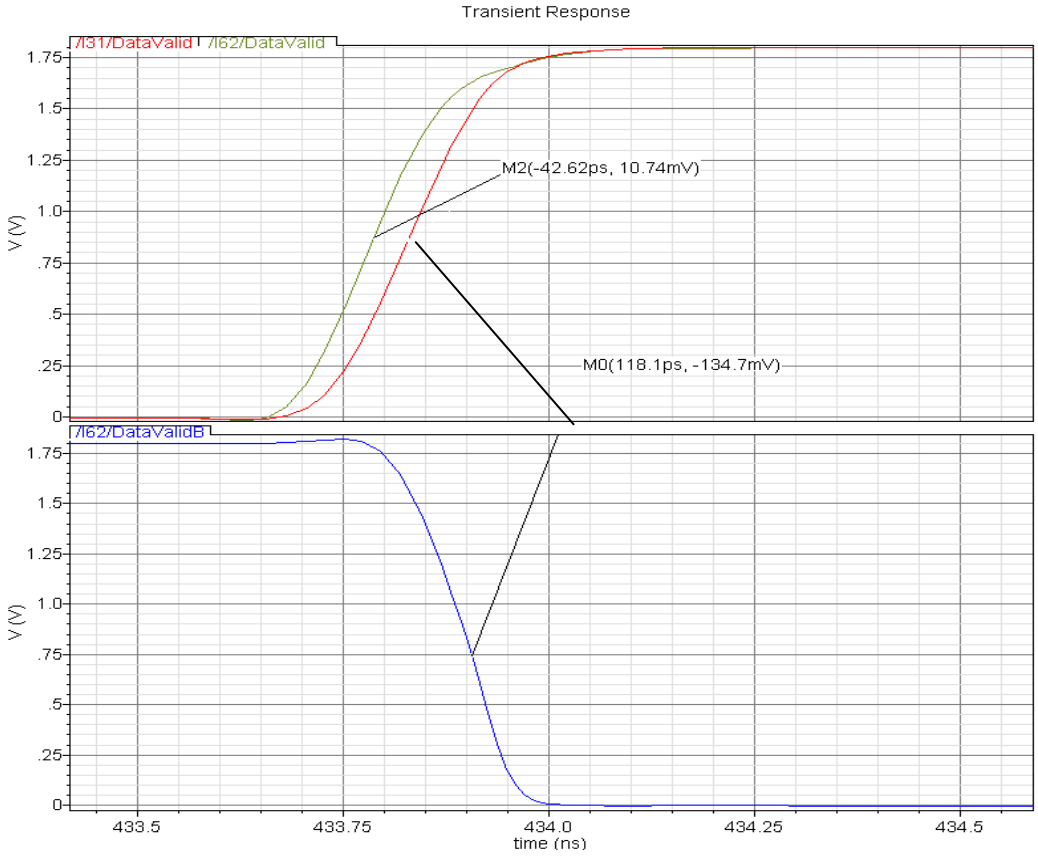
Above: The effect of dataValid lead/lag on the rising edge of write-enable. This shows that any delay to DVb will result in higher total current consumption but not significantly effect the edge of WE. Therefore it is desirable to minimise any lag from DV to DVb, or design for DVb to lead DV.

Full Logic Simulation including SRAM: Old vs. New

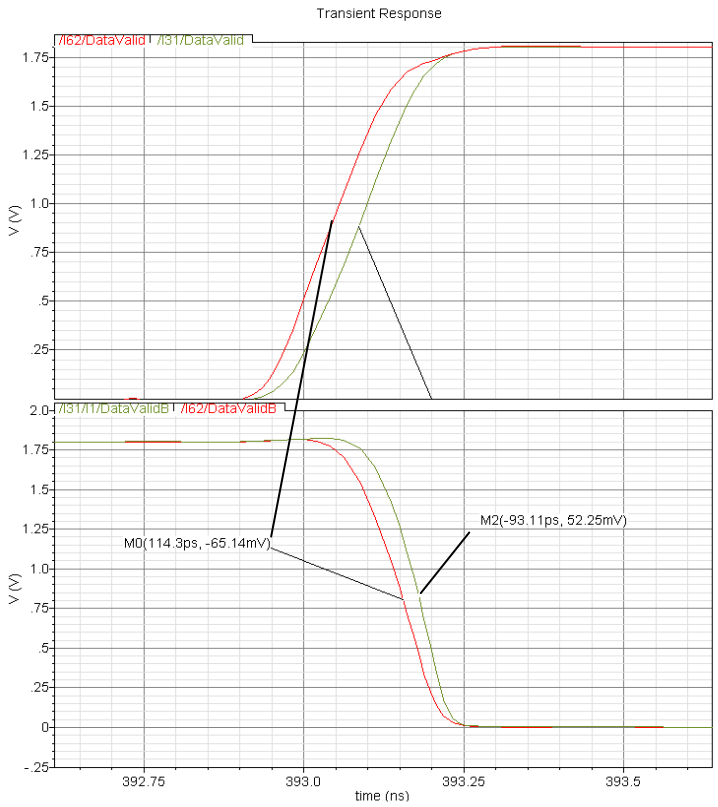
Typical process corner. New logic implementing the level-shifting circuits is driven with identical stimulus as the original logic. Key signals are plotted below to illustrate the difference (similarity) and consistent signal behaviour.



Above: Write signals lag by ~300ps but don't show any significant edge speed degradation due to the width trimming between old and new logic circuits. The DataValidB signal is seen to coincide well with the DataValid signal (note that it cannot lead as is preferred, since it is generated directly from DataValid itself – the alternative arrangement is not possible due to space limitations)



Above: Demonstration of previous (red) and new (green) dataValid signal timing (fs corner) and how the new DataValidB signal compares in timing.



Left: previous (green) and new (red) comparison of dataValid signals. Note how dataValidB in the new case has not been compromised by connecting an additional 19 transistor gates.

Full Logic Simulation

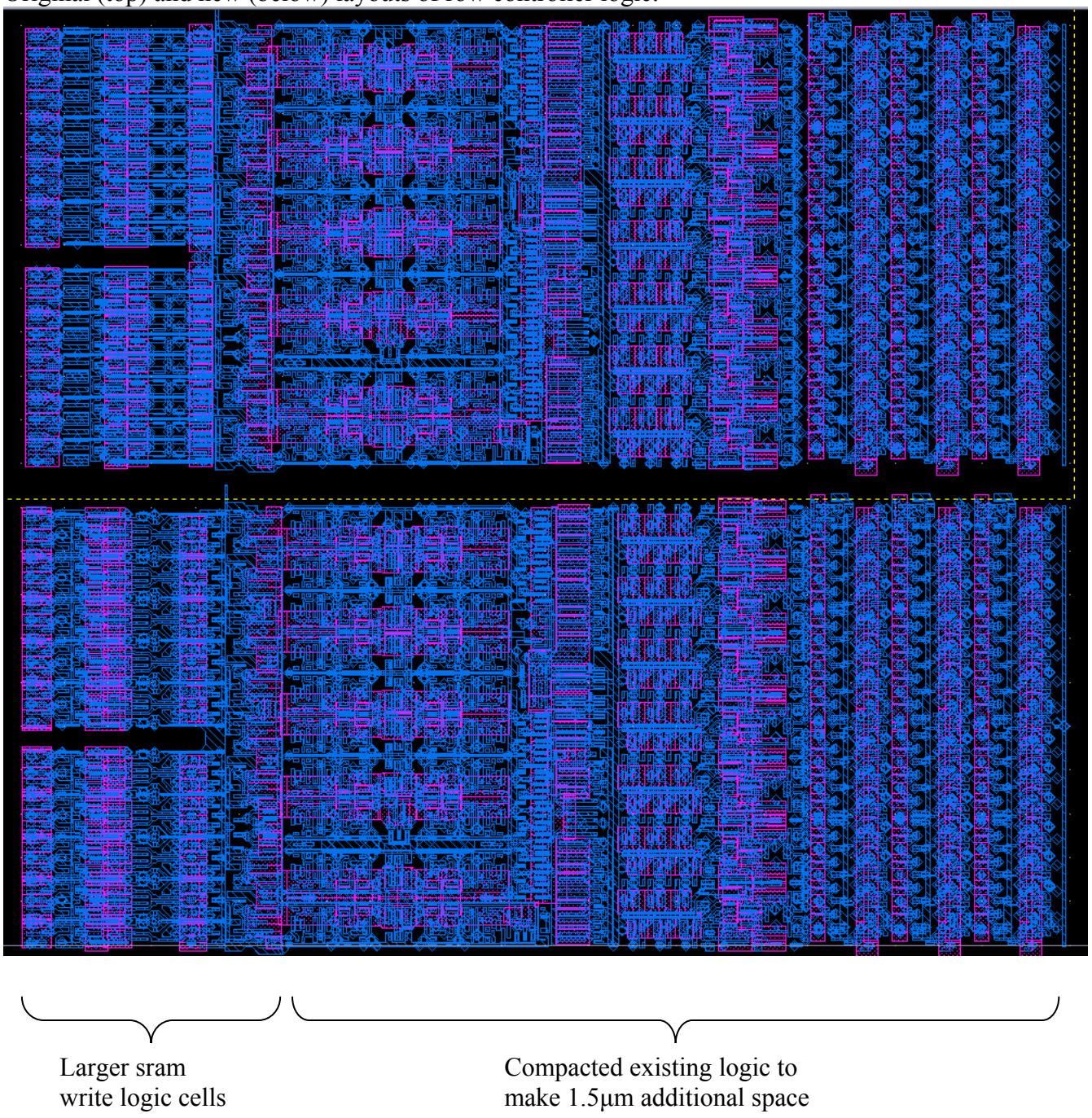
The new logic block is simulated with SRAM memories, data sense amplifiers and self-checking Verilog stimulus to verify that the hits read back correspond to those generated at the input to the logic. The output transcript is pasted below:

```
Reset the SR control registers
SR Reset complete
Latch current Hit Pattern:
Hit pattern   = 10000100000011001100000000000000000000000
Latch current Hit Pattern: Round 2
Hit pattern   = 00011100011000010100010000001100001000001
Latch current Hit Pattern: Round 3 (1 hit)
Hit pattern   = 10000100000000000000000000000010010000000
Readout phase:
SR Initialisation complete
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000011
     >>           BANK = 100
     >>           PATT = 100001
     >>           DATA = 01110000111111111111100
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000011
     >>           BANK = 011
     >>           PATT = 010010
     >>           DATA = 10001001011111111111100
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000010
     >>           BANK = 100
     >>           PATT = 000111
     >>           DATA = 01100011111111111111101
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000010
     >>           BANK = 101
     >>           PATT = 000110
     >>           DATA = 01000011011111111111101
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000010
     >>           BANK = 111
     >>           PATT = 000101
     >>           DATA = 00000011011111111111101
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000010
     >>           BANK = 110
     >>           PATT = 000100
     >>           DATA = 00100011001111111111101
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000010
     >>           BANK = 010
     >>           PATT = 000011
     >>           DATA = 10100001111111111111101
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000010
     >>           BANK = 011
     >>           PATT = 000010
     >>           DATA = 10000010111111111111101
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000010
     >>           BANK = 001
     >>           PATT = 000001
     >>           DATA = 11000000111111111111101
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000001
     >>           BANK = 100
     >>           PATT = 100001
     >>           DATA = 01110000111111111111110
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
     >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
     >> HIT INFO: TIME = 0000000000001
     >>           BANK = 111
     >>           PATT = 110011
     >>           DATA = 00011001111111111111110
     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
 -
 -
 -
 -
 -
 -
 -
 -
 -
 -
 -
 -
 -
 -
 -
 -
Verilog/spectre Interface: 2802 messages sent, 2804 messages received.
391 warnings
0 simulation events (use +profile or +listcounts option to count) + 2 accelerated events
CPU time: 0.1 secs to compile + 0.1 secs to link + 0.6 secs in simulation
End of Tool:          VERILOG-XL 05.50.003-p   Apr 30, 2008  22:57:33
```

Text comparison tkdiff shows the only differences in the date/timestamp headers for all five process corner simulations of the full logic + SRAM + readout sense amps.

Logic layout

Original (top) and new (below) layouts of row controller logic.



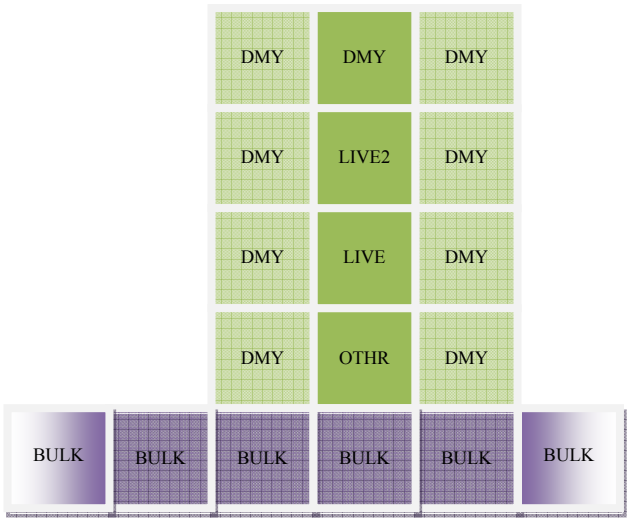
Compacting the logic has meant that various control signals move slightly, hence layout changes were necessary to all the circuit blocks that interface to the logic columns.

preShape Test Structures

The pixel test structures have been updated to implement the shaper pixel topology with access to internal analog nodes.

Additional “DMY” pixels will be added to either side of the primary set of 4 test pixels if time permits.

A brief summary of these test pixels is shown below:

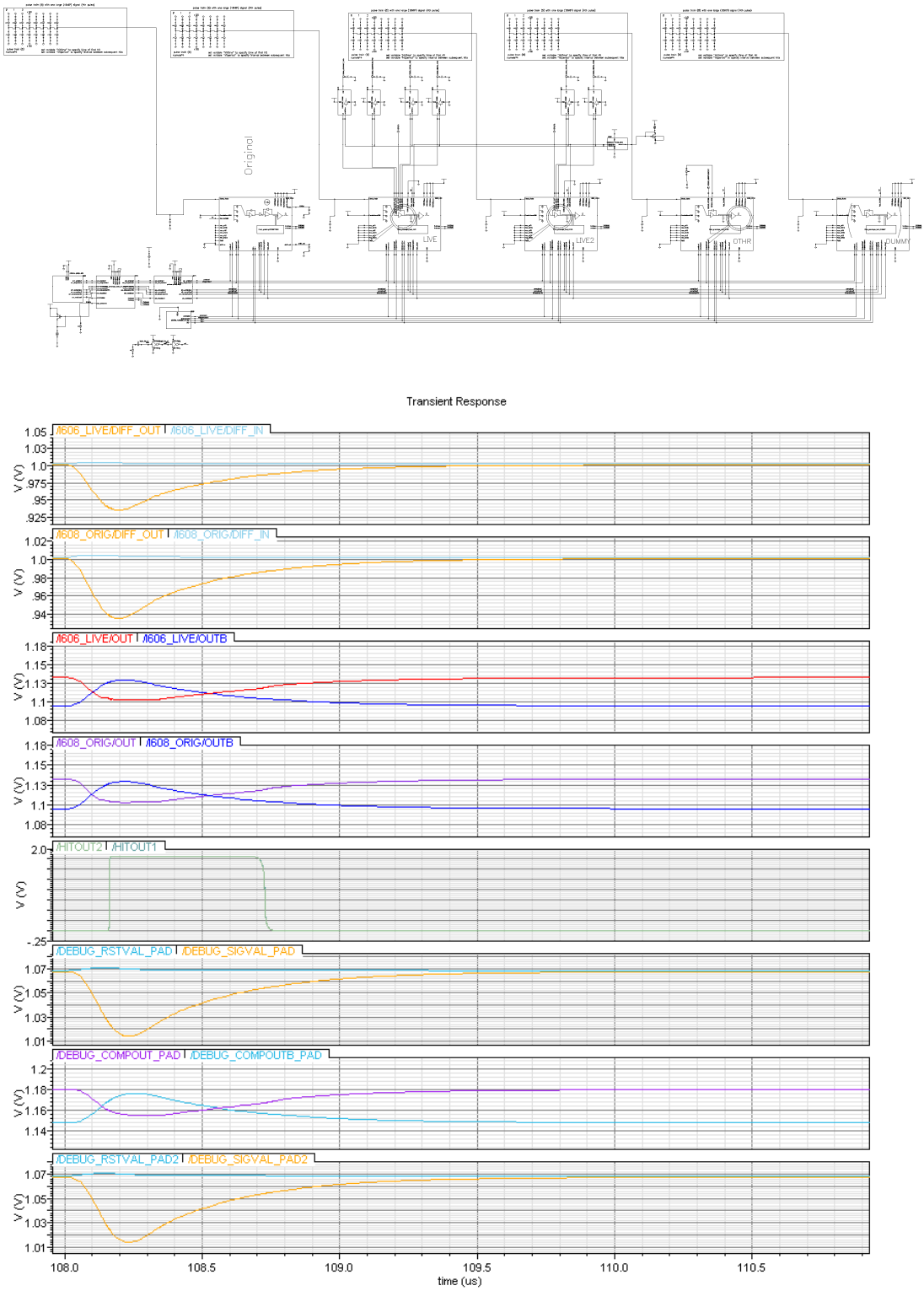


Test pixel	Schematic Cell Name	Pixel Variant	Related Inputs	Related Outputs
-	Pixel_preshape_test_setof4	-	-	-
A	Pixel_ preshape _test_LIVE2	preSample ④ (BTTBTB)	DEBUG_RST200 DEBUG_VTH+ DEBUG_VTH- DEBUG_TRIM[5:0]	DEBUG_SIGVAL2 DEBUG_RSTVAL2 DEBUG_HIT_OUT2
B	Pixel_preshape _test_LIVE	preSample ③ (BTBTTB)	DEBUG_RST200 DEBUG_VTH+ DEBUG_VTH- DEBUG_TRIM[5:0]	DEBUG_SIGVAL1 DEBUG_RSTVAL1 DEBUG_COMPN_OUT DEBUG_COMPN_OUTB DEBUG_HIT_OUT1
C	Pixel_preshape _test_OTHR	preSample ③ (BTBTTB)	DEBUG_HITIN DEBUG_VTH+ DEBUG_VTH- DEBUG_TRIM[5:0]	DEBUG_HIT200
D				

TODO: Update table

Test Pixel simulations

The new test pixels are simulated with the original for comparison to check no adverse effects on pixel performance are introduced by adding buffers to internal analog nodes.



Above: Typical response to charge deposits is checked in all test structure pixels.

preShape array changes

The table below summarises the changes made to the preShape pixel array:

Item	Description	Schematic	Layout
1	Bias block required minor redesign to attach to new pixel layout	✓	✓
2	Created preShape test pixel block using original pixel design for LVS purposes (parallel to new pixel design activity)	✓	✓
3	TO DO: Update the preShape test pixel block to implement the new pixel design	×	×

Top level changes

The table below summarises the changes made to the top level layout:

Item	Description	Schematic	Layout
1	Bias transistor for IOUTBIAS12 updated according to recommendation in problem report 1	✓	×
2	Bias transistor for ISENSEBIAS updated according to recommendation in problem report 2	✓	×
3	All pad openings enlarged from 60µm to 80µm. Overall design size grows within constraints: • Original design = 98359204 μm^2 • New design = 99154811 μm^2	-	✓
4	Full review of power distribution TBD		
5			