OUT

I2

SEL MUX2-1 OUT inv_SEL

SEL1

inv_SEL1

IN0
IN1

I0

SEL MUX2-1 OUT inv_SEL

IN0
IN1

I1

SEL MUX2-1 OUT inv_SEL
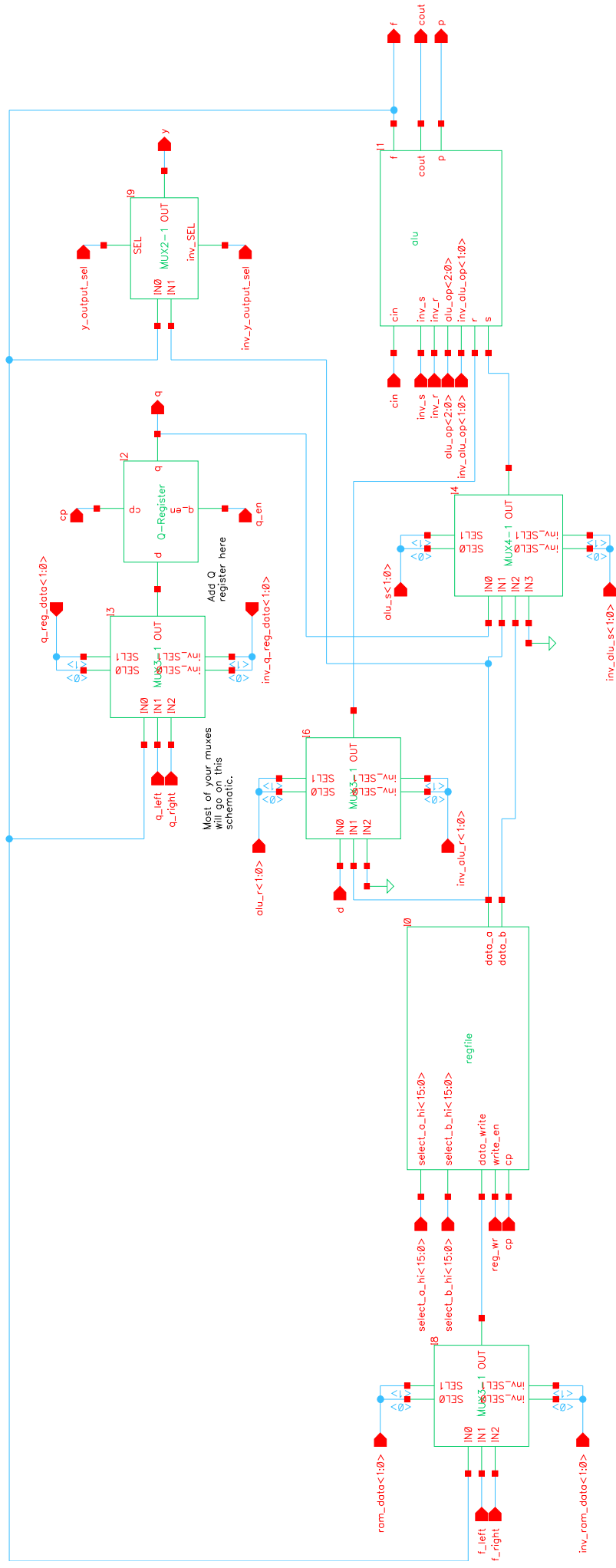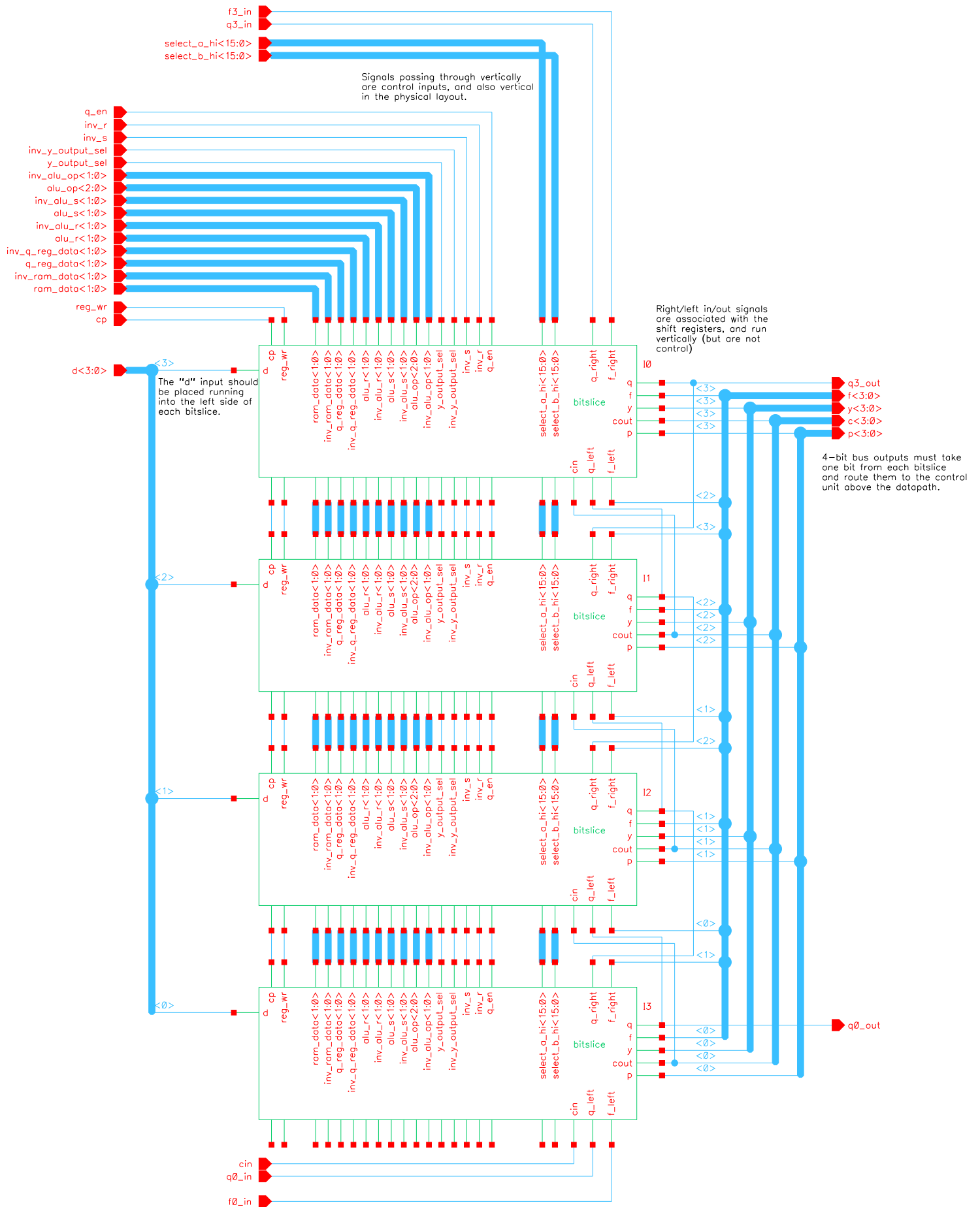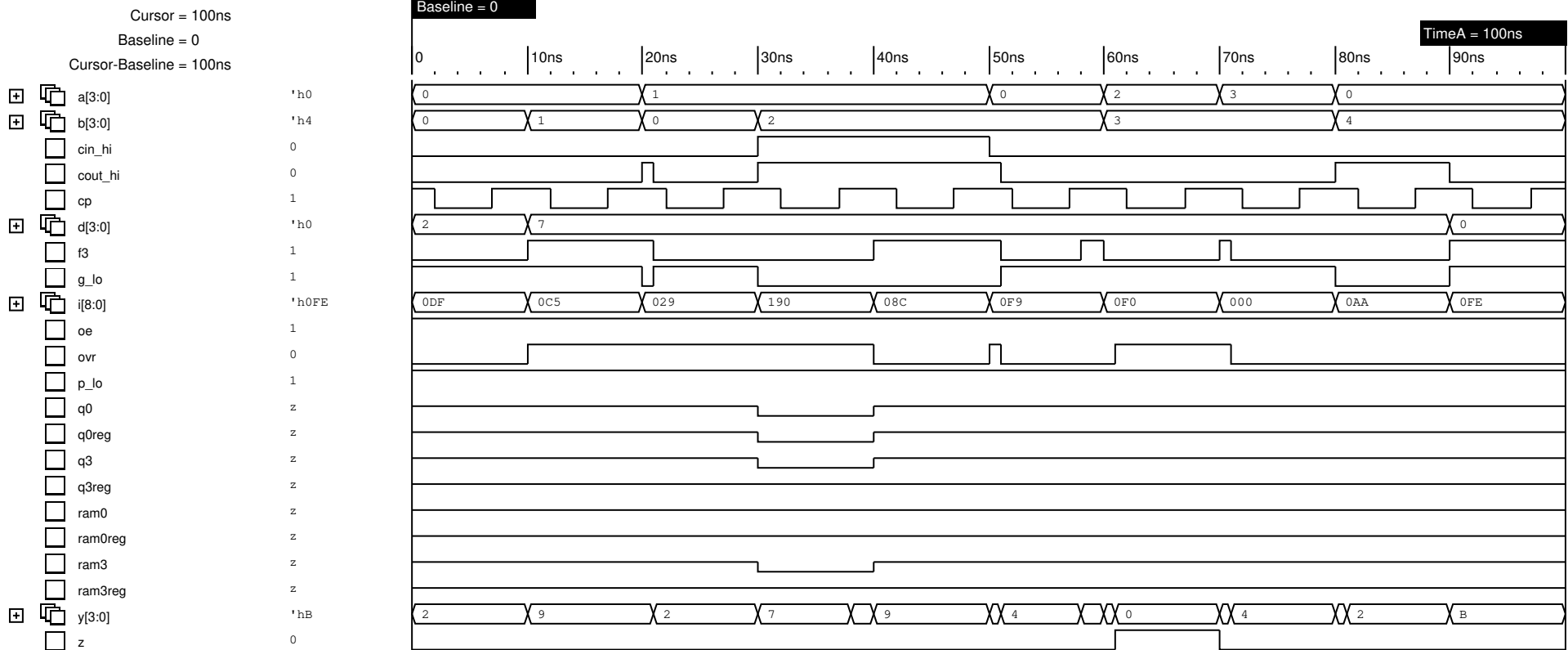
IN0
IN1

IN0
IN1

IN2
IN3

SEL0

inv_SEL0

ADD
SUBS
SUBR
OR
AND
NOTRS
EXOR
EXNOR

The ALU is controlled by the control signals *alu_op<2:0>* and *inv_alu_op<1:0>* where *inv_alu_op<1:0>* is the inverted signals of *alu_op<1:0>* as defined in the RTL code. By referring to Figure 3 in the MP-SP22 document, we were able to derive Boolean expressions to "map" the ALU inputs *i<5:3>* to the control signals *alu_op<2:0>*. These control signals are fed into a 4-IN MUX, which will control the output of the ALU. The first three functions highlighted in yellow are mapped to the binary input "00" as defined by our derived Boolean expressions. In the 4-IN MUX, "00" will pass "IN0," which is the output of the *add* module provided to us in the MP2 library. Addition is straightforward, while subtraction can be realized through the *inv_r* and *inv_s* control signals as part of the *logic* module also provided in the MP2 library. These signals effectively control the sign of the *r* and *s* input, which is necessary in order to carry out binary subtraction (i.e. by flipping the sign of either the *r* or *s* bit input). Next, the OR operation (highlighted in green) is mapped to the binary input "01," which will pass "IN1." OR is realized by passing the *nor* output of the *logic* module through an inverter. Similarly, this is done to realize AND and NOTRS, where instead the *nand* output of the *logic* module is passed through an inverter. NOTRS is essentially the same function as AND, where instead $\bar{R}$ is the input to the AND operation instead of R. Finally, the xor operation is realized by passing the *xor* output of the *logic* module to the input of another xor gate. The second input to this xor gate is the MSB of the *alu_op*, namely, *alu_op<2>*. When this bit is "0", the input to the 4-IN MUX at "IN3" will be the *xor* output of the *logic* module. However, when this bit is "1", the output of the xor gate (and the input to the 4-IN MUX at "IN3") will be the inverted *xor* output of the *logic* module, or effectively *xnor*. This effectively realizes all eight functions of the ALU.

Signals passing through vertically are control inputs, and also vertical in the physical layout.

Right/left in/out signals are associated with the shift registers, and run vertically (but are not control)

The "d" input should be placed running into the left side of each bitslice.

4-bit bus outputs must take one bit from each bitslice and route them to the control unit above the datapath.

Cursor = 100ns
Baseline = 0
Cursor-Baseline = 100ns

Baseline = 0

TimeA = 100ns

Signal	Value										
		0	10ns	20ns	30ns	40ns	50ns	60ns	70ns	80ns	90ns
a[3:0]	'h0	0		1			0	2	3	0	
b[3:0]	'h4	0	1	0	2			3		4	
cin_hi	0										
cout_hi	0										
cp	1										
d[3:0]	'h0	2	7								0
f3	1										
g_lo	1										
i[8:0]	'h0FE	0DF	0C5	029	190	08C	0F9	0F0	000	0AA	0FE
oe	1										
ovr	0										
p_lo	1										
q0	z										
q0reg	z										
q3	z										
q3reg	z										
ram0	z										
ram0reg	z										
ram3	z										
ram3reg	z										
y[3:0]	'hB	2	9	2	7	9	4	0	4	2	B
z	0										

===

Compare set name: c1 (hierarchical)

Databases:

```
  Logical   Database
  Name      Pathname                                                   Type
  ===============================================================================
  g          /class/ece425/mp2/datapath_gold/datapath_gold.trn         Golden
  t          /home/jrarndt2/ece425.work/datapath_run1/shm.db/shm.db.trn Test
```

Compare command:

  compare . –pos 2ns –neg 2ns –maxerrors 200

Comparison parameters:

```
  Total error limit:  100
  This error limit:   200
  Depth:              No limit
  Negative tolerance: 2ns
  Positive tolerance: 2ns
  Start time:         0ns
  End time:           18446744073.709551615s
```

Comparison summary:

```
  Successful compares:    20/24
  Expression miscompares: 12
```

===

Compare set name: c1:cds_globals.gnd_

Comparison parameters:

  Golden expression: g::cds_globals.gnd_
  Test expression: t::cds_globals.gnd_

Comparison summary:

  No problems found.

===

Compare set name: c1:cds_globals.vdd_

Comparison parameters:

  Golden expression: g::cds_globals.vdd_
  Test expression: t::cds_globals.vdd_

Comparison summary:

  No problems found.

===

Compare set name: c1:test.a[3:0]

Comparison parameters:

  Golden expression: g::test.a[3:0]
  Test expression: t::test.a[3:0]

Comparison summary:

  No problems found.

==

Compare set name: c1:test.b[3:0]

Comparison parameters:

  Golden expression: g::test.b[3:0]
  Test expression: t::test.b[3:0]

Comparison summary:

  No problems found.

==

Compare set name: c1:test.cin_hi

Comparison parameters:

  Golden expression: g::test.cin_hi
  Test expression: t::test.cin_hi

Comparison summary:

  No problems found.

==

Compare set name: c1:test.cout_hi

Comparison parameters:

  Golden expression: g::test.cout_hi
  Test expression: t::test.cout_hi

Comparison summary:

  Expression miscompares: 5

Miscompare details:

  Error Type   From   To   Golden/Test Values

  Miscompare   0ns   10ns   1
                               0
  Miscompare   21ns   30ns   1
                               0
  Miscompare   51ns   60ns   1
                               0
  Miscompare   80ns   90ns   0
                               1
  Miscompare   90ns   100ns   1
                               0

==

Compare set name: c1:test.cp

Comparison parameters:

  Golden expression: g::test.cp
  Test expression: t::test.cp

Comparison summary:

  No problems found.

==

Compare set name: c1:test.d[3:0]

Comparison parameters:

  Golden expression: g::test.d[3:0]
  Test expression: t::test.d[3:0]

Comparison summary:

  No problems found.

==

Compare set name: c1:test.f3

Comparison parameters:

  Golden expression: g::test.f3
  Test expression: t::test.f3

Comparison summary:

  No problems found.

==

Compare set name: c1:test.g_lo

Comparison parameters:

  Golden expression: g::test.g_lo
  Test expression: t::test.g_lo

Comparison summary:

  Expression miscompares: 5

Miscompare details:

Error Type	From	To	Golden/Test Values
Miscompare	0ns	10ns	0
			1
Miscompare	21ns	30ns	0
			1
Miscompare	51ns	60ns	0
			1
Miscompare	80ns	90ns	1
			0

```
  Miscompare   90ns    100ns    0
                                1
```

==

Compare set name: c1:test.i[8:0]

Comparison parameters:

  Golden expression: g::test.i[8:0]
  Test expression: t::test.i[8:0]

Comparison summary:

  No problems found.

==

Compare set name: c1:test.oe

Comparison parameters:

  Golden expression: g::test.oe
  Test expression: t::test.oe

Comparison summary:

  No problems found.

==

Compare set name: c1:test.ovr

Comparison parameters:

  Golden expression: g::test.ovr
  Test expression: t::test.ovr

Comparison summary:

  Expression miscompares: 1

Miscompare details:

```
  Error Type   From   To     Golden/Test Values
  ---------------------------------------------
  Miscompare   51ns   58ns   1
                             0
```

==

Compare set name: c1:test.p_lo

Comparison parameters:

  Golden expression: g::test.p_lo
  Test expression: t::test.p_lo

Comparison summary:

  Expression miscompares: 1

Miscompare details:

```
  Error Type    From    To     Golden/Test Values
  ----------------------------------------------
  Miscompare    50ns    51ns    0
                               1
```

===

Compare set name: c1:test.q0

Comparison parameters:

  Golden expression: g::test.q0
  Test expression: t::test.q0

Comparison summary:

  No problems found.

===

Compare set name: c1:test.q0reg

Comparison parameters:

  Golden expression: g::test.q0reg
  Test expression: t::test.q0reg

Comparison summary:

  No problems found.

===

Compare set name: c1:test.q3

Comparison parameters:

  Golden expression: g::test.q3
  Test expression: t::test.q3

Comparison summary:

  No problems found.

===

Compare set name: c1:test.q3reg

Comparison parameters:

  Golden expression: g::test.q3reg
  Test expression: t::test.q3reg

Comparison summary:

  No problems found.

===

Compare set name: c1:test.ram0

Comparison parameters:

  Golden expression: g::test.ram0
  Test expression: t::test.ram0

Comparison summary:

  No problems found.

==

Compare set name: c1:test.ram0reg

Comparison parameters:

  Golden expression: g::test.ram0reg
  Test expression: t::test.ram0reg

Comparison summary:

  No problems found.

==

Compare set name: c1:test.ram3

Comparison parameters:

  Golden expression: g::test.ram3
  Test expression: t::test.ram3

Comparison summary:

  No problems found.

==

Compare set name: c1:test.ram3reg

Comparison parameters:

  Golden expression: g::test.ram3reg
  Test expression: t::test.ram3reg

Comparison summary:

  No problems found.

==

Compare set name: c1:test.y[3:0]

Comparison parameters:

  Golden expression: g::test.y[3:0]
  Test expression: t::test.y[3:0]

Comparison summary:

  No problems found.

==

Compare set name: c1:test.z

Comparison parameters:

   Golden expression: g::test.z
   Test expression: t::test.z

Comparison summary:

   No problems found.