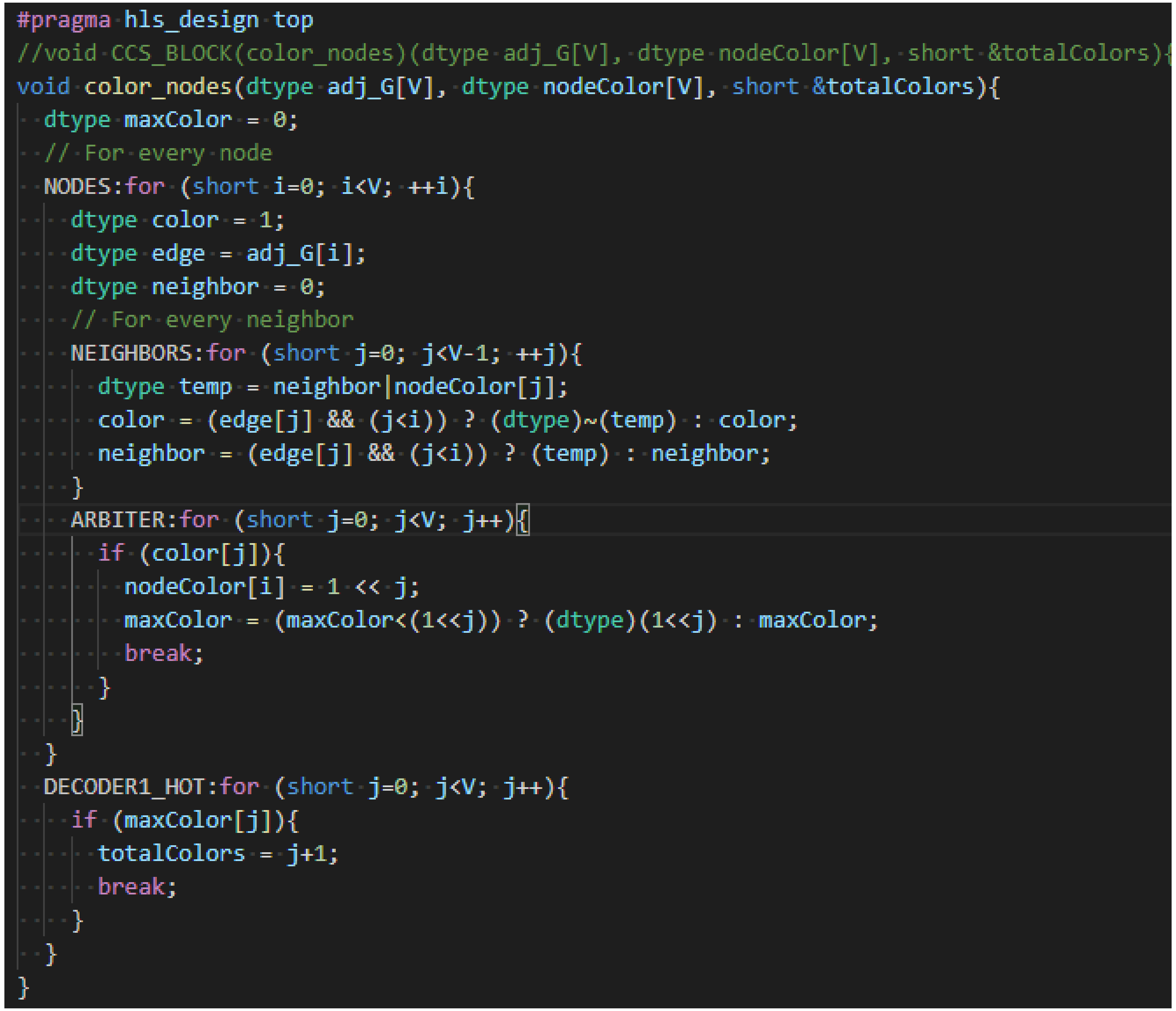
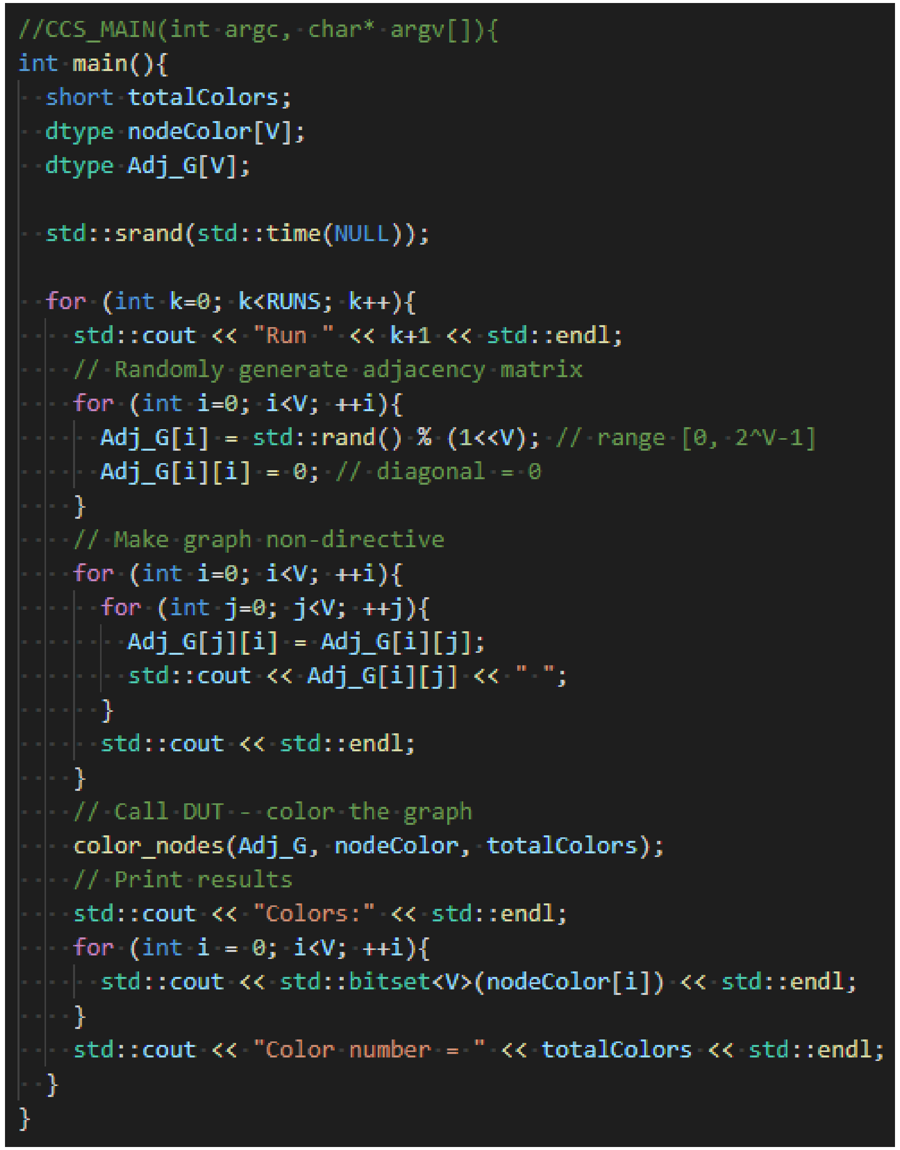
Catapult – Lab 3:
Color Nodes: Maximum Throughput

Margomenos Nikos

Algorithm Implementation & Code



```
#pragma hls_design top
//void CCS_BLOCK(color_nodes)(dtype adj_G[V], dtype nodeColor[V], short &totalColors){
void color_nodes(dtype adj_G[V], dtype nodeColor[V], short &totalColors){
  dtype maxColor = 0;
  // For every node
  NODES:for (short i=0; i<V; ++i){
    dtype color = 1;
    dtype edge = adj_G[i];
    dtype neighbor = 0;
    // For every neighbor
    NEIGHBORS:for (short j=0; j<V-1; ++j){
      dtype temp = neighbor|nodeColor[j];
      color = (edge[j] && (j<i)) ? (dtype)~(temp) : color;
      neighbor = (edge[j] && (j<i)) ? (temp) : neighbor;
    }
    ARBITER:for (short j=0; j<V; j++){
      if (color[j]){
        nodeColor[i] = 1 << j;
        maxColor = (maxColor<(1<<j)) ? (dtype)(1<<j) : maxColor;
        break;
      }
    }
  }
  DECODER1_HOT:for (short j=0; j<V; j++){
    if (maxColor[j]){
      totalColors = j+1;
      break;
    }
  }
}
```



```
//CCS_MAIN(int argc, char* argv[]){
int main(){
  short totalColors;
  dtype nodeColor[V];
  dtype Adj_G[V];

  std::srand(std::time(NULL));

  for (int k=0; k<RUNS; k++){
    std::cout << "Run " << k+1 << std::endl;
    // Randomly generate adjacency matrix
    for (int i=0; i<V; ++i){
      Adj_G[i] = std::rand() % (1<<V); // range [0, 2^V-1]
      Adj_G[i][i] = 0; // diagonal = 0
    }
    // Make graph non-directive
    for (int i=0; i<V; ++i){
      for (int j=0; j<V; ++j){
        Adj_G[j][i] = Adj_G[i][j];
        std::cout << Adj_G[i][j] << " ";
      }
      std::cout << std::endl;
    }
    // Call DUT - color the graph
    color_nodes(Adj_G, nodeColor, totalColors);
    // Print results
    std::cout << "Colors:" << std::endl;
    for (int i = 0; i<V; ++i){
      std::cout << std::bitset<V>(nodeColor[i]) << std::endl;
    }
    std::cout << "Color number = " << totalColors << std::endl;
  }
}
```

Maximum Throughput with Unroll & Pipeline

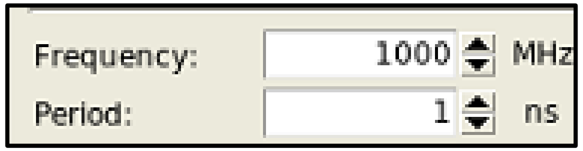
Combinational logic procedures are **unrolled** in order to be executed within a single cycle:

- ARBITER
- DECODER1_HOT

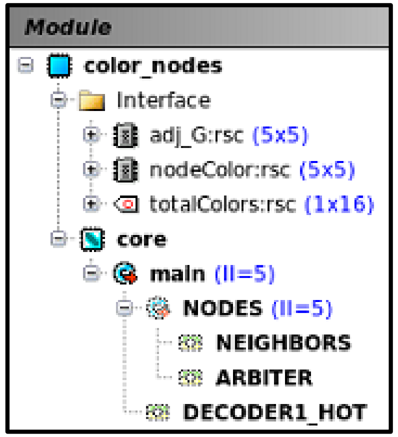
NEIGHBORS loop is **unrolled** to create a deeper but more efficient main pipeline with less wasted stages (**II=5**) :

- Not unrolling the loop would create a pipeline with II=2, but that would need 8 cycles per array row (**3 wasted cycles**)

For a 5x5 array, this scheduling requires 5 cycles per array row (**25 cycles**)

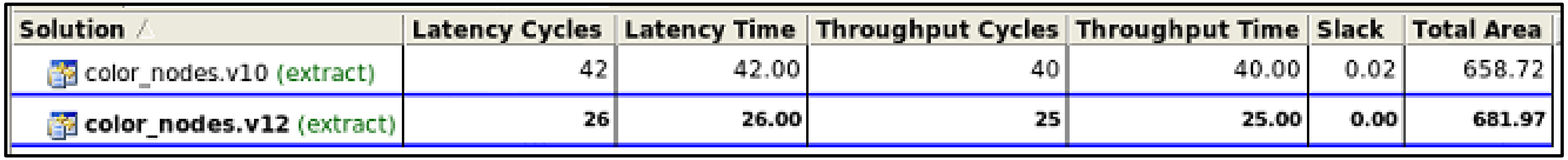


Frequency: 1000 MHz
Period: 1 ns

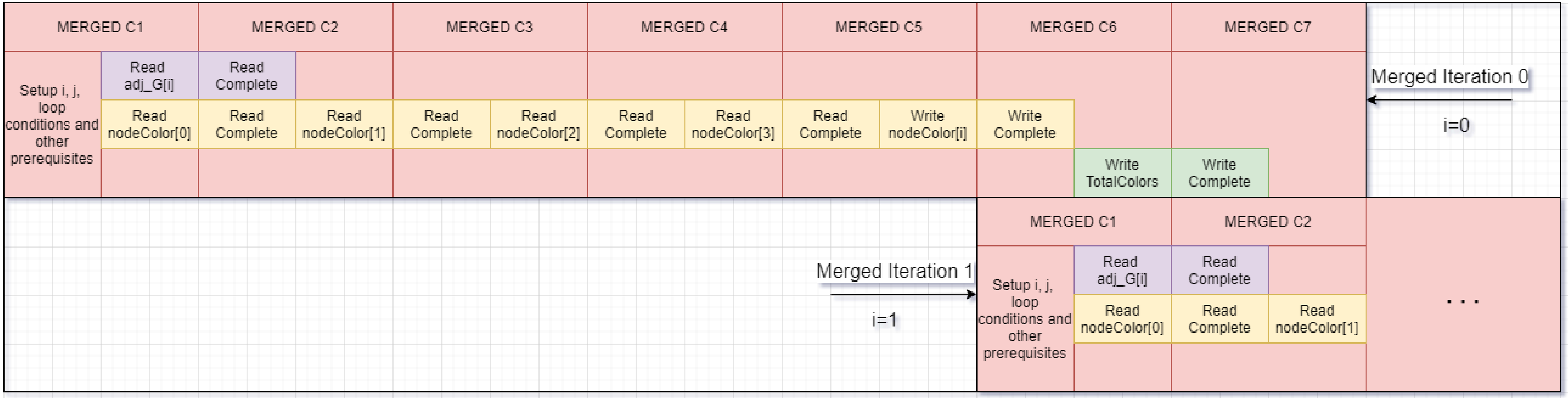


Module

- color_nodes
  - Interface
    - adj_G:rsc (5x5)
    - nodeColor:rsc (5x5)
    - totalColors:rsc (1x16)
  - core
    - main (II=5)
      - NODES (II=5)
        - NEIGHBORS
        - ARBITER
      - DECODER1_HOT



Solution	Latency Cycles	Latency Time	Throughput Cycles	Throughput Time	Slack	Total Area
color_nodes.v10 (extract)	42	42.00	40	40.00	0.02	658.72
color_nodes.v12 (extract)	**26**	**26.00**	**25**	**25.00**	**0.00**	**681.97**



MERGED C1 | MERGED C2 | MERGED C3 | MERGED C4 | MERGED C5 | MERGED C6 | MERGED C7

Setup i, j, loop conditions and other prerequisites

Read adj_G[i] → Read Complete

Read nodeColor[0] → Read Complete → Read nodeColor[1] → Read Complete → Read nodeColor[2] → Read Complete → Read nodeColor[3] → Read Complete → Write nodeColor[i] → Write Complete

Write TotalColors → Write Complete

Merged Iteration 0

i=0

Merged Iteration 1

i=1

MERGED C1 | MERGED C2

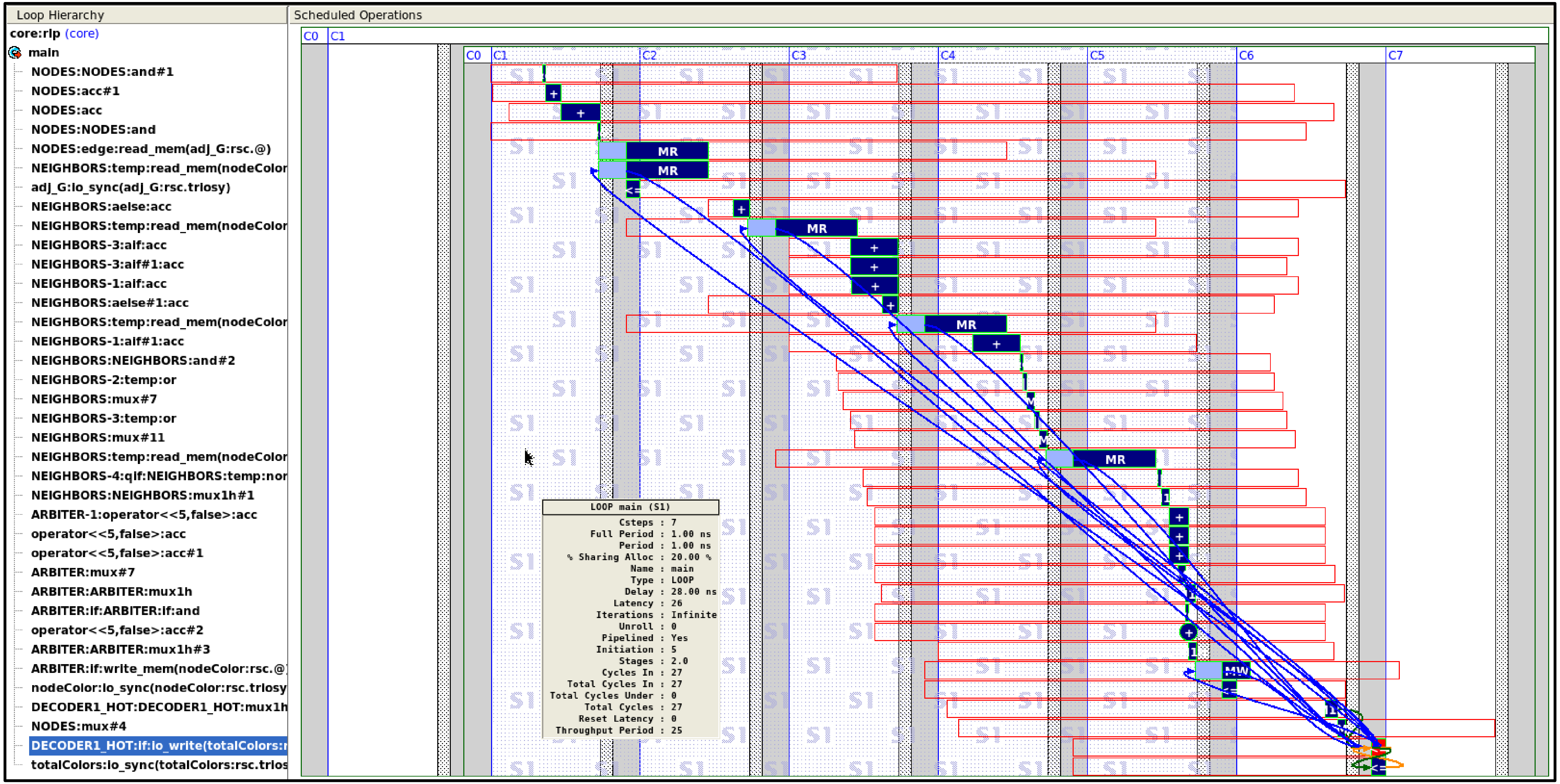
Setup i, j, loop conditions and other prerequisites

Read adj_G[i] → Read Complete

Read nodeColor[0] → Read Complete → Read nodeColor[1]

. . .

Maximum Throughput with Unroll & Pipeline



Loop Hierarchy

core:rlp (core)

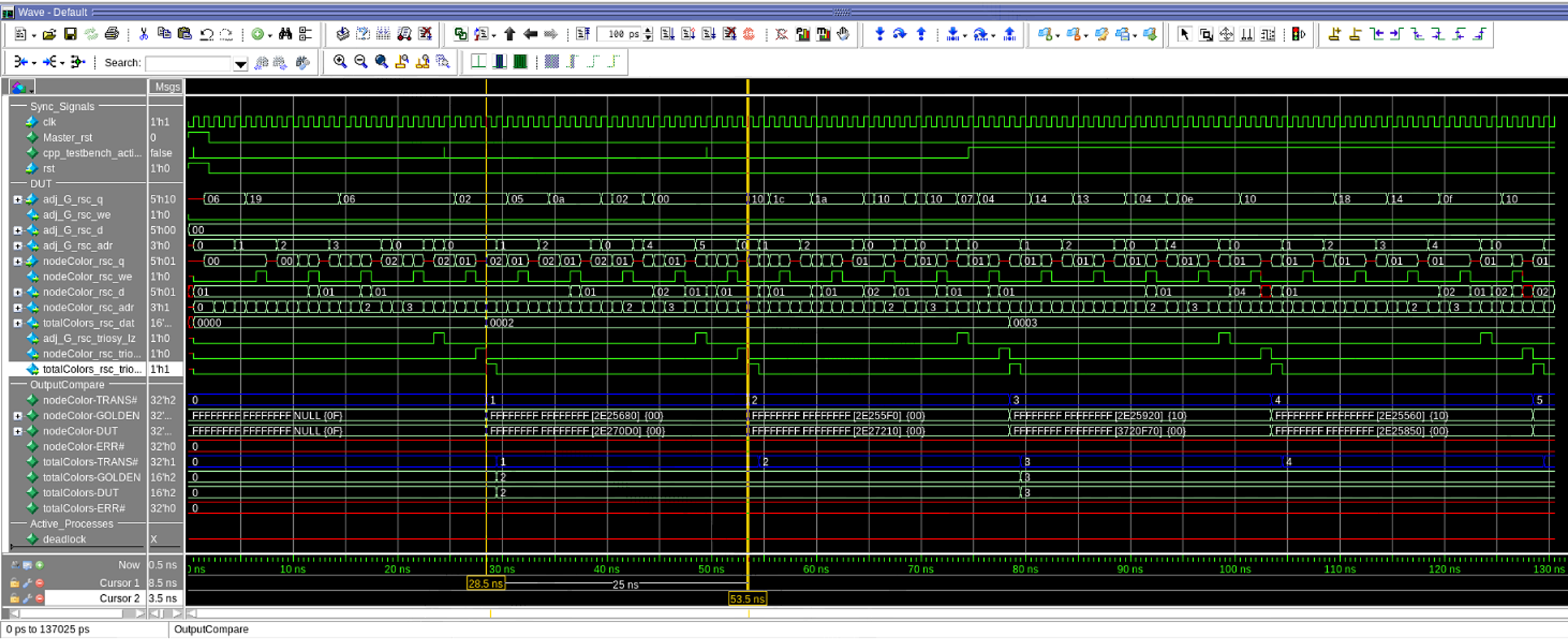
main

- NODES:NODES:and#1
- NODES:acc#1
- NODES:acc
- NODES:NODES:and
- NODES:edge:read_mem(adj_G:rsc.@)
- NEIGHBORS:temp:read_mem(nodeColor
- adj_G:io_sync(adj_G:rsc.triosy)
- NEIGHBORS:aelse:acc
- NEIGHBORS:temp:read_mem(nodeColor
- NEIGHBORS-3:aif:acc
- NEIGHBORS-3:aif#1:acc
- NEIGHBORS-1:aif:acc
- NEIGHBORS:aelse#1:acc
- NEIGHBORS:temp:read_mem(nodeColor
- NEIGHBORS-1:aif#1:acc
- NEIGHBORS:NEIGHBORS:and#2
- NEIGHBORS-2:temp:or
- NEIGHBORS:mux#7
- NEIGHBORS-3:temp:or
- NEIGHBORS:mux#11
- NEIGHBORS:temp:read_mem(nodeColor
- NEIGHBORS-4:qif:NEIGHBORS:temp:nor
- NEIGHBORS:NEIGHBORS:mux1h#1
- ARBITER-1:operator<<5,false>:acc
- operator<<5,false>:acc
- operator<<5,false>:acc#1
- ARBITER:mux#7
- ARBITER:ARBITER:mux1h
- ARBITER:if:ARBITER:if:and
- operator<<5,false>:acc#2
- ARBITER:ARBITER:mux1h#3
- ARBITER:if:write_mem(nodeColor:rsc.@)
- nodeColor:io_sync(nodeColor:rsc.triosy
- DECODER1_HOT:DECODER1_HOT:mux1h
- NODES:mux#4
- DECODER1_HOT:if:io_write(totalColors:r
- totalColors:io_sync(totalColors:rsc.trios

Scheduled Operations

```
        LOOP main (S1)
              Csteps : 7
         Full Period : 1.00 ns
              Period : 1.00 ns
    % Sharing Alloc : 20.00 %
                Name : main
                Type : LOOP
               Delay : 28.00 ns
             Latency : 26
          Iterations : Infinite
              Unroll : 0
           Pipelined : Yes
          Initiation : 5
              Stages : 2.0
           Cycles In : 27
     Total Cycles In : 27
  Total Cycles Under : 0
        Total Cycles : 27
       Reset Latency : 0
   Throughput Period : 25
```

Questa Sim: Simulation & Results



Wave - Default

Search:

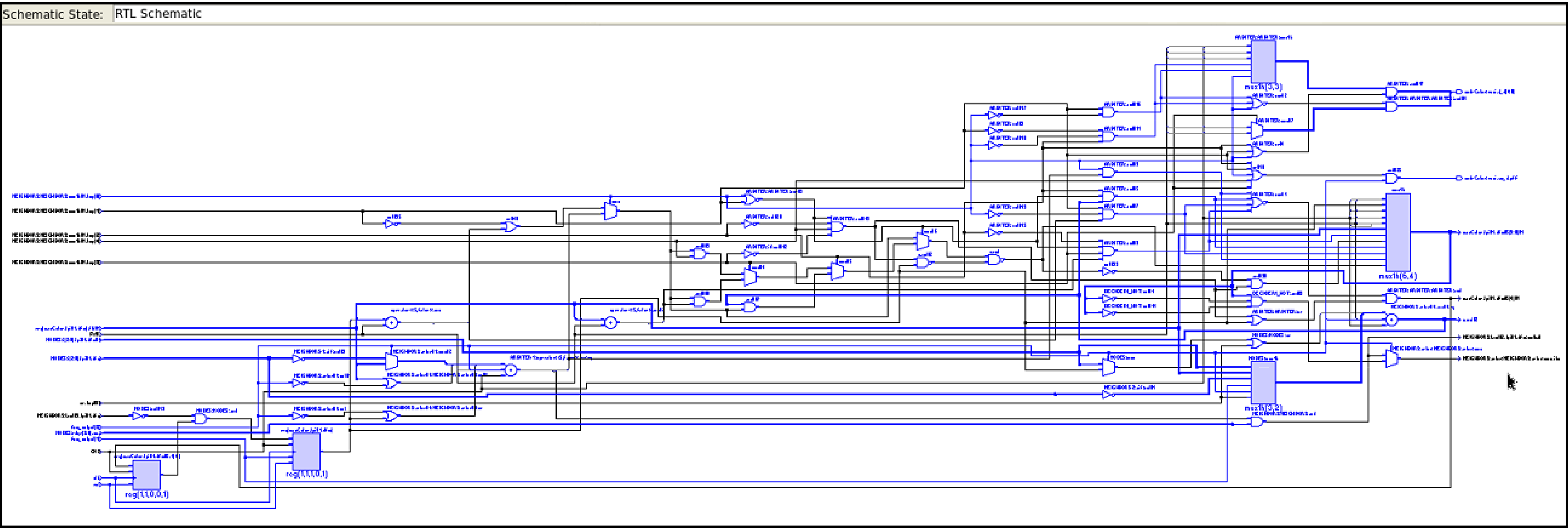
	Msgs
Sync_Signals	
clk	1'h1
Master_rst	0
cpp_testbench_acti...	false
rst	1'h0
DUT	
adj_G_rsc_q	5'h10
adj_G_rsc_we	1'h0
adj_G_rsc_d	5'h00
adj_G_rsc_adr	3'h0
nodeColor_rsc_q	5'h01
nodeColor_rsc_we	1'h0
nodeColor_rsc_d	5'h01
nodeColor_rsc_adr	3'h1
totalColors_rsc_dat	16'...
adj_G_rsc_triosy_lz	1'h0
nodeColor_rsc_trio...	1'h0
totalColors_rsc_trio...	1'h1
OutputCompare	
nodeColor-TRANS#	32'h2
nodeColor-GOLDEN	32'...
nodeColor-DUT	32'...
nodeColor-ERR#	32'h0
totalColors-TRANS#	32'h1
totalColors-GOLDEN	16'h2
totalColors-DUT	16'h2
totalColors-ERR#	32'h0
Active_Processes	
deadlock	X

Now	0.5 ns
Cursor 1	8.5 ns
Cursor 2	3.5 ns

0 ps to 137025 ps | OutputCompare

100 ps
06 19 06 02 05 0a 02 00 10 1c 1a 10 10 07 04 14 13 04 0e 10 18 14 0f 10
00
0000
0002
0003
0 1 2 3 4 5
FFFFFFFF FFFFFFFF NULL {0F}
FFFFFFFF FFFFFFFF [2E25680] {00}
FFFFFFFF FFFFFFFF [2E255F0] {00}
FFFFFFFF FFFFFFFF [2E25920] {10}
FFFFFFFF FFFFFFFF [2E25560] {10}
FFFFFFFF FFFFFFFF [2E270D0] {00}
FFFFFFFF FFFFFFFF [2E27210] {00}
FFFFFFFF FFFFFFFF [3720F70] {00}
FFFFFFFF FFFFFFFF [2E25850] {00}
0 ns
10 ns
20 ns
30 ns
40 ns
50 ns
60 ns
70 ns
80 ns
90 ns
100 ns
110 ns
120 ns
130 ns
28.5 ns
25 ns
53.5 ns

Improved Design – Main Pipeline



Schematic State: RTL Schematic