Mastermind Game

Hilary, Ricky, and Alexa

Group E

Functional Description

- Key and guess
- Scoring
- Guess counter
- Low score (no. of guesses)
- MUXing of output
- Resets: game vs. lowscore

Example Game

Key	G1	G2	G3
G	Y	B	G
B	R	G	B
B	G	B	B
B	B	B	B
	1 Black 1 White	2 Black 2 White	Win

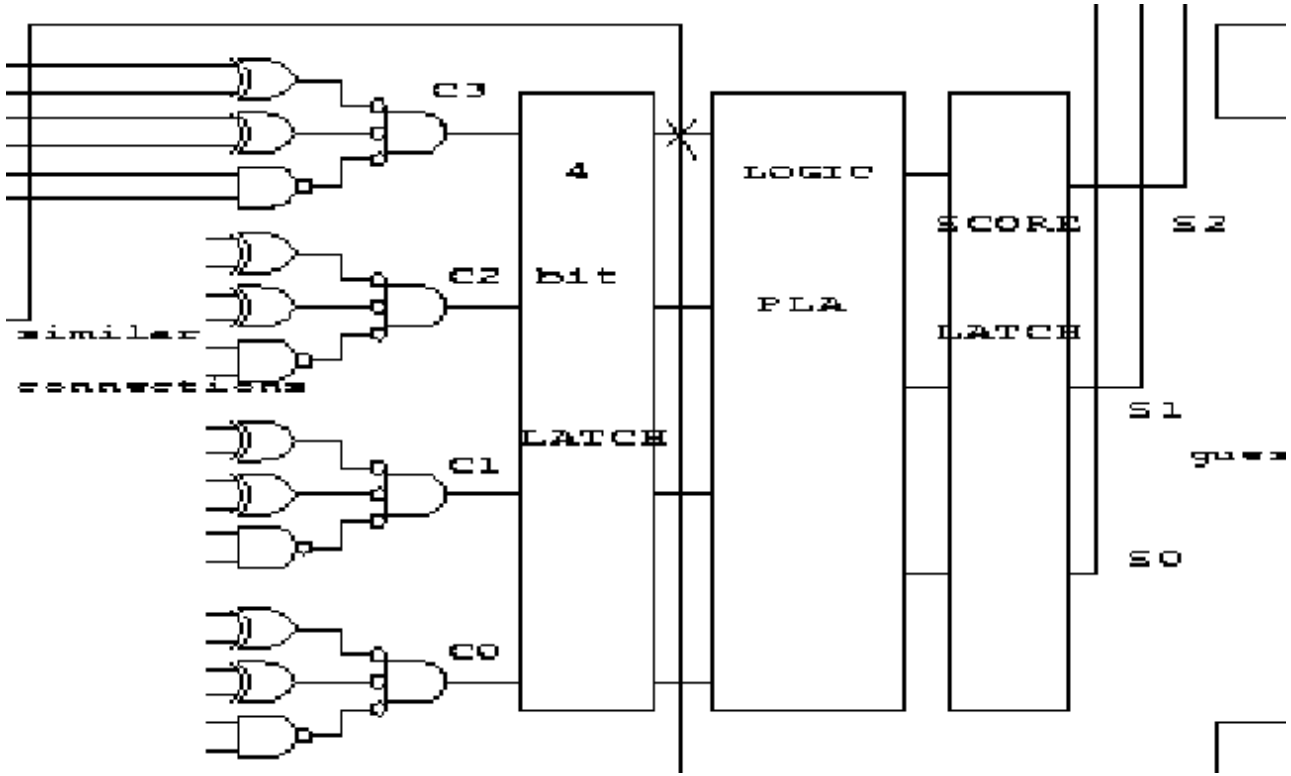
Algorithm

- Input latches and valid bits
- Valid-bit feedback calculation
- Comparison logic (decoder PLA)
- Scores to white adder, black latch, and FSM (win?)
- If win assert win bits & restart
- Else keep guessing (7 tries)

Logic Diagrams

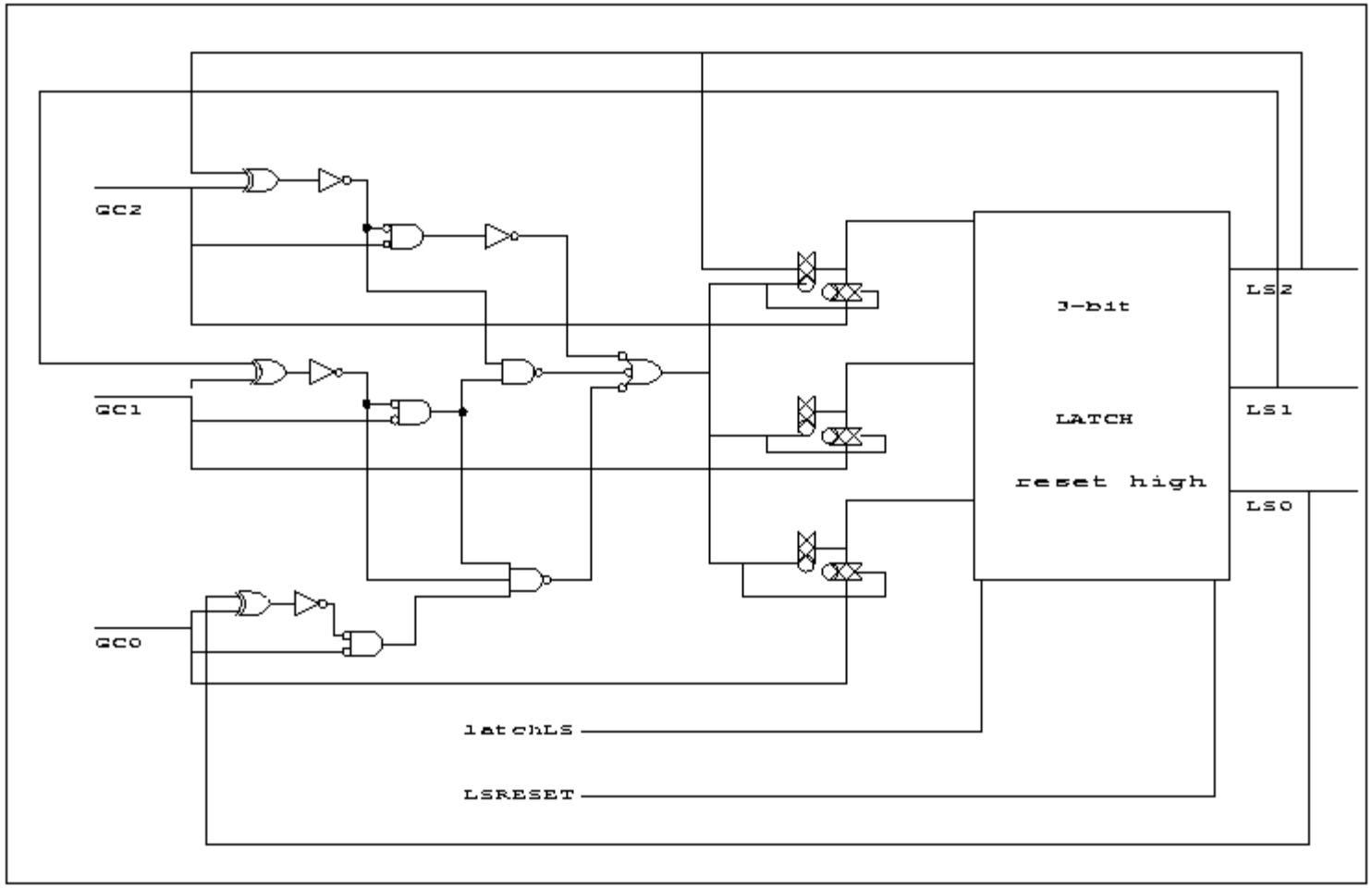
- **Comparison logic**
- **Low score logic**
- **Ripple carry adder**

Comparison Logic

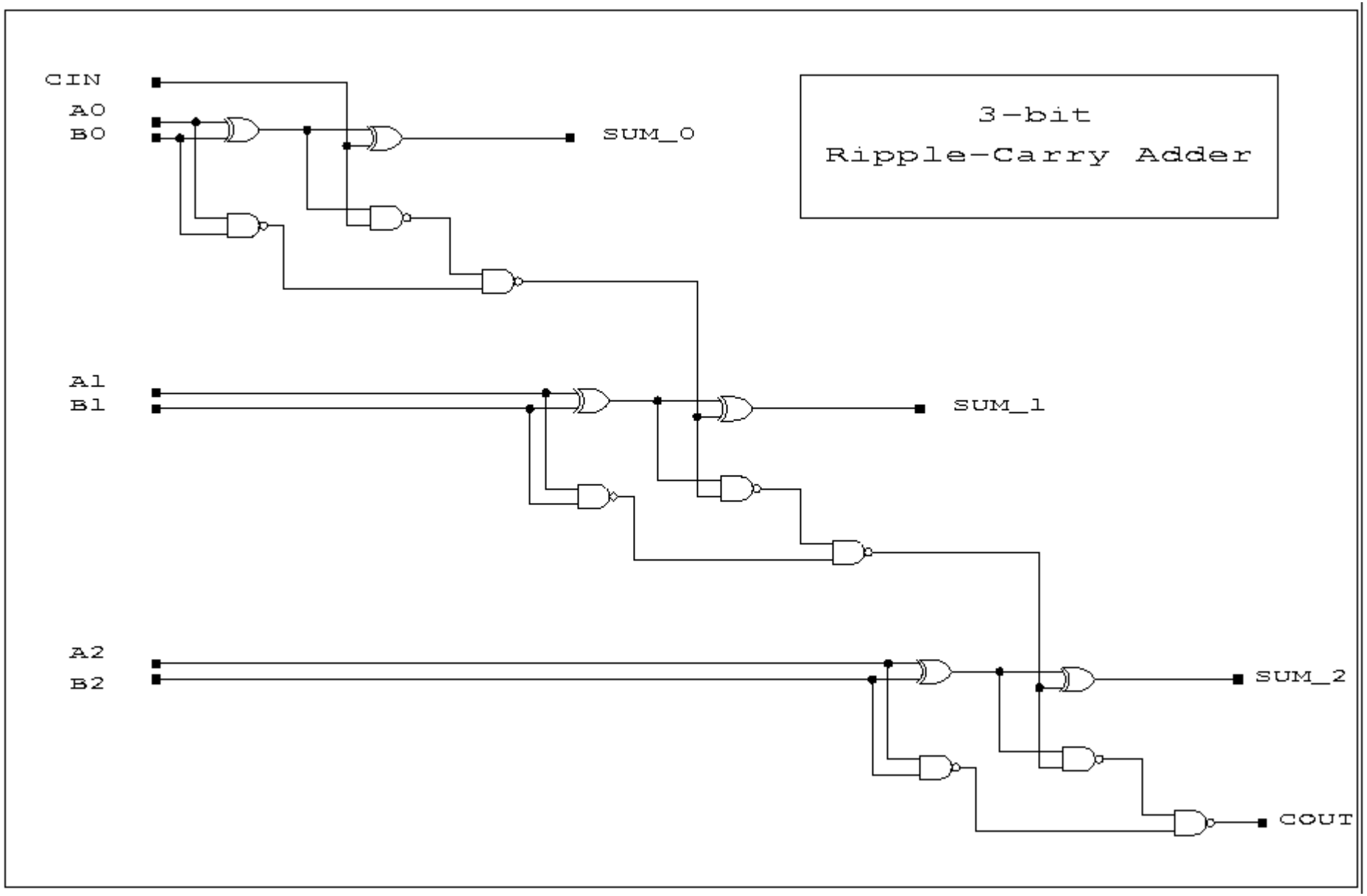


Logic PLA returns a binary count of the number of correct matches

Low Score Logic



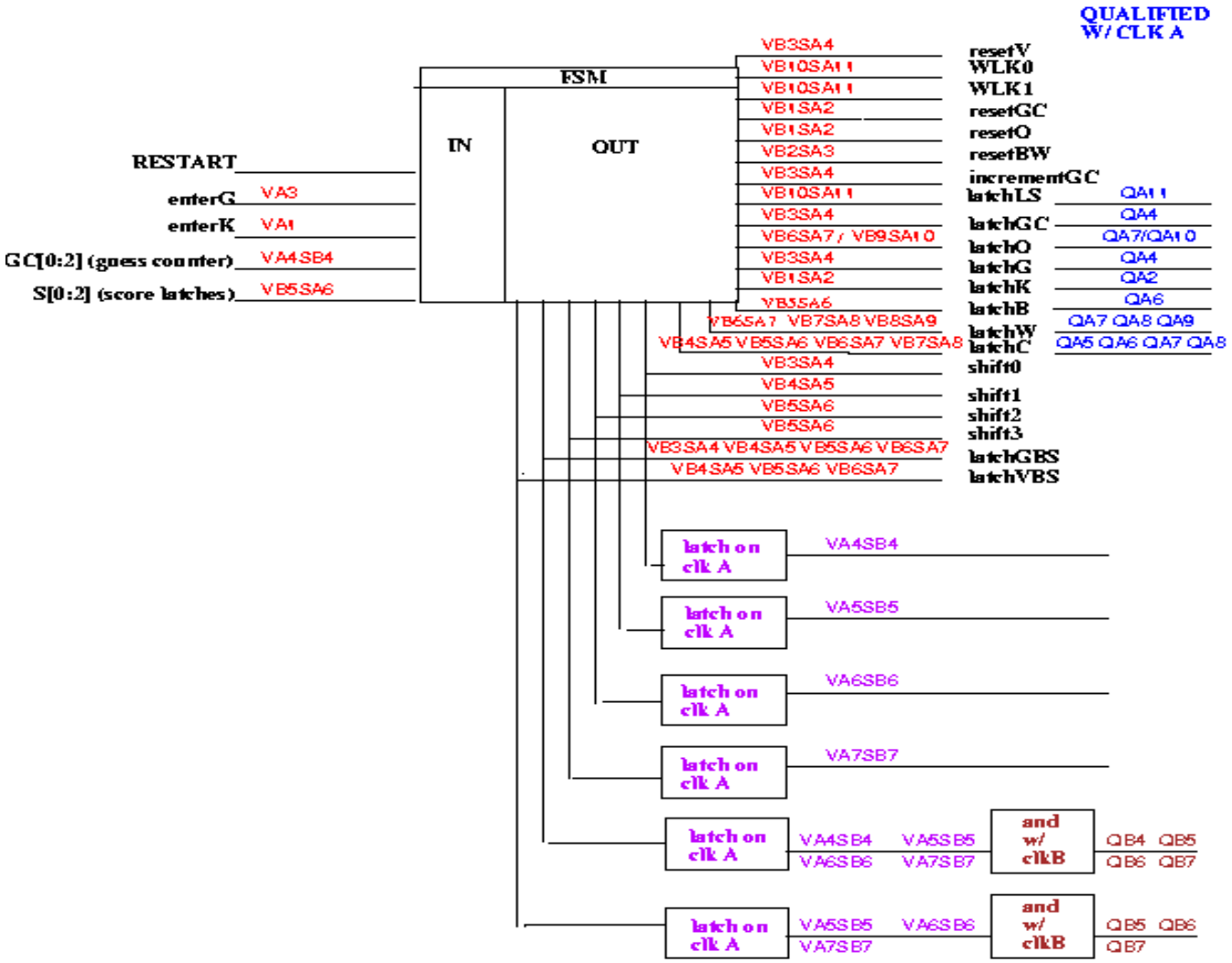
Ripple-Carry Adder



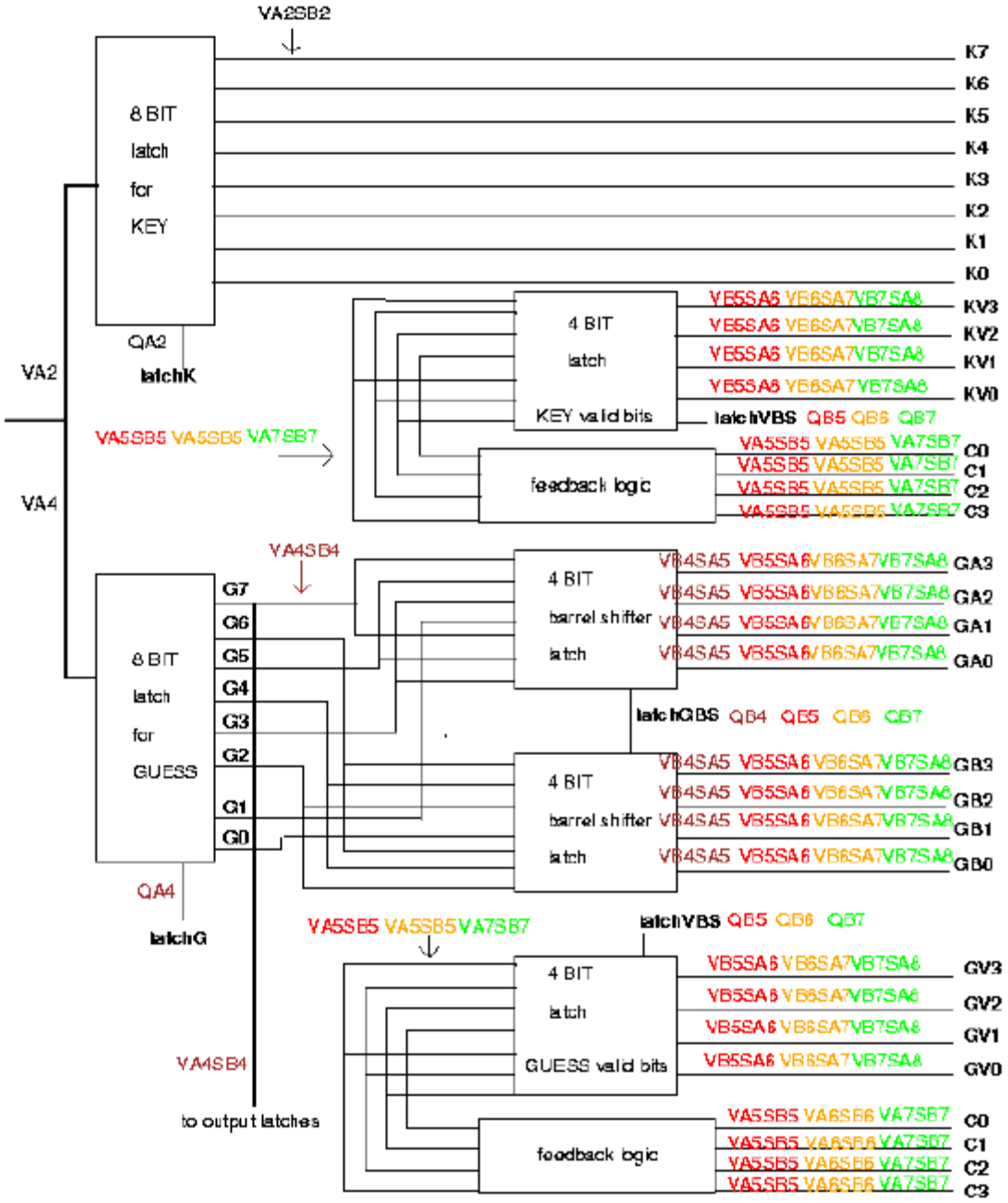
System Timing

- **Optimize flow using clka & clkb**
- **Latch control signals for clkb**
- **Valid bit feedback**
- **Latch feedbacks**

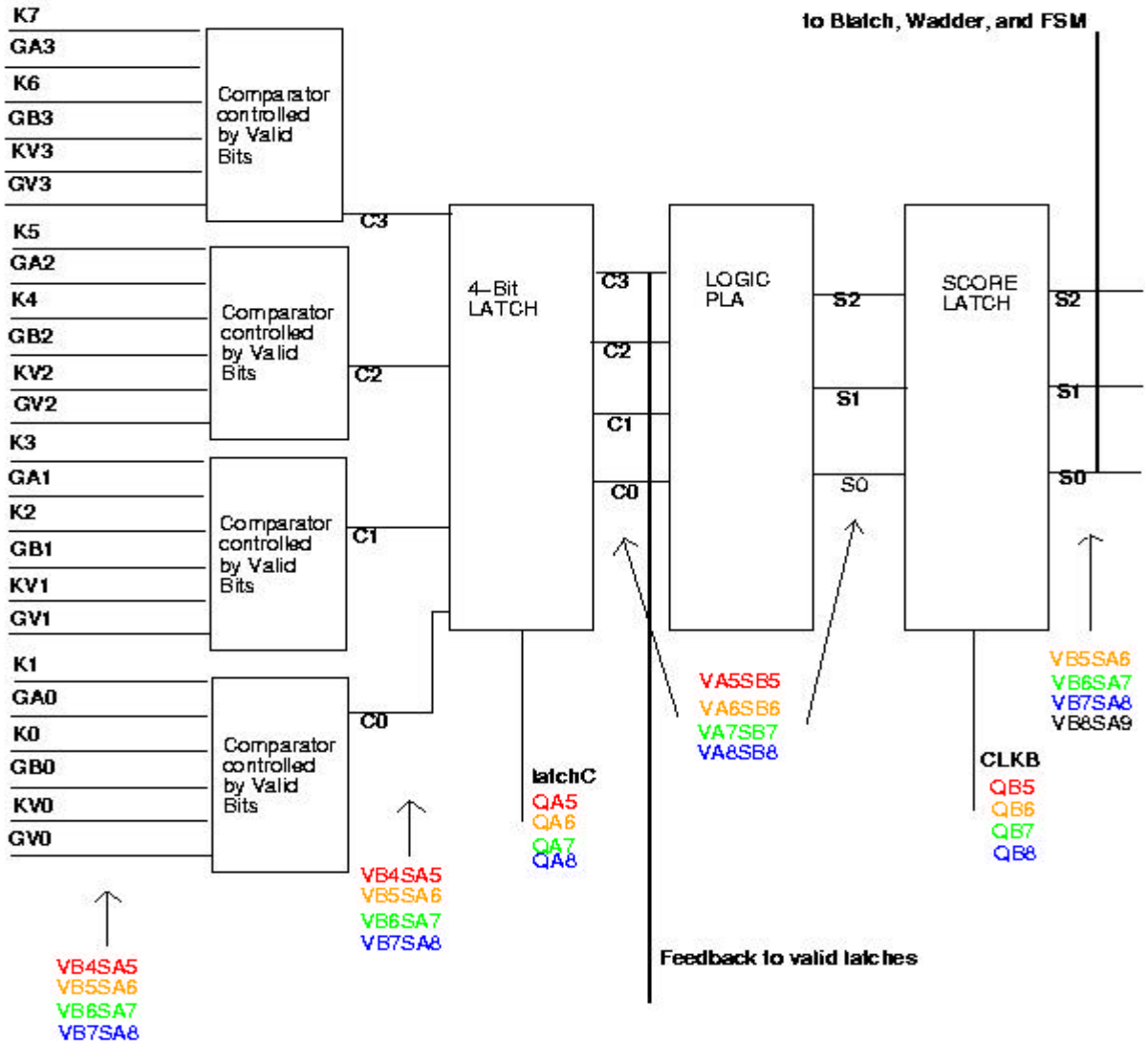
FSM Timing Diagram



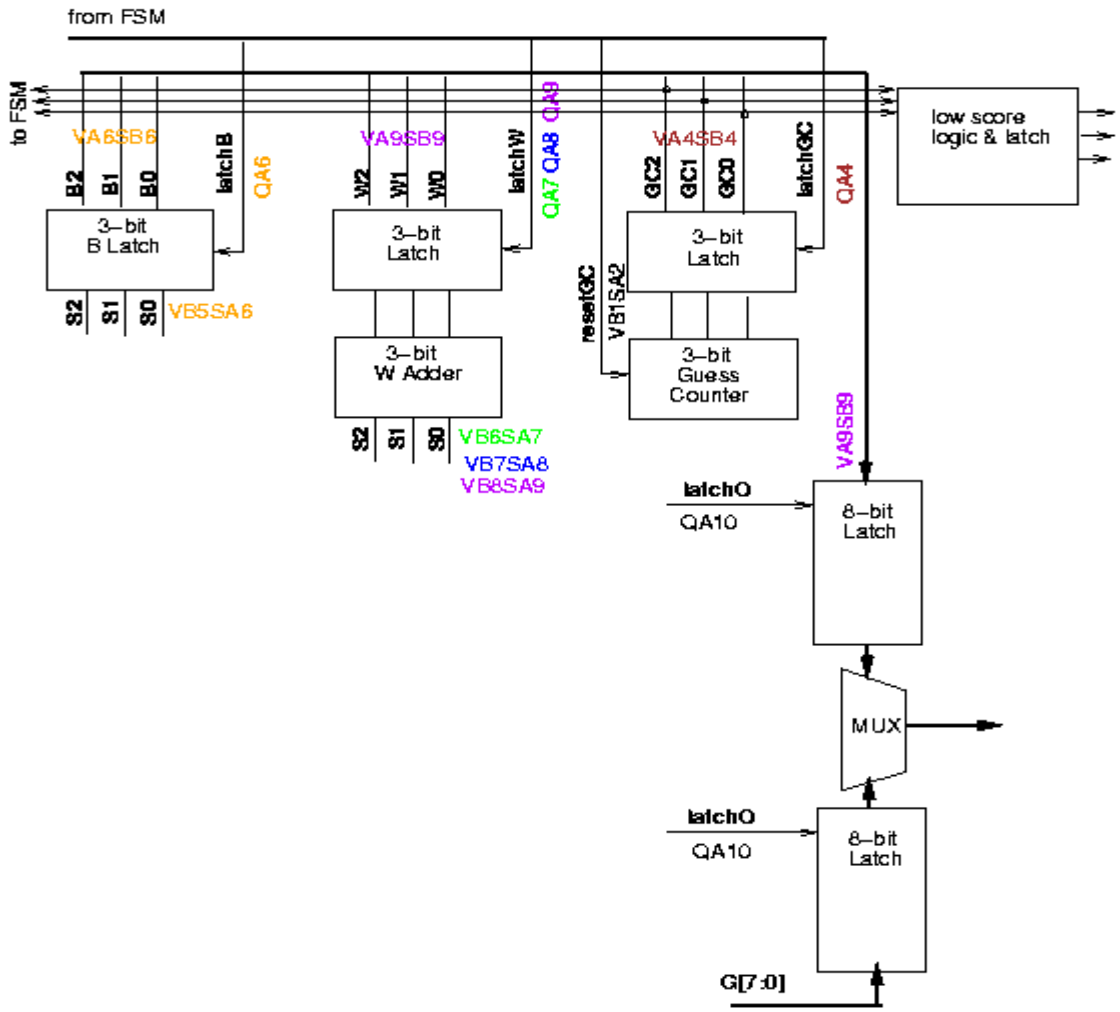
Input Timing Diagram



Logic Timing Diagram



Output Timing Diagram



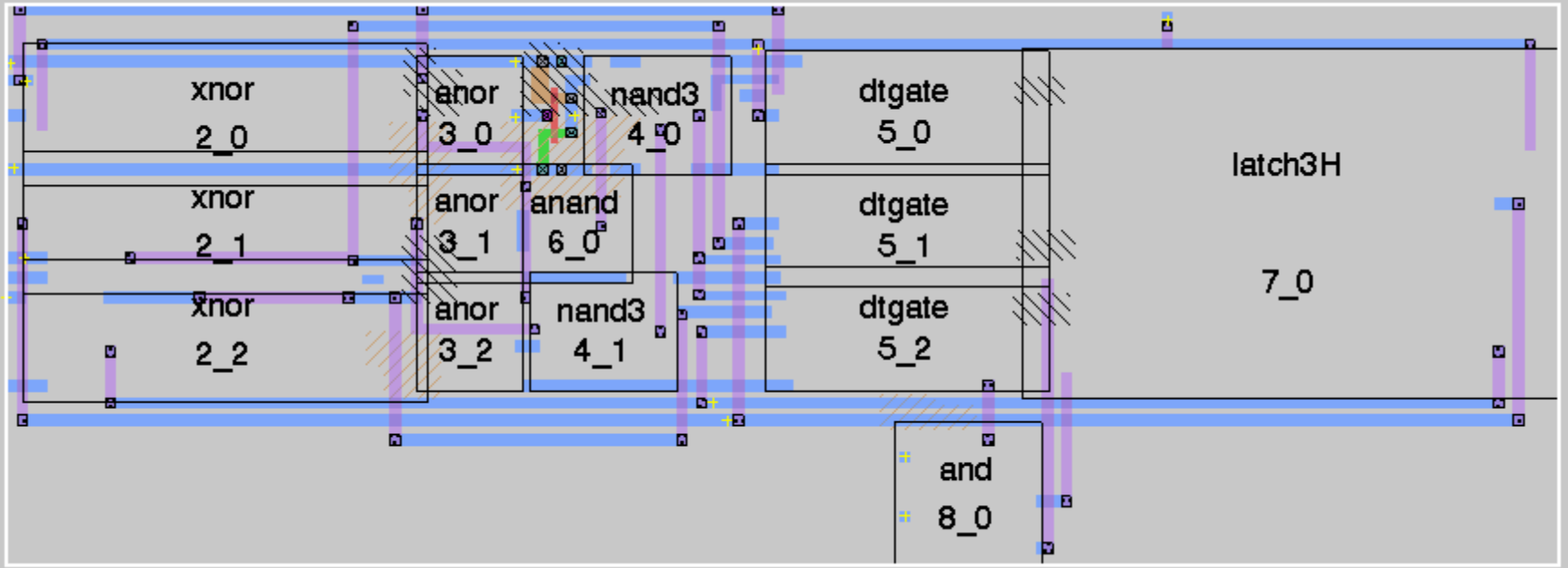
FSM States

```
--MEG program
--state transition on every clock

INPUTS: S2 S1 S0 RESTART enterG enterK GC2 GC1
GC0;

OUTPUTS: latchG shift0 shift1 shift2 shift3 latchGBS
resetV latchVBS latchK WLK1 WLK0 latchGC resetGC
incrementGC latchLS latchB resetBW latchW resetO latchO
latchC;

RESET ON RESTART TO A;

A: IF enterK THEN B(resetO resetGC latchK)
    ELSE LOOP;

B: GOTO C(resetBW);

C: IF enterG THEN D(latchG incrementGC latchGC shift0
latchGBS resetV)
   ELSE LOOP;

D: GOTO E(latchC shift1 latchVBS latchGBS);

E: GOTO F(latchB latchC shift2 latchGBS latchVBS);

F: CASE (S0 S1 S2)
    0 0 1 => K(latchO);
    ENDCASE => G(latchW latchC shift3 latchGBS
latchVBS);

G: GOTO H(latchW latchC);

H: GOTO I(latchW);

I: CASE (GC0 GC1 GC2)
  1 1 1 => J(latchO);
  ENDCASE => B(latchO);

J: GOTO A(WLK0);

K: GOTO A(WLK1 WLK0 latchLS);
```

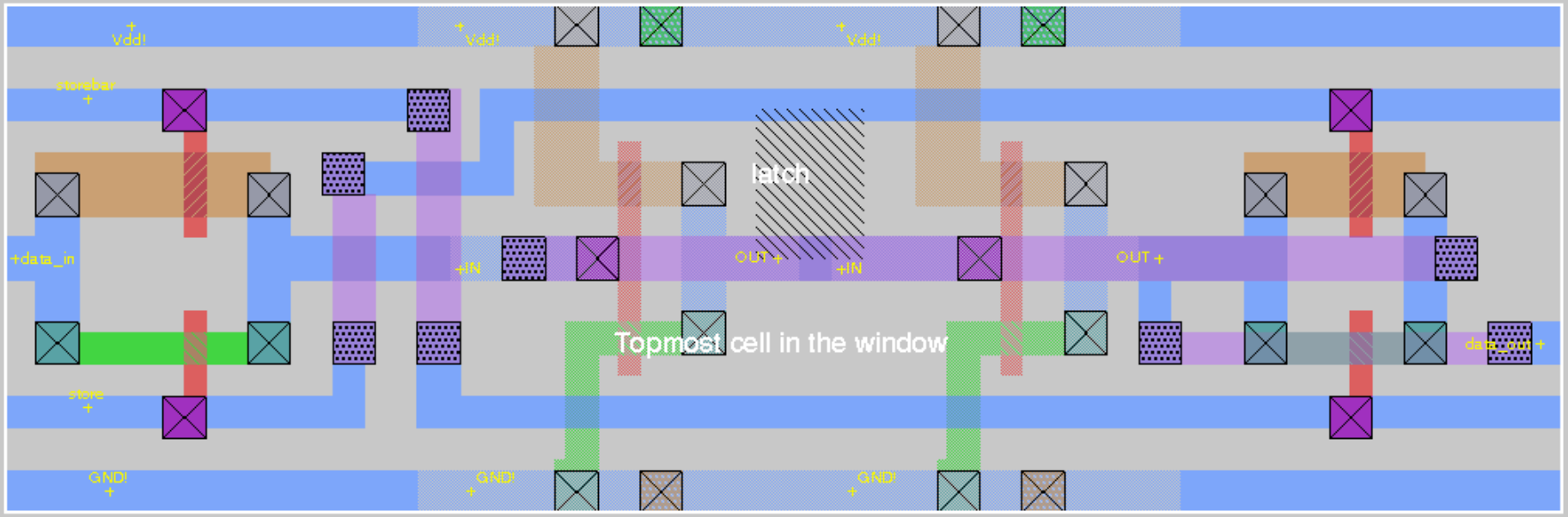
Circuit Layout

- **Cell hierarchy**
- **Sub-cell layout**
- **Floor plan**
- **Chip layout**

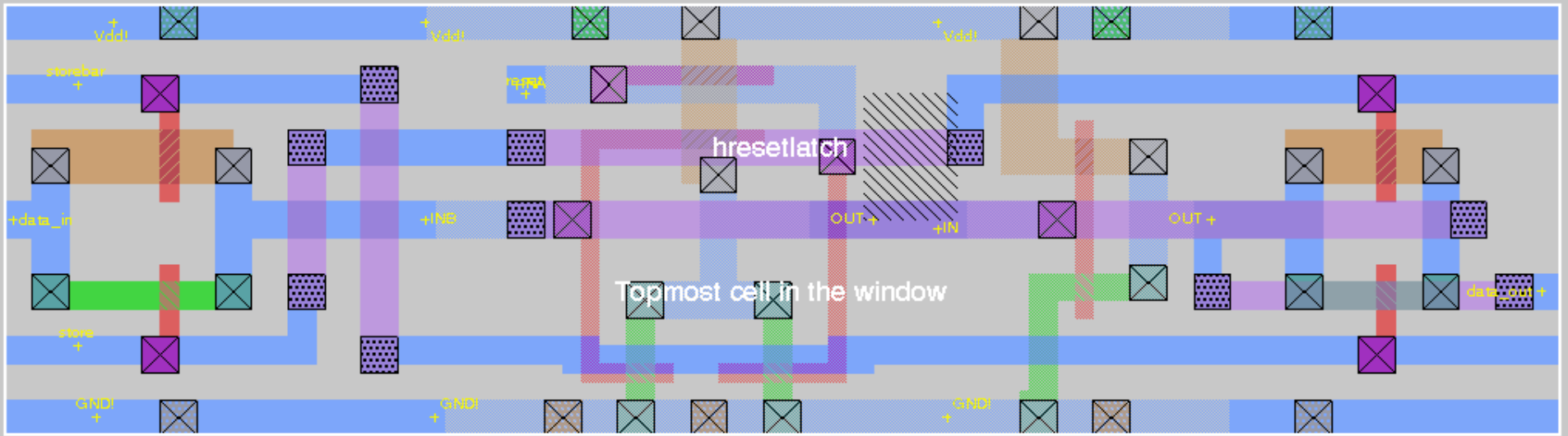
Low Score Logic--Cell Hierarchy



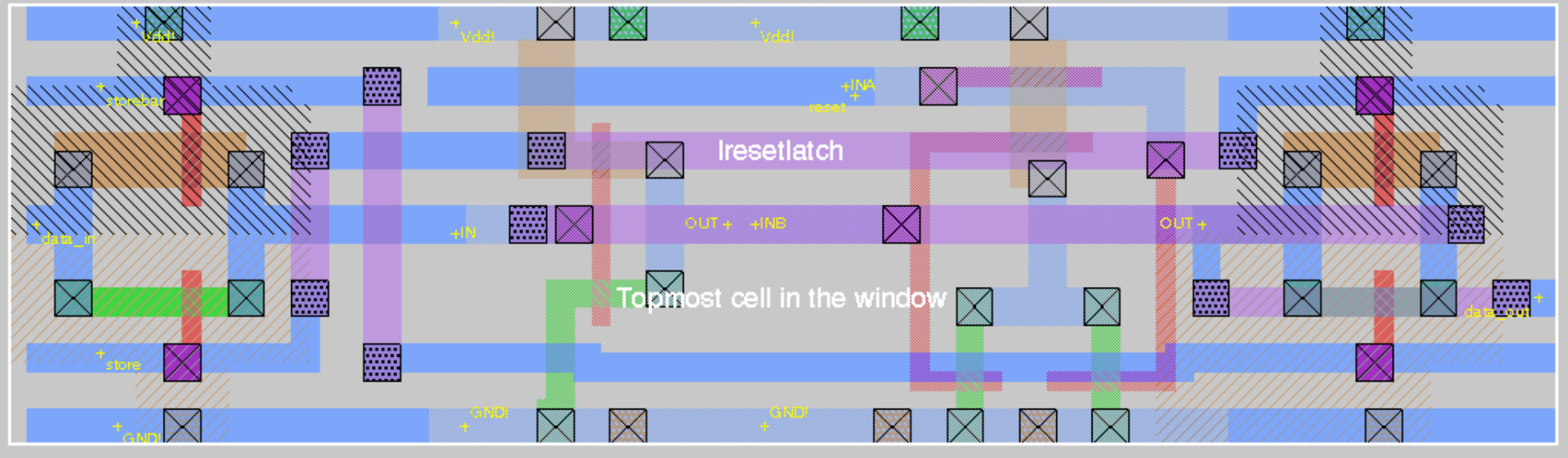
Latch


Vdd!
Vdd!
Vdd!
storebar
latch
+data_in
+IN
OUT +
+IN
OUT +
Topmost cell in the window
store
GND!
GND!
GND!

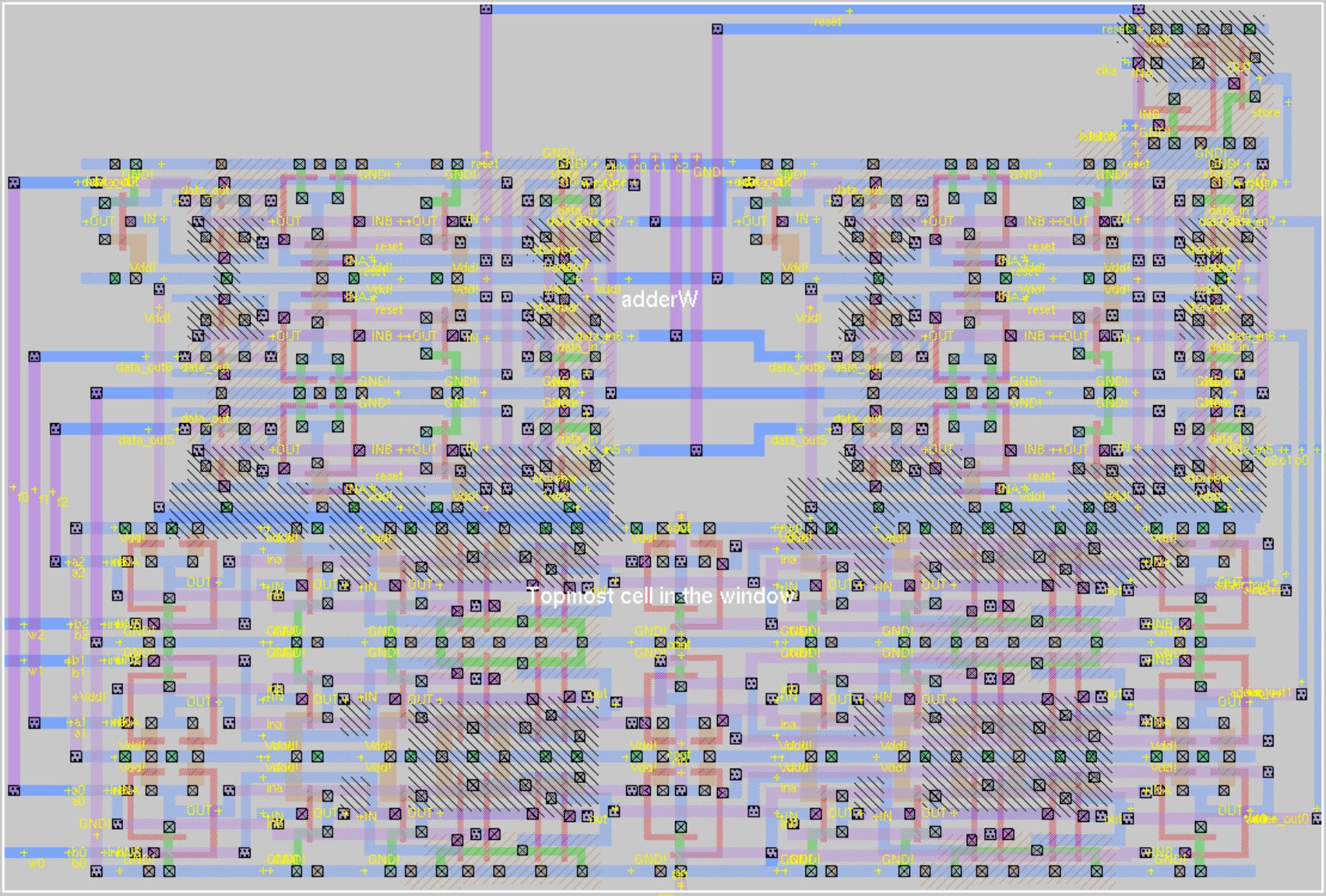
Latch w/ Reset High



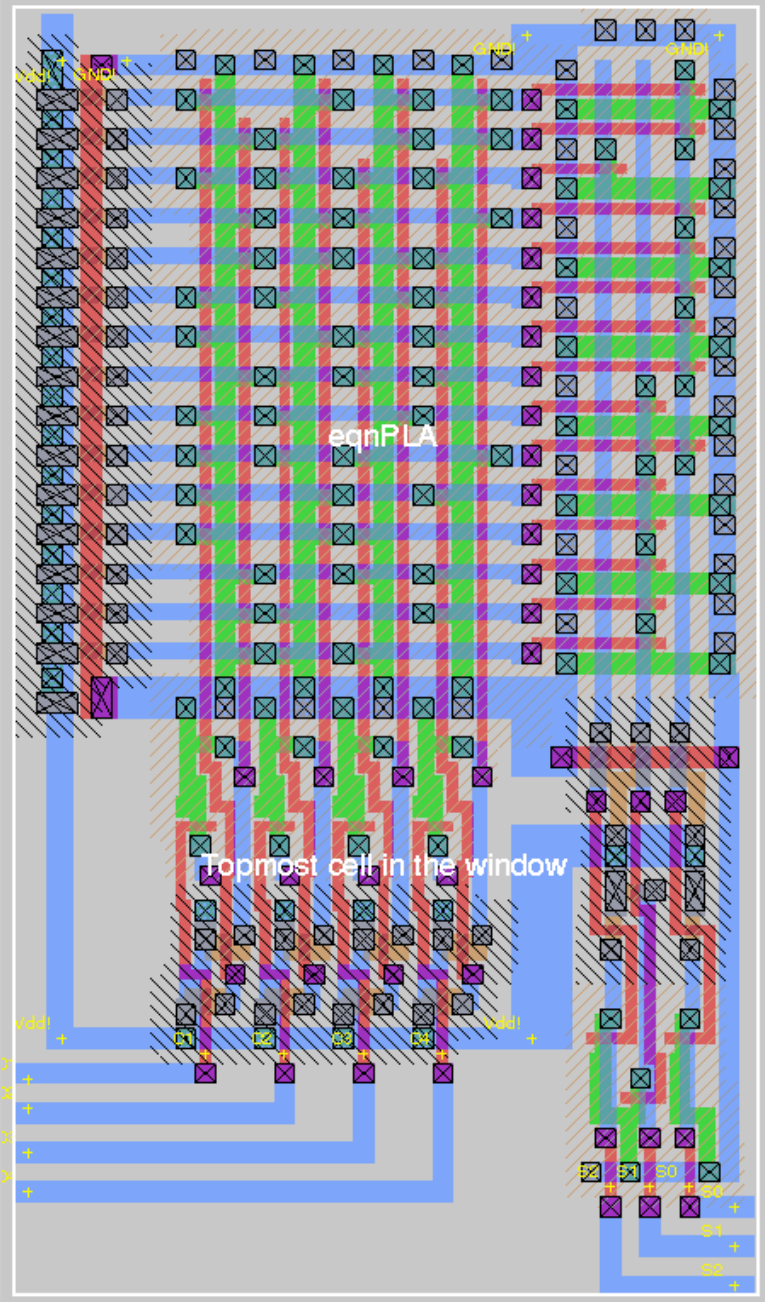
Latch w/ Reset Low



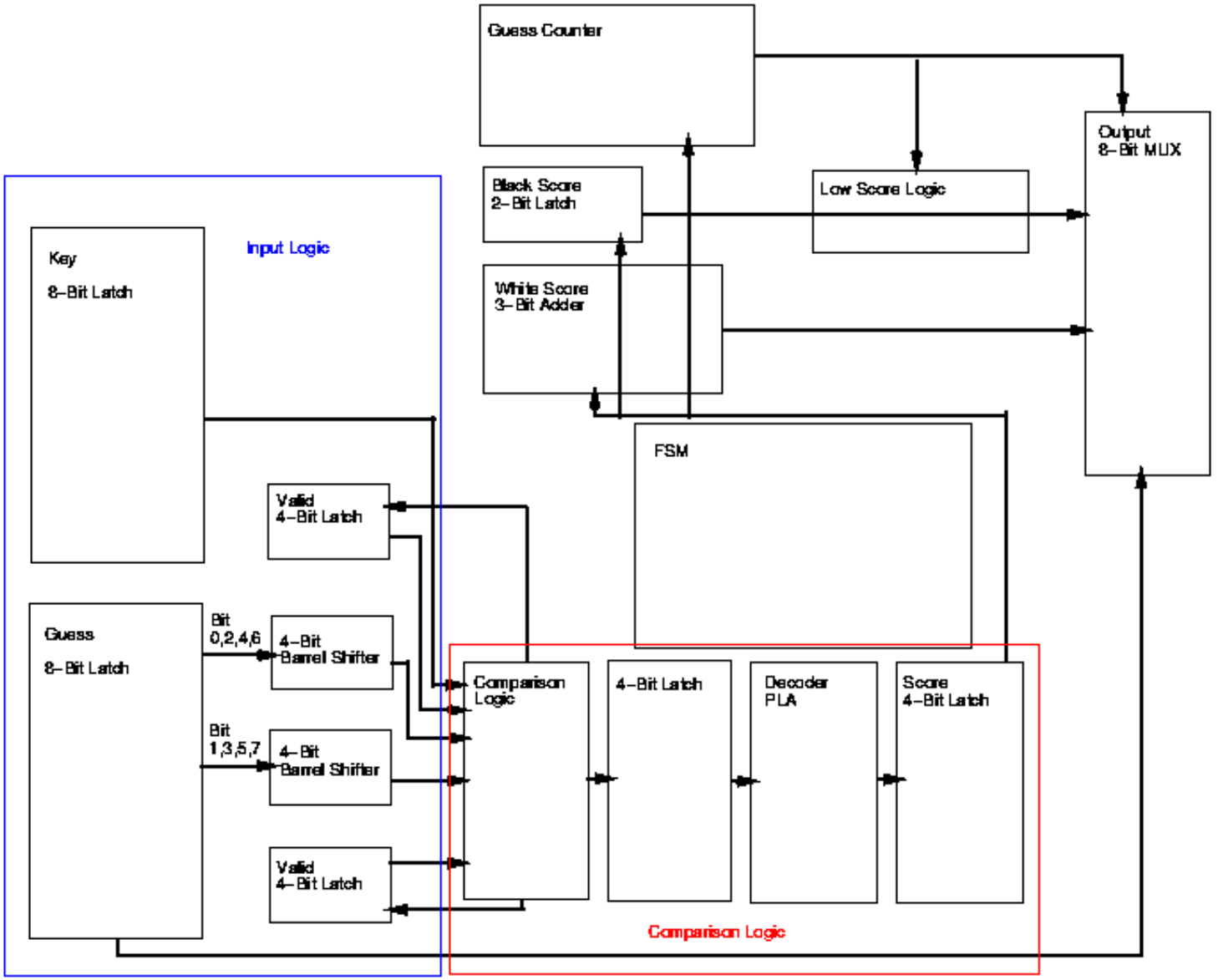
Ripple Carry Adder

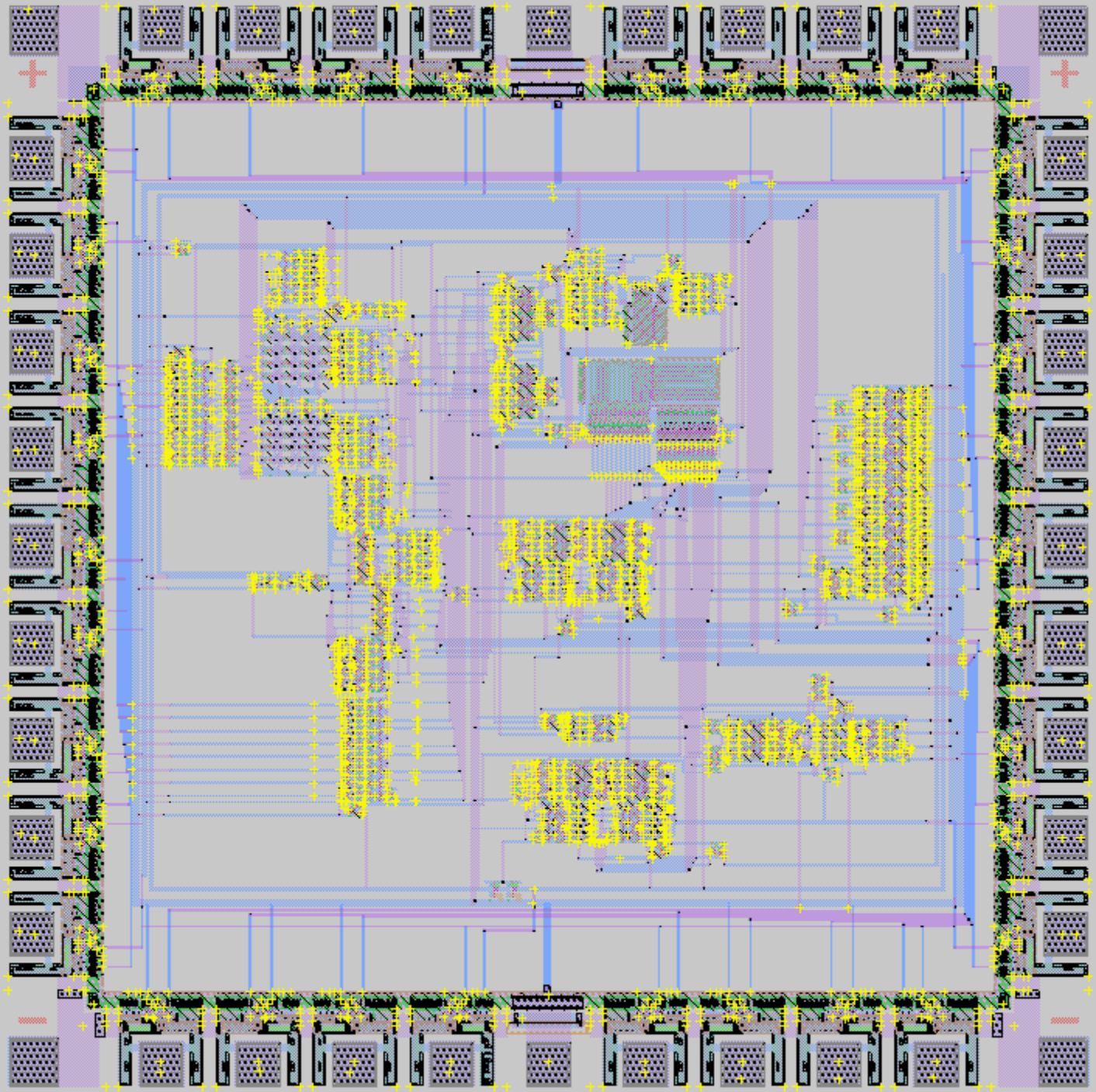


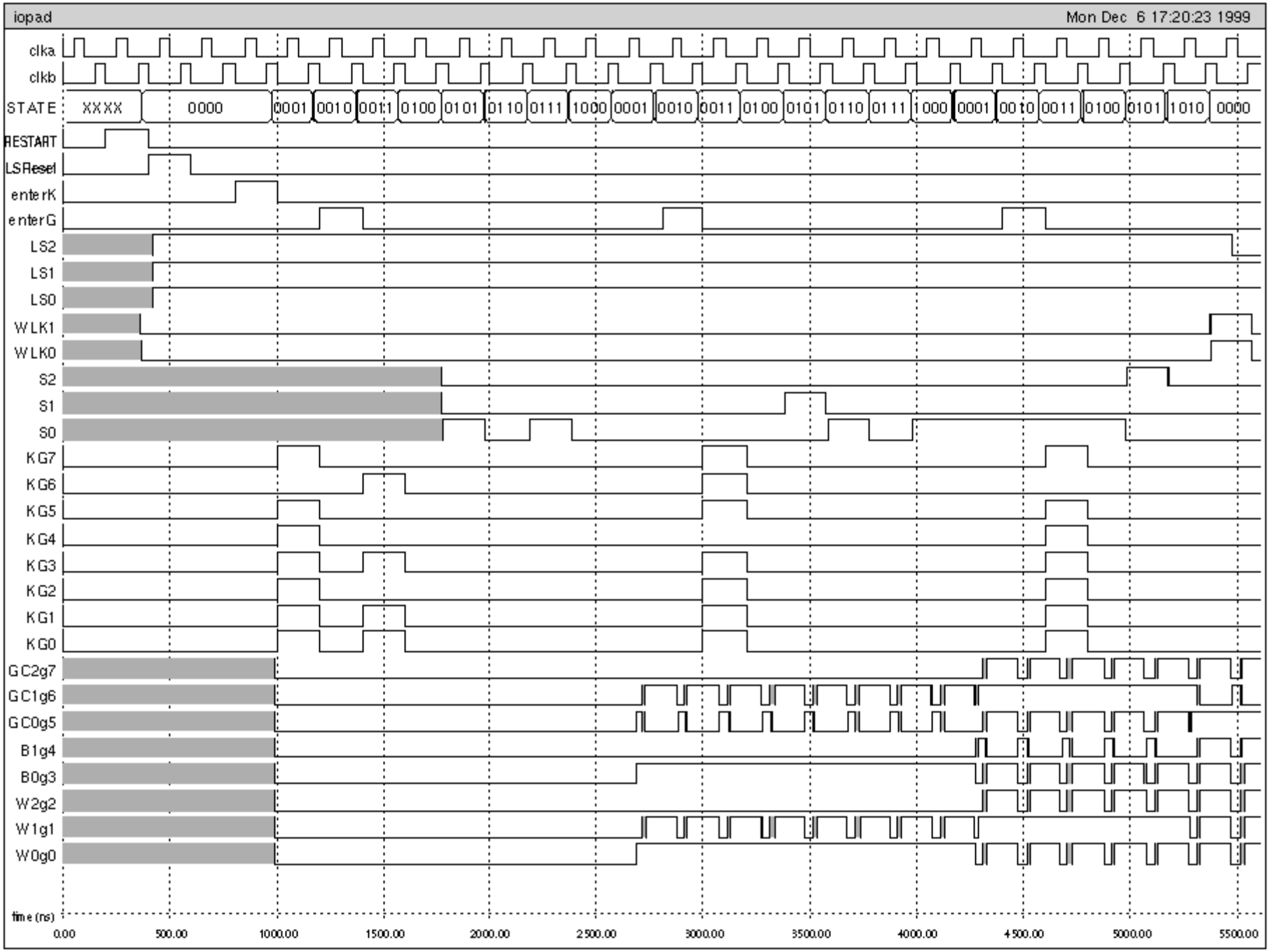
Decoder PLA



Floor Plan


Guess Counter
Output
8-Bit MUX
Black Score
2-Bit Latch
Low Score Logic
Input Logic
Key
8-Bit Latch
White Score
3-Bit Adder
FSM
Valid
4-Bit Latch
Guess
8-Bit Latch
Bit
0,2,4,6
4-Bit
Barrel Shifter
Comparison
Logic
4-Bit Latch
Decoder
PLA
Score
4-Bit Latch
Bit
1,3,5,7
4-Bit
Barrel Shifter
Valid
4-Bit Latch
Comparison Logic

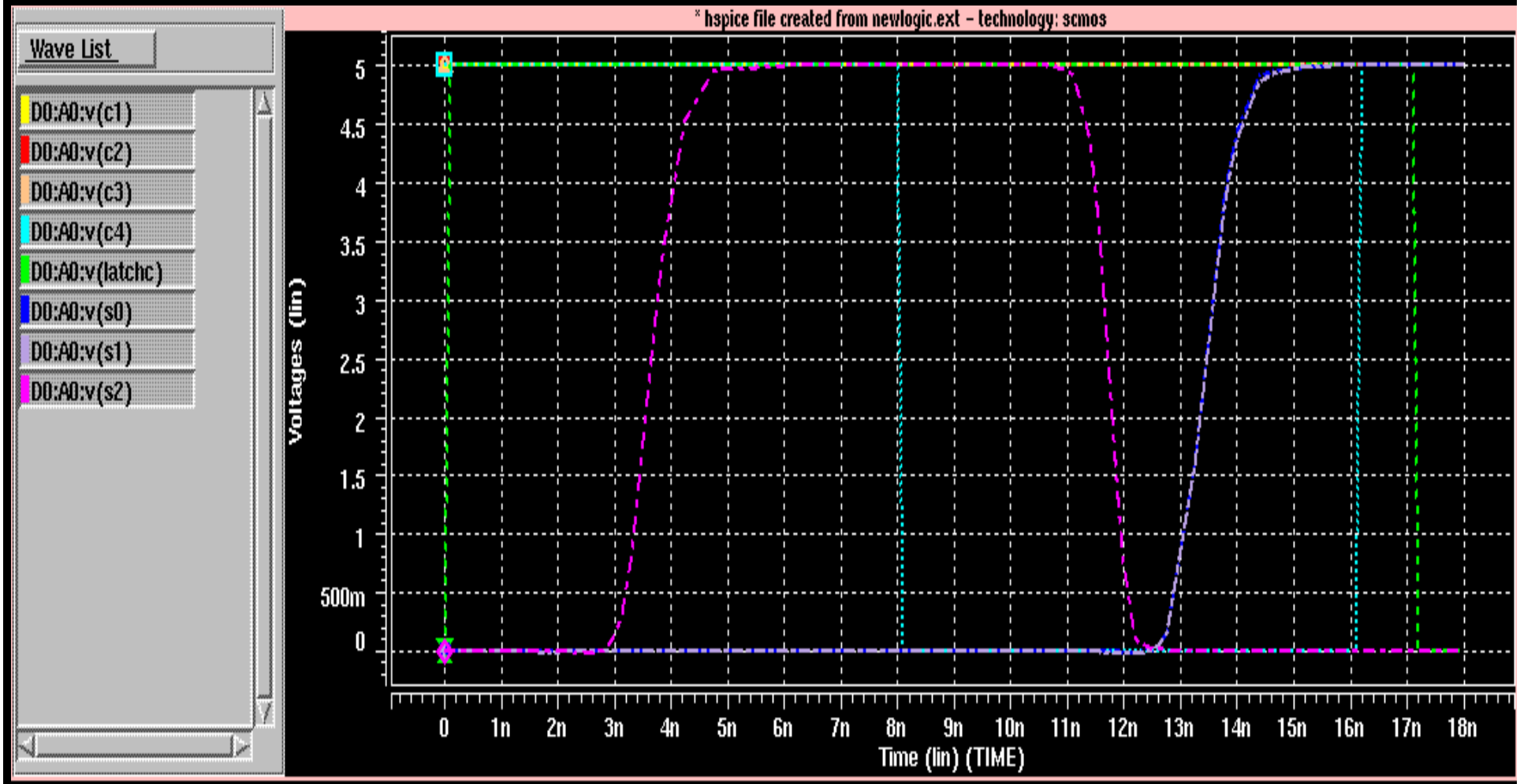



iopad
Mon Dec 6 17:20:23 1999
clka
clkb
STATE
XXXX
0000
0001
0010
0011
0100
0101
0110
0111
1000
0001
0010
0011
0100
0101
0110
0111
1000
0001
0010
0011
0100
0101
1010
0000
RESTART
LSReset
enterK
enterG
LS2
LS1
LS0
WLK1
WLK0
S2
S1
S0
KG7
KG6
KG5
KG4
KG3
KG2
KG1
KG0
GC2g7
GC1g6
GC0g5
B1g4
B0g3
W2g2
W1g1
W0g0
time (ns)
0.00
500.00
1000.00
1500.00
2000.00
2500.00
3000.00
3500.00
4000.00
4500.00
5000.00
5500.00

Spice Analysis of Critical Cells

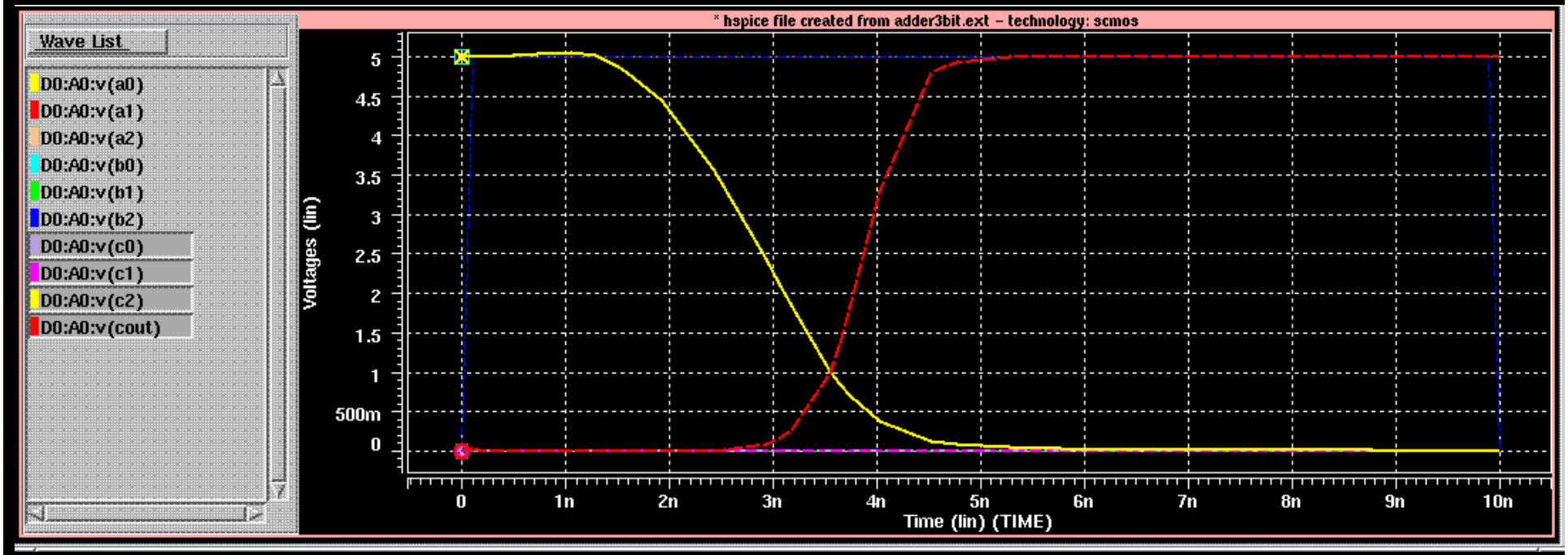
- **Comparison logic (longest path between clocks)**
- **Adder**

Spice Analysis of Comparison Logic



T = 2*7.47 = 14.94nS Frequency = 66.93MHz

Spice Analysis of Adder



T = 2*6.2 = 12.4nS Frequency = 80.65MHz > 67MHz

Q.E.D.