

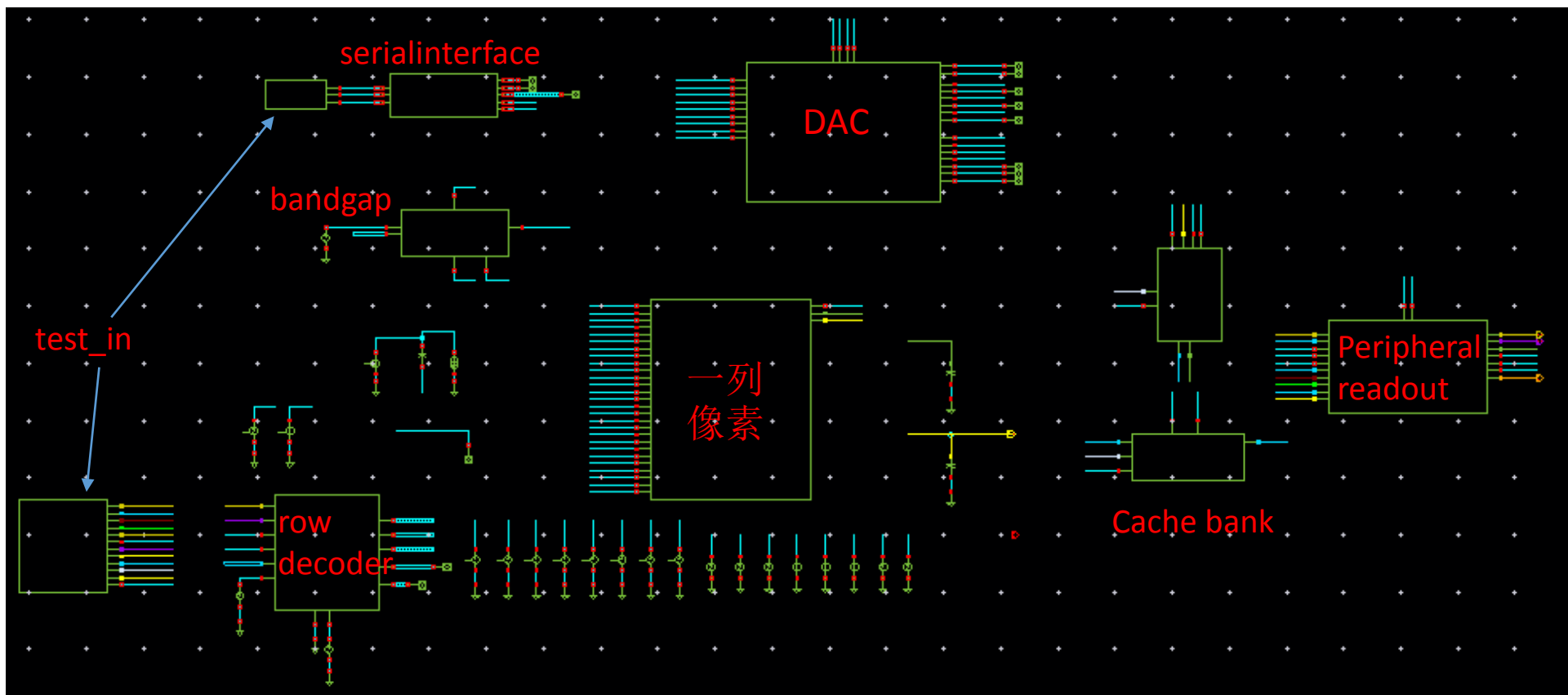
仿真验证

吴志岗

2019.7.26

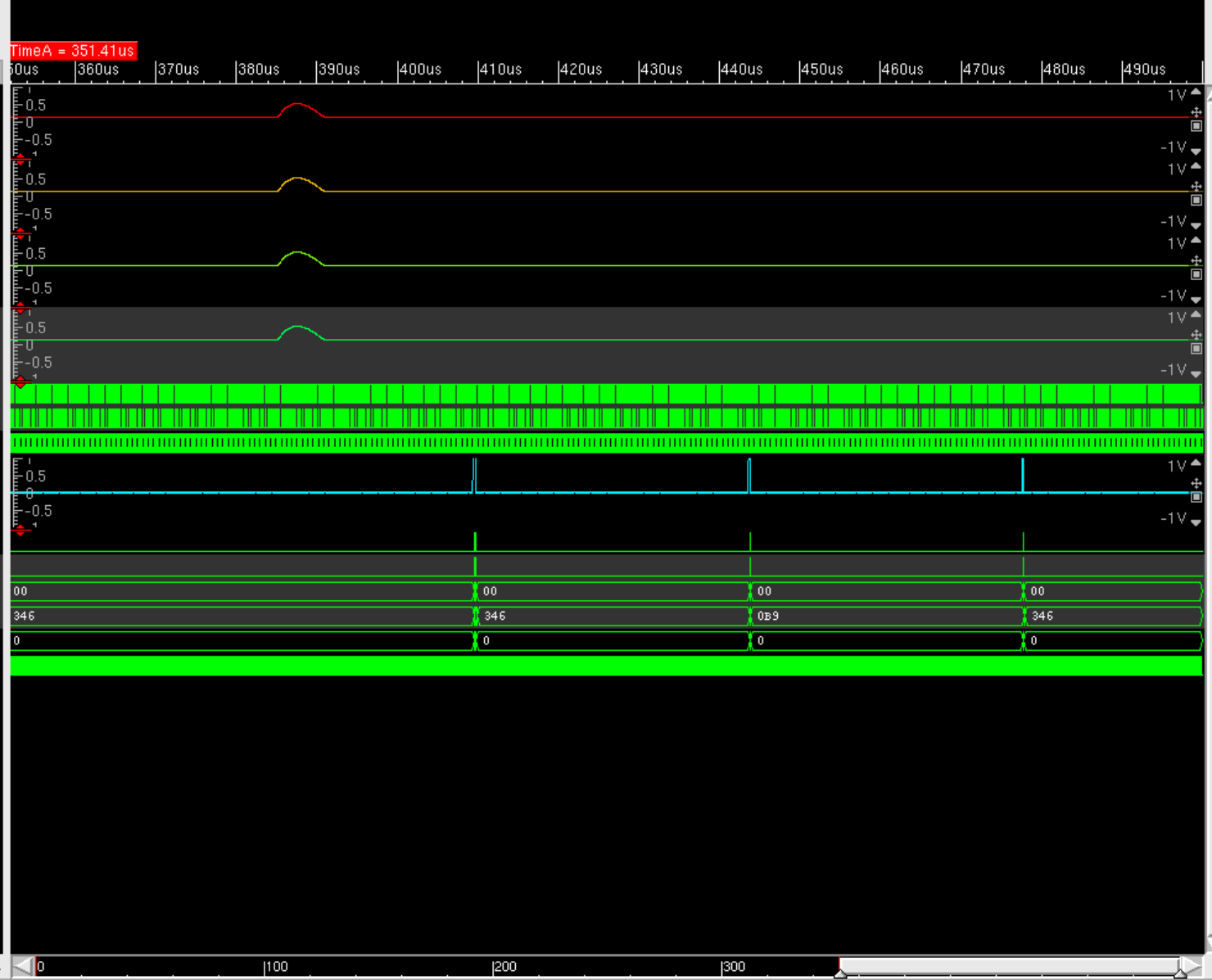
1. 联合仿真结果
2. 外围数字读出电路使用tips

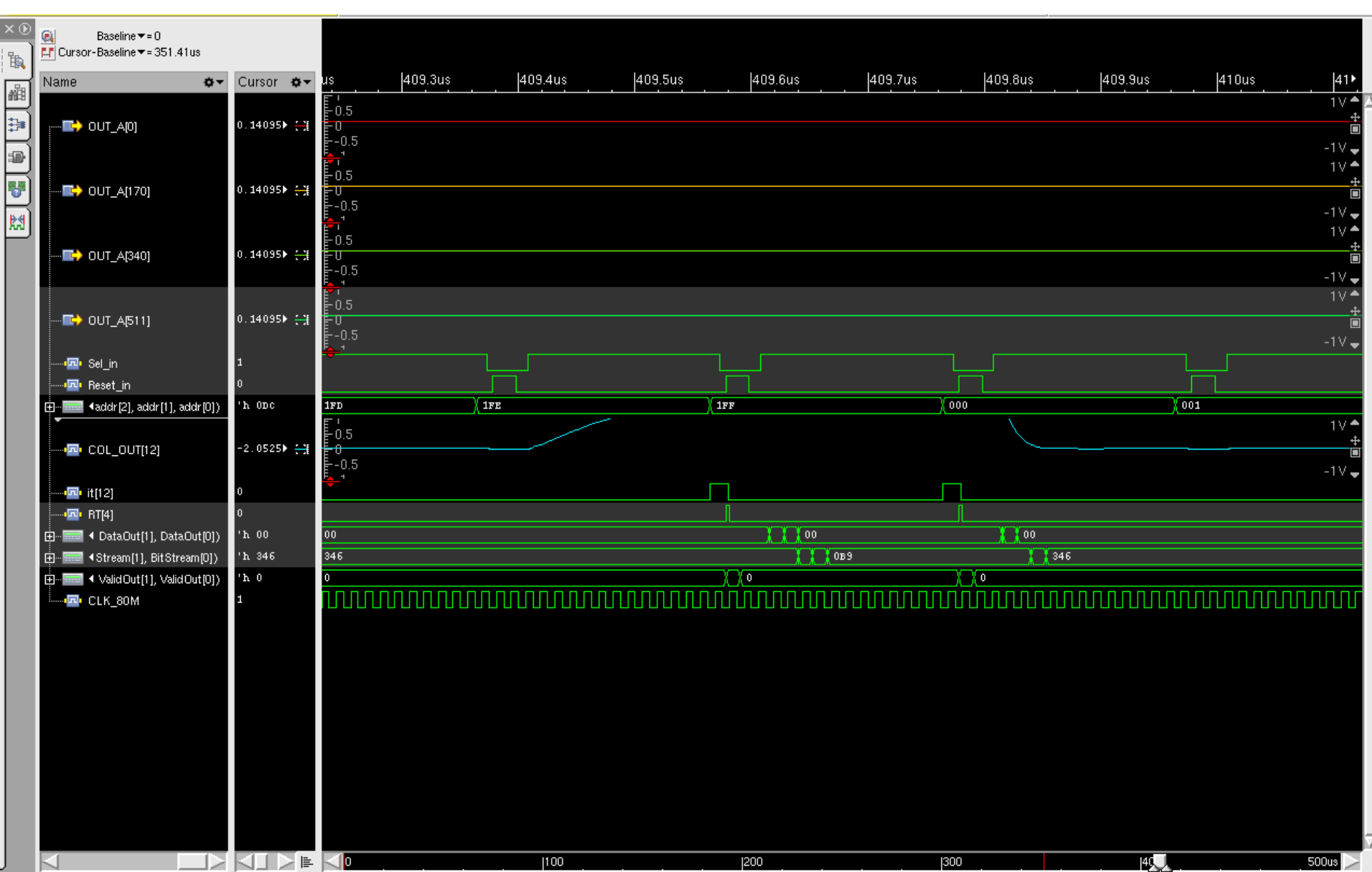
全芯片功能模块验证（不包括：IO pad, PLL, Serializer, LVDS）



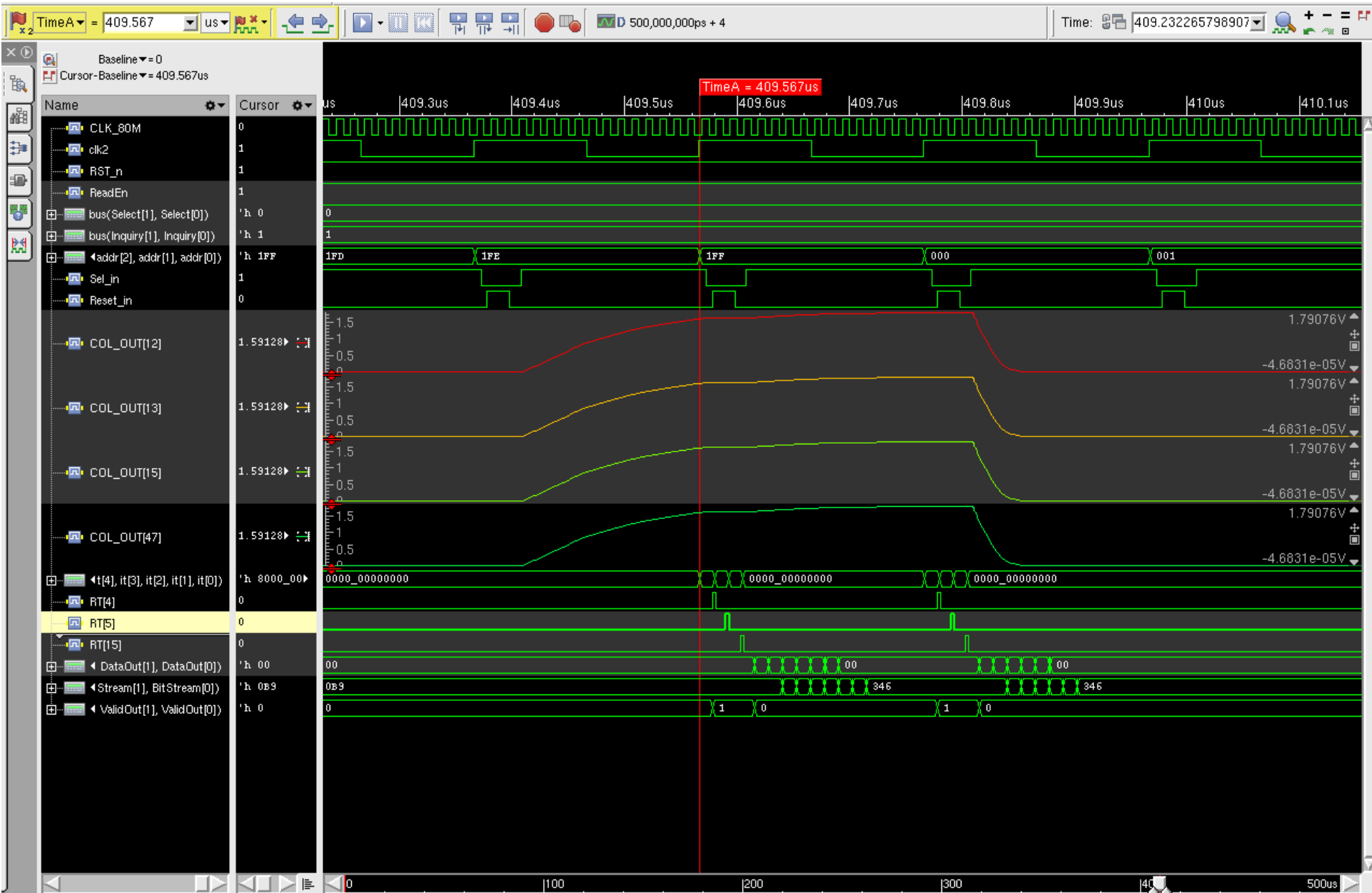
Baseline = 0
Cursor-Baseline = 351.41us

Name	Cursor
OUT_A[0]	0.14095
OUT_A[170]	0.14095
OUT_A[340]	0.14095
OUT_A[511]	0.14095
Sel_in	1
Reset_in	0
addr[2], addr[1], addr[0]	'h 0dc
COL_OUT[12]	-2.0525
it[12]	0
RT[4]	0
DataOut[1], DataOut[0]	'h 00
Stream[1], BitStream[0]	'h 346
ValidOut[1], ValidOut[0]	'h 0
CLK_80M	1





将一列像素扩展至4列，产生4*4个像素击中 (Row: 0,170,340,511, Col: 12,13,15,47)



外围读出电路tips

- 时序控制信号
 - 外围电路输入：CLK_80M, RST_n, ReadEn, Inquiry, Select, CACHE_CLK
 - 行选信号输入：Row_clk, Sel_in, Reset_in

CLK_80M: 下降沿有效。

Inquiry<1:0>选择模式：2'b00 输出K28.5, 2'b01 读FIFO（若FIFO为空，则读出8'b0），2'b10 读FIFO状态（共八位，高四位FIFO4~FIFO1 满标志，低四位FIFO4~FIFO1空标志），2'b11保留

Select<1:0>选择FIFO通路

CACHE_CLK: 上升沿有效，缓存HIT和Addr信息。CACHE_CLK的上升沿延迟

CLK_80M的下降沿要足够大（8ns），以防止每行的最后一个击中将下一行的最后一个击中RST

上述信号的有效沿应远离CLK_80M的下降沿以保证时序正确

编码和8b/10b

- 16位编码： 9位addr+4位col+3位hit
- 8b/10b： 查找表方法

5B/6B code

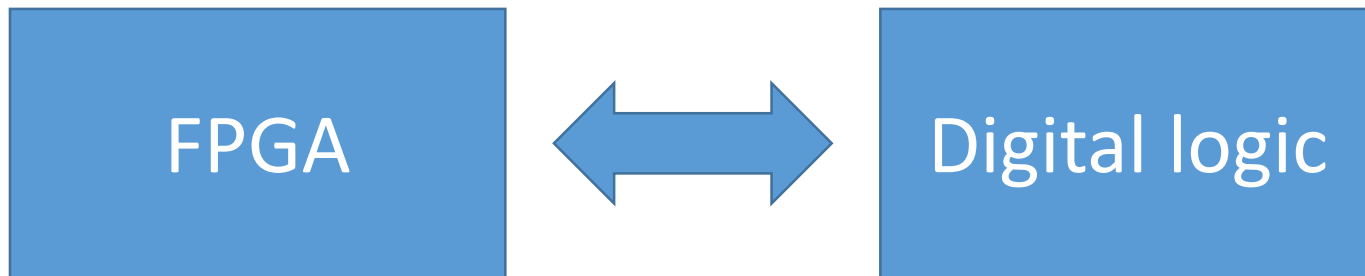
input	RD = -1		RD = +1		input	RD = -1		RD = +1	
	EDCBA	abcdei		EDCBA		abcdei			
D.00	00000	100111	011000		D.16	10000	011011	100100	
D.01	00001	011101	100010		D.17	10001	100011		
D.02	00010	101101	010010		D.18	10010	010011		
D.03	00011	110001		D.19	10011	110010			
D.04	00100	110101	001010		D.20	10100	001011		
D.05	00101	101001		D.21	10101	101010			
D.06	00110	011001		D.22	10110	011010			
D.07	00111	111000	000111		D.23 †	10111	111010	000101	
D.08	01000	111001	000110		D.24	11000	110011	001100	
D.09	01001	100101		D.25	11001	100110			
D.10	01010	010101		D.26	11010	010110			
D.11	01011	110100		D.27 †	11011	110110	001001		
D.12	01100	001101		D.28	11100	001110			
D.13	01101	101100		D.29 †	11101	101110	010001		
D.14	01110	011100		D.30 †	11110	011110	100001		
D.15	01111	010111	101000		D.31	11111	101011	010100	
					K.28	11100	001111	110000	

3b/4b code

input	RD = -1		RD = +1		input	RD = -1		RD = +1	
	HGF	fghj		HGF		fghj			
D.x.0	000	1011	0100		K.x.0	000	1011	0100	
D.x.1	001	1001			K.x.1 ‡	001	0110	1001	
D.x.2	010	0101			K.x.2 ‡	001	1010	0101	
D.x.3	011	1100	0011		K.x.3	011	1100	0011	
D.x.4	100	1101	0010		K.x.4	100	1101	0010	
D.x.5	101	1010			K.x.5 ‡	001	0101	1010	
D.x.6	110	0110			K.x.6 ‡	001	1001	0110	
D.x.P7 †	111	1110	0001						
D.x.A7 †	111	0111	1000		K.x.7 †‡	111	0111	1000	

使用方式

- 仅单个sector工作时
 - 可以一直工作在读FIFO的状态，监控ValidOut，在ValidOut有效的若干周期后BitStream产生有效读出
- 多个sector同时工作
 - 可以先读FIFO的空满状态信息，根据该信息去读非空的FIFO，读至连续输出两个8'b0时，FIFO读空



FPGA起着大脑的作用，需要依据不同的输出信号判断数据的类型，必要时加上标签，再输出至PC中