컴퓨터 구조

2013210061

채윤병

Lab Session 4

**fulladder.v**

module fa(c\_out,sum,c\_in,a,b);

output c\_out,sum;

input c\_in,a,b;

/\*to do\*/

wire c\_in,a,b;

wire c\_out,sum;

wire d,e,f;

xor(sum,c\_in,a,b);

and(d,a,b);

and(e,a,c\_in);

and(f,b,c\_in);

or(c\_out,d,e,f);

endmodule

**RCA.v**

`include "fulladder.v"

module RCA (c\_out, sum, c\_in, a, b);

input c\_in;

input [15:0] a, b;

output c\_out;

output [15:0] sum;

wire c[14:0];

/\* to do \*/

fa fa00 (c[0],sum[0], c\_in, a[0], b[0]);

fa fa01 (c[1],sum[1], c[0], a[1], b[1]);

fa fa02 (c[2],sum[2], c[1], a[2], b[2]);

fa fa03 (c[3],sum[3], c[2], a[3], b[3]);

fa fa04 (c[4],sum[4], c[3], a[4], b[4]);

fa fa05 (c[5],sum[5], c[4], a[5], b[5]);

fa fa06 (c[6],sum[6], c[5], a[6], b[6]);

fa fa07 (c[7],sum[7], c[6], a[7], b[7]);

fa fa08 (c[8],sum[8], c[7], a[8], b[8]);

fa fa09 (c[9],sum[9], c[8], a[9], b[9]);

fa fa10 (c[10],sum[10], c[9], a[10], b[10]);

fa fa11 (c[11],sum[11], c[10], a[11], b[11]);

fa fa12 (c[12],sum[12], c[11], a[12], b[12]);

fa fa13 (c[13],sum[13], c[12], a[13], b[13]);

fa fa14 (c[14],sum[14], c[13], a[14], b[14]);

fa fa15 (c\_out,sum[15], c[14], a[15], b[15]);

endmodule

**tb4RCA.v**

include "RCA.v"

module tb4RCA;

wire C\_OUT;

wire [15:0] SUM;

reg C\_IN;

reg [15:0] A, B;

RCA RCA00 (C\_OUT, SUM, C\_IN, A, B);

initial

begin

C\_IN = 1'd1;

A = 16'd21;

B = 16'd56;

#5

C\_IN = 1'd0;

A = 16'd23;

B = 16'd198;

#5

C\_IN = 1'd1;

A = 16'd423;

B = 16'd345;

#5

C\_IN = 1'd0;

A = 16'd999;

B = 16'd0;

#5

C\_IN = 1'd1;

A = 16'd5435;

B = 16'd3456;

#5

C\_IN = 1'd1;

A = 16'd2454;

B = 16'd9867;

#5

C\_IN = 1'd1;

A = 16'd1;

B = 16'd2;

#5

C\_IN = 1'd1;

A = 16'd3;

B = 16'd4;

#5

C\_IN = 1'd1;

A = 16'd5;

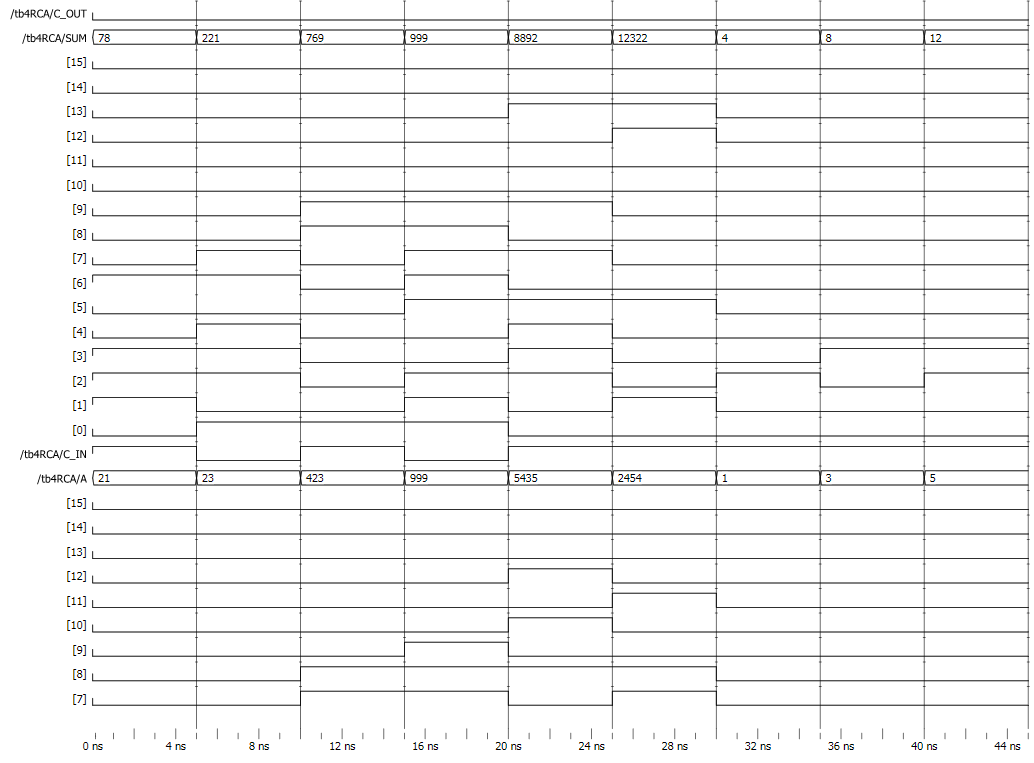
B = 16'd6;

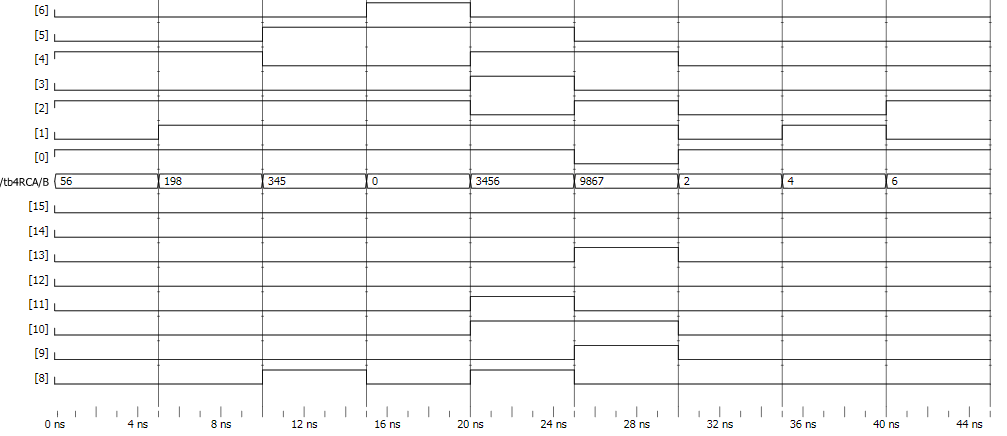
#5

$stop;

end

endmodule





리플 캐리 가산기는 전가산기 여러 개를 연결해서 이진수의 합을 연산하는 계산기다. 전가산기는 입력을 3개 받고 2개를 출력하는데 입력 값 2개는 더하려는 값 2개고 나머지 하나는 캐리이다. 출력 값 2개는 합의 결과와 캐리이다. 캐리 값은 입력 값 3개 중에 2개 이상이 1일 때 발생하고 그것을 위의 표에서 확인할 수 있다. 전가산기 16개를 연결하고 각 전가산기는 앞의 전가산기의 캐리를 입력을 받으며 서로 연결되어있다. 앞의 전가산기의 출력값을 입력값으로 받기 때문에 앞의 전가산기의 연산이 끝나기 전엔 연산을 시작할 수 없으므로 더하려는 비트의 자리수가 많아질수록 시간이 더 오래 걸리는 것도 특징이다. 16비트는 부호비트가 있다면 총 양수를 2^15까지 표현할 수 있기 때문엔 위의 표에선 마지막 전가산기에서 캐리가 출력되지 않았음을 확인할 수 있었다.

이 실험 중에 아쉬웠던 점은 워낙 연산이 빠르기 때문이었는지 앞의 전가산기 떄문에 연산이 지연되는 것을 그래프로 확인할 수 없었다는 점이다. 모든 출력값이 동시에 출력되는 것처럼 보이기 때문에 처음에 예상했던 결과와 달랐고 이 프로그램으로는 그것을 확인할 수 없을지 알아보았지만 방법을 찾지 못했다.