

# LAB 04 ( Ripple Carry Adder )

2013210111 남세현

## 1. 코드

[ 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 [14:0] carry;
```

```
    fa fa00 ( carry[0], sum[0], c_in, a[0], b[0]);
```

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

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

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

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

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

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

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

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

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

```
    fa fa10 ( carry[1], sum[10], carry[9], a[10], b[10]);
```

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

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

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

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

```
    fa fa15 ( c_out, sum[15], carry[14], a[15], b[15]);
```

```
endmodule
```

## [ fulladder.v ]

```
module fa(c_out,sum,c_in,a,b);
    output c_out,sum;
    input c_in,a,b;

    assign c_out = ( a & b ) | ( b & c_in ) | ( c_in & a );
    assign sum = a ^ b ^ c_in;
endmodule
```

## 2. 결과






	/tb4RCA/C_OUT	-No Data-									
	/tb4RCA/SUM	-No Data-	16'd78	16'd221	16'd769	16'd999	16'd8892	16'd12322	16'd4	16'd8	16'd12
	/tb4RCA/C_IN	-No Data-									
	/tb4RCA/A	-No Data-	16'd21	16'd23	16'd423	16'd999	16'd5435	16'd2454	16'd1	16'd3	16'd5
	/tb4RCA/B	-No Data-	16'd56	16'd198	16'd345	16'd0	16'd3456	16'd9867	16'd2	16'd4	16'd6

Figure 1 - RCA TB

	16'h0006
	16'hffff
	16'h0006

Figure 2 - Test for Carrying

## 3. 결과 분석

16Bit RCA는 16비트의 두 숫자 A, B와 C\_IN이 주어졌을 때 그 값을 더해서 SUM과 C\_COUT을 처리하는 기능을 한다. 값들이 제대로 더해져서 나옴을 WaveForm에서 확인할 수 있다.

RCA에 대한 설명은 이 보고서의 핵심이 아니므로 여기에 작성하지 않겠다.

## 4. 구현시 어려웠던 점 - 없음