```
//////////////////////////////////////////////////////////////////////////////////
// Company: TEC
// Engineer: Giovanni Villalobos Quirós,Eduardo Ortiz,Adriana Chavarría, César Ortega
//
// Create Date:    19:16:01 09/17/2016
// Design Name:         alineadorDeDirecciones
// Module Name:    alineadorDeDirecciones
// Project Name:         I investigacion
// Target Devices: Nexys 2
// Description:  Alineador de direcciones para la implementacion de una arquitectura uniciclo
MIPS
//
//
//Target Devices:
// ------------------------------------------------
// | Device |
// ------------------------------------------------
// | Family | Spartan3e |
// | Part | xc3s500e |
// | Package | fg320 |
// | Temp Grade | Commercial |
// | Process | Typical |
// | Speed Grade | -4 |
// | Characterization | PRODUCTION,v1.2,06-23-09 |
// ------------------------------------------------
//
//
// | IOB Name | Descripcion | Type | Direction | IO Standard | Drive | Slew | IOB |
// | | | | | | Strength | Rate | Delay |
// +-----------------------------------------------------------------------------------------------+
// | dir[0-31] | Direccion a operar  | IBUF | INPUT | LVCMOS25 | | | 0 / 0 |
// | dirOut[0-31] | Resultado de la alineacion de direcciones | IOB | OUTPUT | LVCMOS25 |
12 | SLOW | 0 / 0 |
// +-----------------------------------------------------------------------------------------------+
// Delay: 19.321ns
/// Area dentro del dispositivo:
//
// Number of occupied Slices Used: 21, Available: 4,656 Utilization: 1%
//
// Number of bonded IOBs Used:64, Available: 232, Utilization:27%
//
// Number of 4 input LUTs Used:41 Available:9,312 , Utilization: 1%
// Energy: 81mW
//////////////////////////////////////////////////////////////////////////////////
module AlineadorDirecciones_V1_ChavarriaOrtizOrtegaVillalobos(
        input [31:0] dir,
```

```
            output [31:0] dirOut
    );

            //Se instancia el sumador para sumarle 4 a la dirección de entrada
            thirtyTwoBitsFullAdder ttbfa (
            .a(dir),
            //Se suma 4 a la dirección entrante
            .b(4),
            .s(dirOut),
            .c0(0),
            .cout()
);

endmodule

module thirtyTwoBitsFullAdder(
    input [31:0] a,
    input [31:0] b,
    output [31:0] s,
            input c0,
            output cout
    );

            wire c1,c2,c3,c4,c5,c6,c7;

            fourbitFullAdder fbfa1 (.a(a[3:0]), .b(b[3:0]),.c0(0),.c4(c1),.s(s[3:0])
            );

            fourbitFullAdder fbfa2 (.a(a[7:4]),.b(b[7:4]),.c0(c1),.c4(c2),.s(s[7:4])
            );

            fourbitFullAdder fbfa3 (.a(a[11:8]),.b(b[11:8]),.c0(c2),.c4(c3),.s(s[11:8])
            );

            fourbitFullAdder fbfa4 (.a(a[15:12]),.b(b[15:12]),.c0(c3),.c4(c4),.s(s[15:12])
            );

            fourbitFullAdder fbfa5 (.a(a[19:16]),.b(b[19:16]),.c0(c4),.c4(c5),.s(s[19:16])
            );

            fourbitFullAdder fbfa6 (.a(a[23:20]), .b(b[23:20]),.c0(c5),.c4(c6), .s(s[23:20])
            );

            fourbitFullAdder fbfa7 (.a(a[27:24]), .b(b[27:24]),.c0(c6), .c4(c7),.s(s[27:24])
            );
```

```
        fourbitFullAdder fbfa8 (.a(a[31:28]), .b(b[31:28]), .c0(c7), .c4(cout),.s(s[31:28])
        );

endmodule

module fourbitFullAdder(
    input [3:0] a,
    input [3:0] b,
    input c0,
        output wire c4,
    output [3:0] s
        );
        wire [3:0] c;
        wire [3:0] p;
        wire [3:0] g;
        //Instanciacion de 4 half adders para hacer un solo sumador completo
        // de 4 bits

        halfAdder ha1 (
                .a(a[0]),
                .b(b[0]),
                .s(p[0]),
                .c(g[0])
        );

        halfAdder ha2 (
                .a(a[1]),
                .b(b[1]),
                .s(p[1]),
                .c(g[1])
        );

        halfAdder ha3 (
                .a(a[2]),
                .b(b[2]),
                .s(p[2]),
                .c(g[2])
        );

        halfAdder ha4 (
                .a(a[3]),
                .b(b[3]),
                .s(p[3]),
                .c(g[3])
        );
```

```
	//Se instancia la unidad del carry de 4 bits anticipado
	carryUnit4Bits uut (
		.p(p),
		.g(g),
		.c0(c0),
		.c(c)
	);
	//Cada uno de estos xor representa la salida de los 4 digitos del sumador.
	xor(s[0],c0,p[0]);
	xor(s[1],c[0],p[1]);
	xor(s[2],c[1],p[2]);
	xor(s[3],c[2],p[3]);
	or(c4,c[3],0);
endmodule


module halfAdder(
	input wire a,
	input wire b,
	output wire s,
	output wire c
   );
	//Ecuacion de la salida sumada
	xor(s,a,b);
	//Ecuacion para el carry de salida
	and(c,a,b);
endmodule
```