

Prueba 1: Procesado digital de la señal con FPGAs

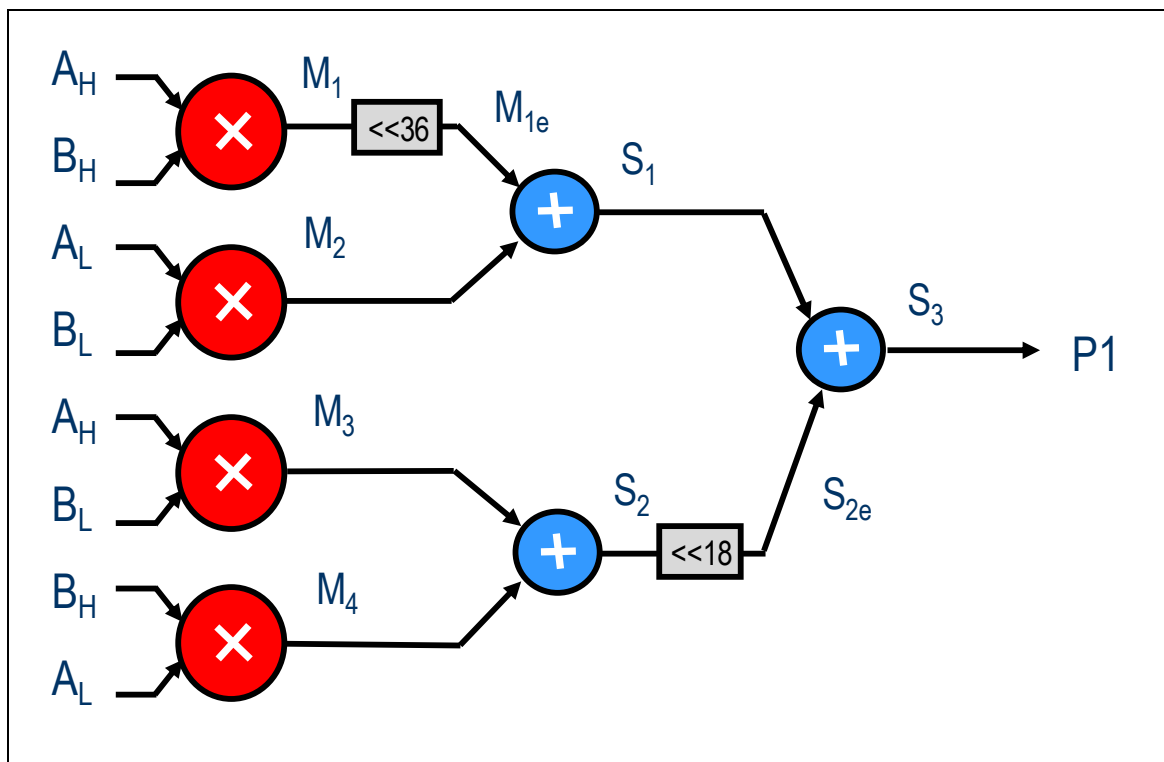
Alumno:

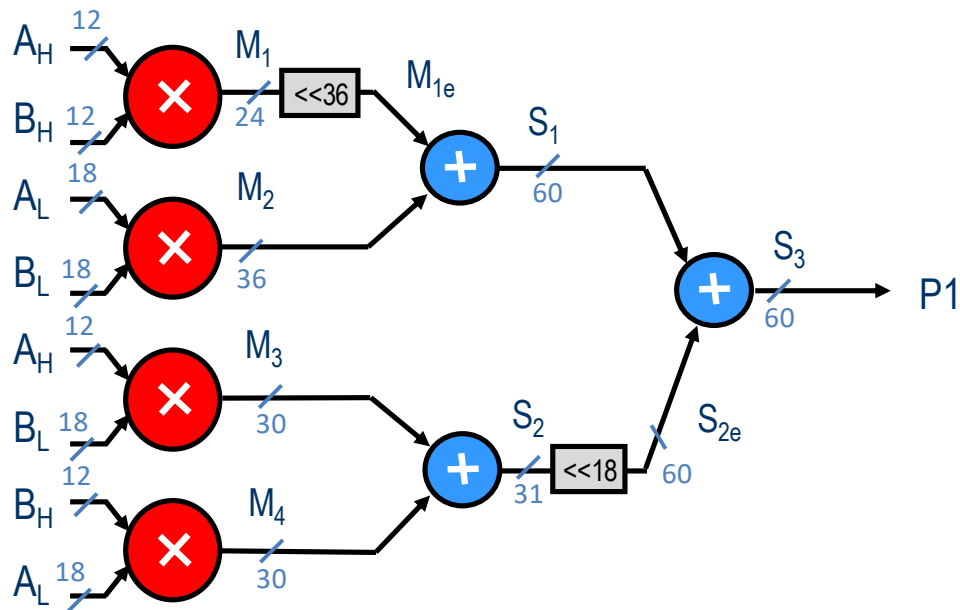
29/3/2019

Se desea realizar la implementación de un multiplicador con operandos de 30 bits para operar a alta velocidad. El formato numérico de los operandos A y B es [30,29] con signo en complemento a dos. Para facilitar el funcionamiento del multiplicador a alta velocidad en un dispositivo Cyclone IV, que dispone de *hard-cores* multiplicadores de 18x18 bits con y sin signo, se va a implementar el multiplicador descomponiendo los operandos en dos palabras concatenadas ($A=A_H \cdot 2^{18} + A_L$ y $B=B_H \cdot 2^{18} + B_L$) atendiendo a la siguiente ecuación:

$$P1 = A \cdot B = A_H \cdot B_H \cdot 2^{36} + A_L \cdot B_L + (A_H \cdot B_L + B_H \cdot A_L) \cdot 2^{18}$$

1) Indique en el siguiente esquema los tamaños de palabra en cada punto del circuito.





2) Suponiendo que el tiempo de propagación de un sumador es $t_{sum} = 1.5ns$, el del multiplicador es $t_{mult} = 3.5ns$, y que se consideran nulos los tiempos de t_{su} y t_{co} de los registros, indique cuáles son los puntos óptimos para segmentar el circuito y calcule la frecuencia máxima de operación del operador segmentado. Asuma que las entradas llegan registradas.

El circuito se debe segmentar a las salidas de los multiplicadores y después del sumador final (S3) debido a que $t_{mult} > 2t_{sum}$. La frecuencia máxima de operación $f_{max} = 1/(t_{co}+t_{mult}+t_{su}) \approx 1/(t_{mult}) = 285.7 \text{ MHz}$.

3) Modele con Verilog el multiplicador segmentado. Utilice los nombres de las señales que aparecen en el dibujo.

```
module mult(
input clk,
input signed [29:0] A,B,
output signed [59:0] P1);

wire signed [11:0] AH,BH;
wire [17:0] AL,BL;

reg signed [23:0] M1; // Mult de 12x12 bits
reg [35:0] M2; // Mult de 18x18 bits
reg signed [30:0] M3,M4; // Mults de 12x18 bits

wire signed [59:0] S1;
wire signed [31:0] S2;
reg signed [59:0] S3;

// Señales extendidas
//wire signed [59:0] M1e;
```

```

wire signed [59:0] S2e;

// Formar sub-palabras
assign AL = A[17:0];
assign AH = A[29:18];
assign BL = B[17:0];
assign BH = B[29:18];

// Mults registrados

always@(posedge clk)
begin
    M1 <= AH * BH;
    M2 <= AL * BL;
    M3 <= AH * $signed({1'b0,BL});
    M4 <= BH * $signed({1'b0,AL});
end

// Sumas parciales
//assign M1e = M1<<<36;
//S1 = M1e + $signed({1'b0,M2});
assign S1 = {M1,M2};
assign S2 = M3 + M4;

// Suma final registrada
assign S2e = S2<<<18;

always@(posedge clk)
    S3 <= S1 + S2e;

//Asignación final
assign P1 = S3;

endmodule

```

4) Escriba cómo se debe modificar el código de Verilog de la cuestión anterior para que la salida del multiplicador se compute con un redondeo “nearest” al formato [30,28] con signo en complemento a dos.

```

module mult(
input clk,
input signed [29:0] A,B,
output signed [29:0] P2);

// El código anterior es igual hasta el cálculo de S3

always@(posedge clk)
    P2 = S3[59:30]+S3[29];

endmodule

```

5) Escriba la ecuación de Matlab que modela la operación del multiplicador con la salida redondeada “nearest” al formato [30,28] a partir de los operandos A y B.

$$P2 = \text{nearest}(A \cdot B \cdot 2^{28}) \cdot 2^{-28}$$

6) Indique si la inclusión del redondeo cambia la frecuencia máxima de funcionamiento del multiplicador segmentado. Justifique su respuesta.

Sí que la cambia porque ahora el camino critico pasa a la parte de los sumadores al añadir un sumador adicional por el redondeo: $t_{mult} < 3t_{sum}$.

7) Indique los recursos hardware necesarios para implementar el multiplicador con redondeo. Tenga en cuenta las posibles simplificaciones del circuito. Justifique su respuesta.

Elemento	Número
Mult 18x18	4
LEs	103 (31 de S2, 42 de S3 y 30 del sumador de redondeo)
FFs	150 (24 de M1, 36 de M2, 30 de M3, 30 de M4 y 30 de la salida)

Hay que tener en cuenta que:

- S1 no necesita implementar un sumador, sino que concatenando los dos operandos ya se obtiene la operación deseada (M1e está desplazado a la izquierda 36 bits y M2 tiene un tamaño de 36 bits)*
- S2e está desplazado 18 bits, por tanto esos 18 bits menos significativos no requieren ser sumados*