UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Trabalho 1 - Sistemas Digitais
Matheus Claudino Bica
Cartão 164383

Caixa 1 - Operador AND de 8 bits de entrada, sendo saída "1111"&resultado_and
**Caixa 2 - Multiplicador sem sinal usando operador *

Sumário
Frequência Máxima: 189.451MHz

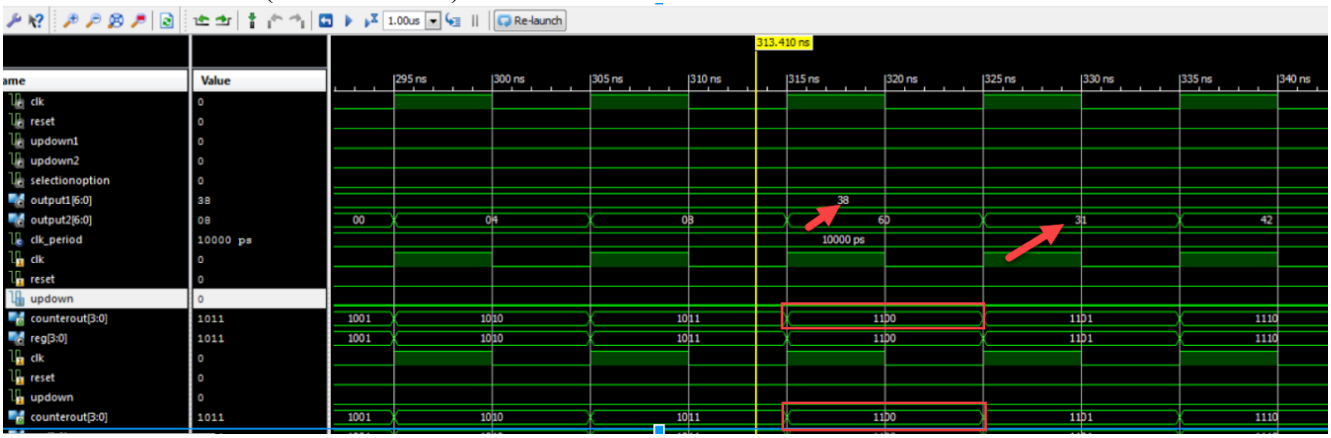
main Project Status (09/13/2018 - 20:05:58)			
Project File:	trabalho1_Bica_2018_2.xise	**Parser Errors:**	No Errors
Module Name:	main	**Implementation State:**	Placed and Routed
Target Device:	xc3s50-5pq208	**• Errors:**	No Errors
Product Version:	ISE 14.7	**• Warnings:**	No Warnings
Design Goal:	Balanced	**• Routing Results:**	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	**• Timing Constraints:**	All Constraints Met
Environment:	System Settings	**• Final Timing Score:**	0 (Timing Report)

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	16	1,536	1%	
Number of 4 input LUTs	30	1,536	1%	
Number of occupied Slices	18	768	2%	
Number of Slices containing only related logic	18	18	100%	
Number of Slices containing unrelated logic	0	18	0%	
Total Number of 4 input LUTs	30	1,536	1%	
Number of bonded IOBs	19	124	15%	
Number of MULT18X18s	1	4	25%	
Number of BUFGMUXs	1	8	12%	
Average Fanout of Non-Clock Nets	3.32			

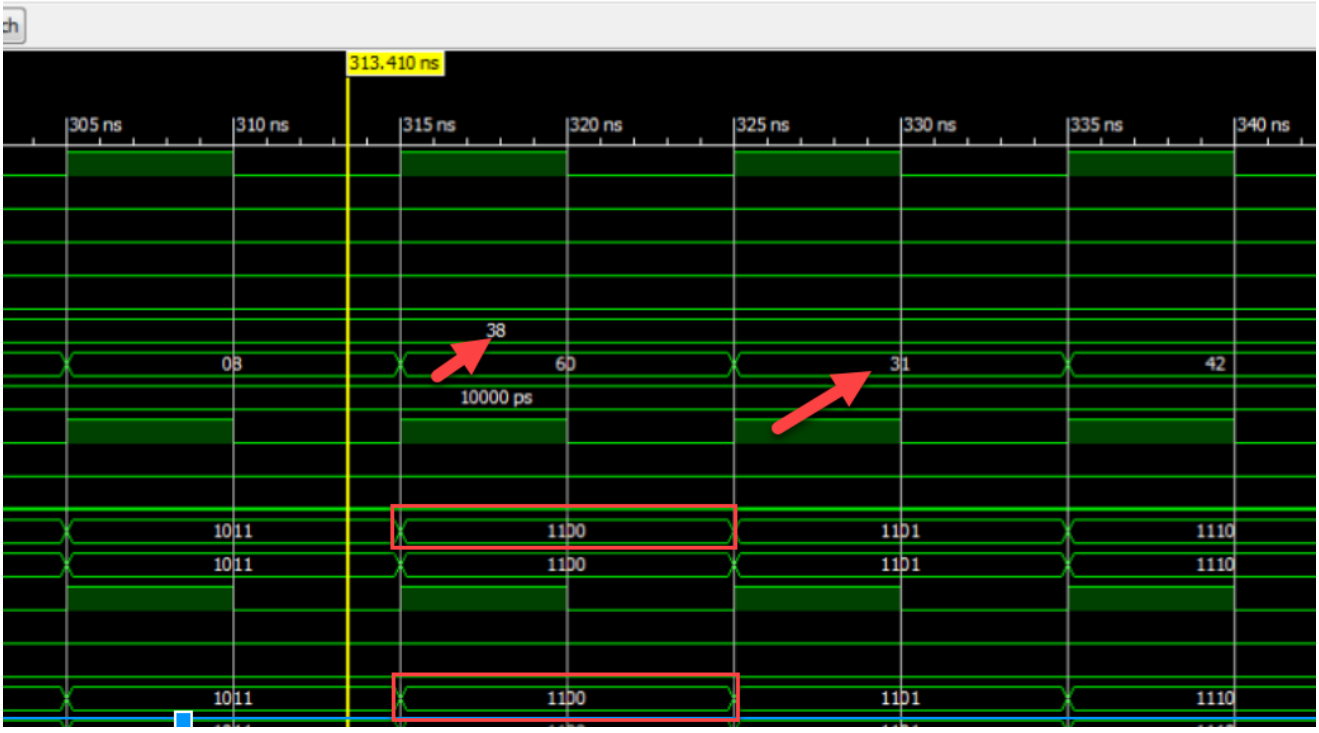
Parte do código do TestBench relevante para análise:

```
    -- Stimulus process
    stim_proc: process
    begin
        wait for 100 ns;
        wait for clk_period*10;
        -- insert stimulus here
        reset <= '0';

        updown1 <= '0';
        updown2 <= '0';
        selectionOption <= '0';
      wait for clk_period*16;

        updown1 <= '0';
        updown2 <= '0';
        selectionOption <= '1';
      wait for clk_period*16;

        updown1 <= '1';
        updown2 <= '0';
        selectionOption <= '0';
      wait for clk_period*16;

        updown1 <= '1';
        updown2 <= '0';
        selectionOption <= '1';
      wait for clk_period*16;

      wait;
    end process;

END;
```

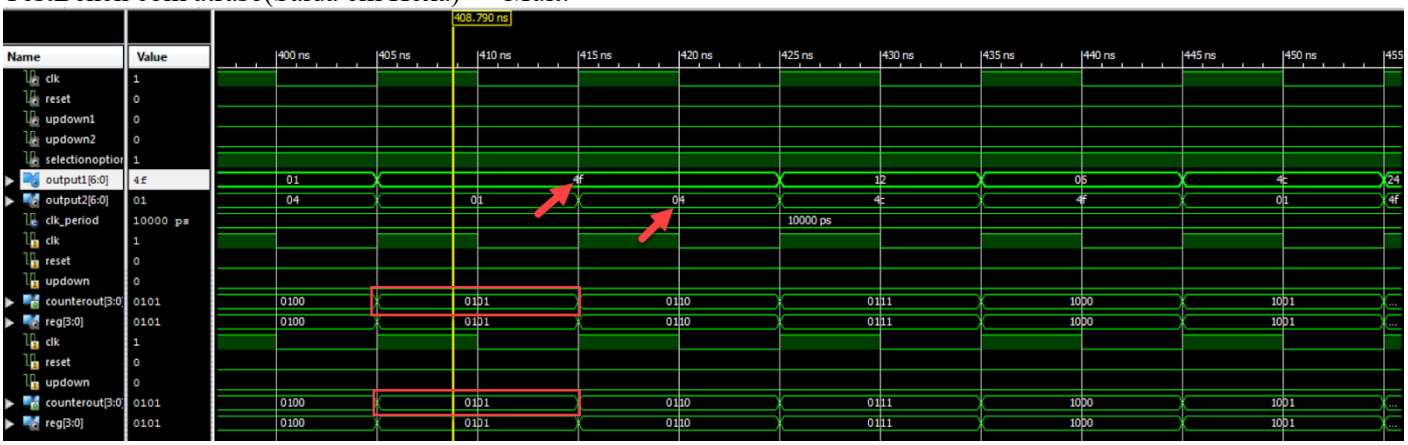
TestBench com atraso(Saída em Hexa) → AND:



Zoom(Saída em Hexa) → AND:



TestBench com atraso(Saída em Hexa) → Mult:



Zoom(Saída em Hexa) → Mult:

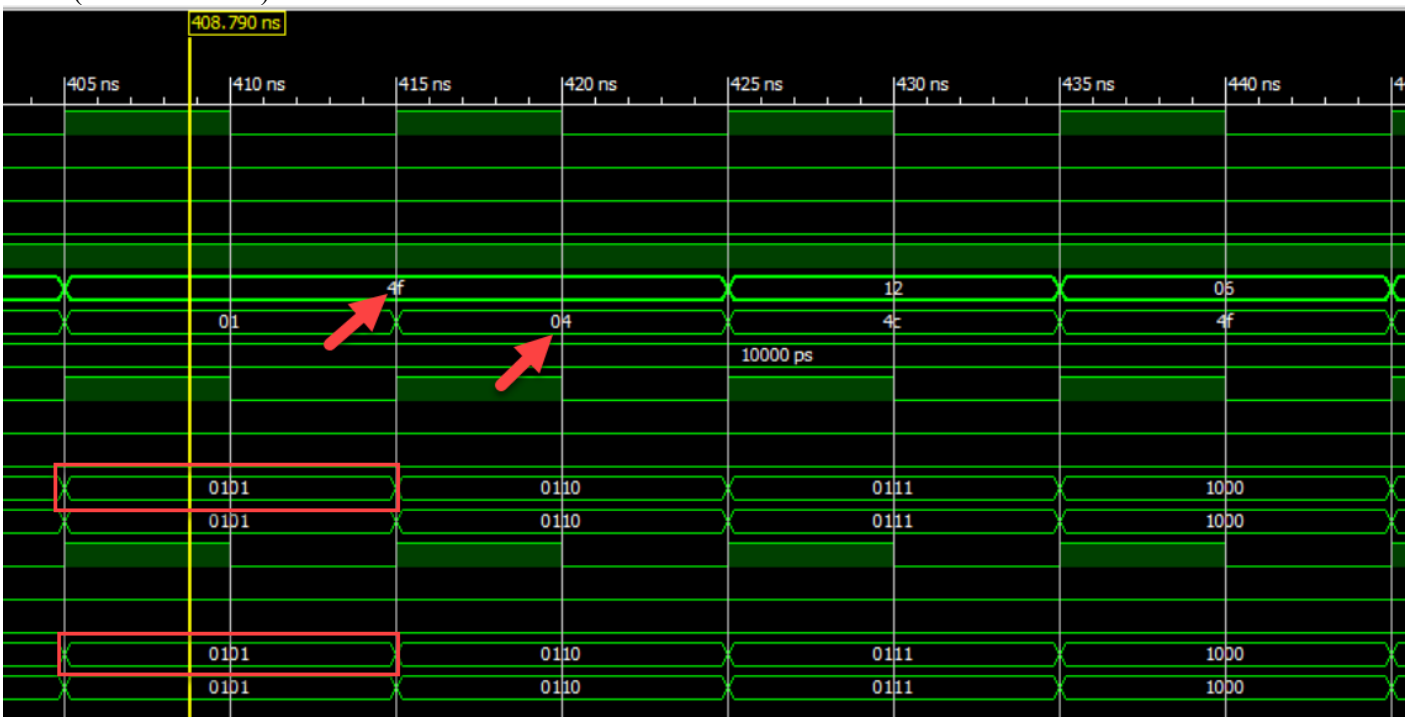
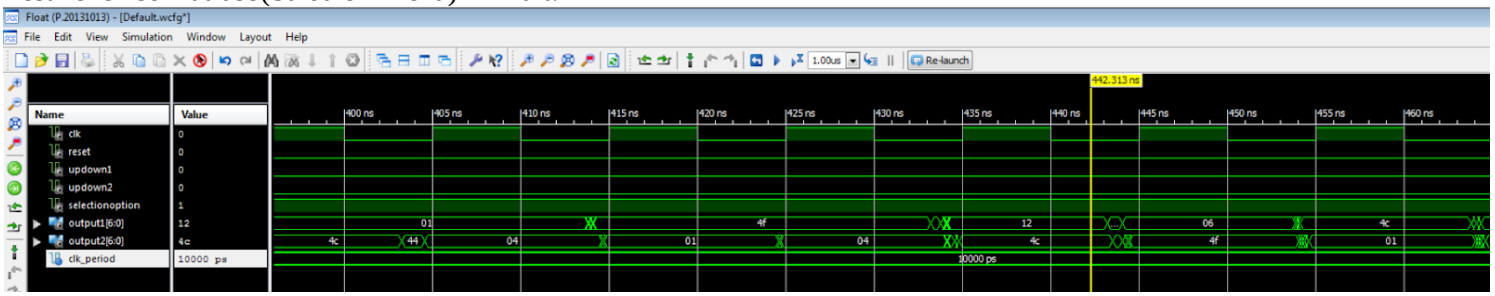


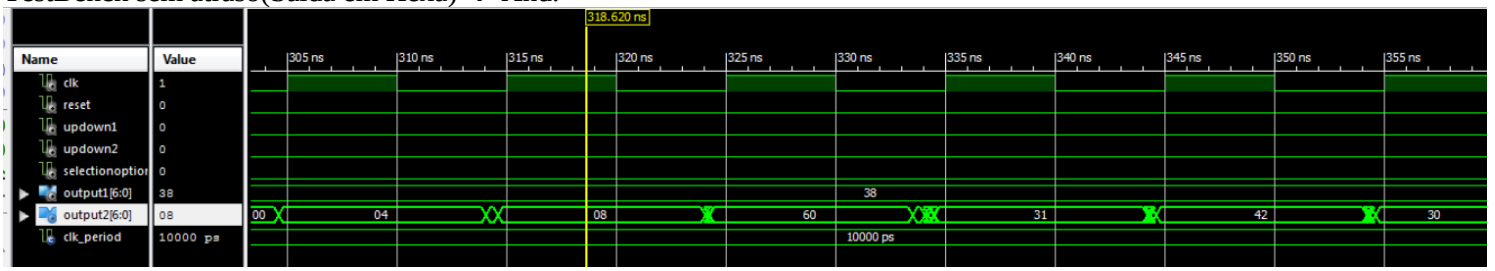
Tabela para comparação de saída do nibble7seg (em hexa):
Número de Saída → Representação em Hexa

0 → 1	1 → 4F	2 → 12	3 → 6	4 → 4C	5 → 24	6→ 20	7→ F
8→ 0	9→ 4	A→ 8	B→ 60	C→ 31	D→ 42	E→ 30	F→ 38

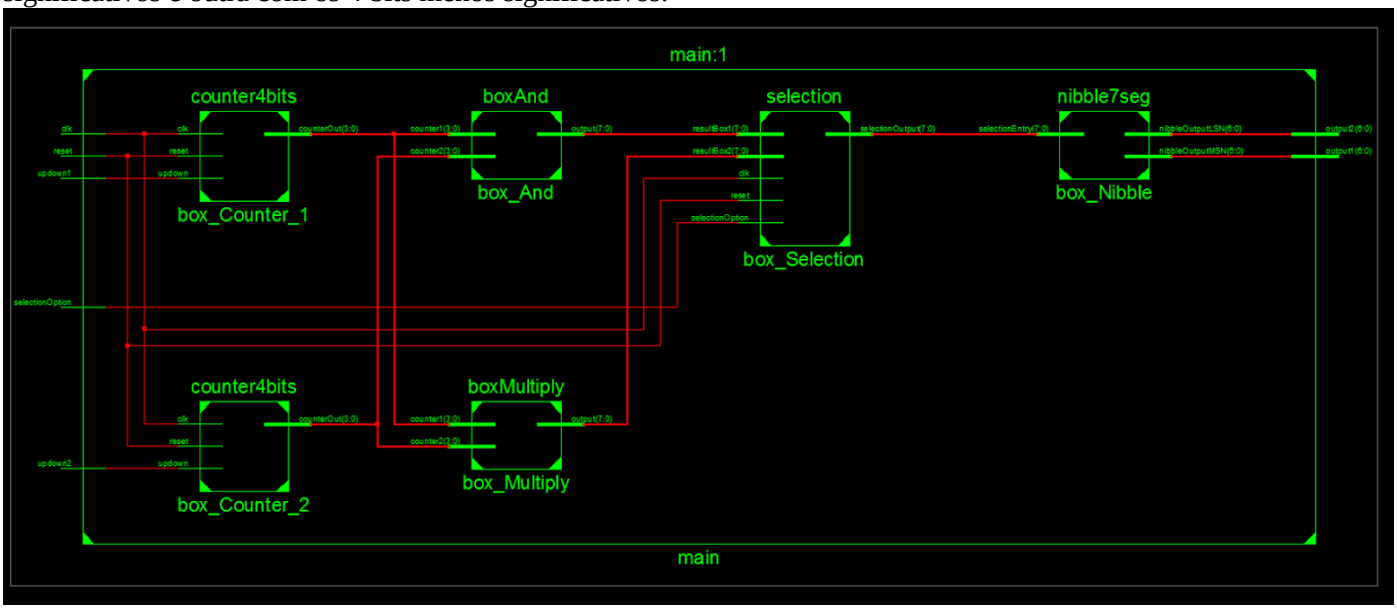
TestBench sem atraso(Saída em Hexa) →Mult:



TestBench sem atraso(Saída em Hexa) → And:



RTL Schematic:
Utilizei somente um **nibble7seg** que recebe os 8bits e divide em cada saída um 7 segmentos com os 4bits mais significativos e outra com os 4 bits menos significativos.



Technology Schematic: