Projecto de Sistemas Digitais
Digital Systems Design
MEEC/MEAer, 2018/19, 2nd Exam

Table 1: Timing characteristics and resource utilization of logic-arithmetic components				
	Area	t_P	t_{SETUP}	t_{HOLD}
16-bit Register	10	2 ns	1 ns	1 ns
BRAM	100	2 ns	1 ns	1 ns
16-bit Adder / Subtractor	16	5 ns		
16-bit Multiplier (simple)	120	14 ns		
16-bit Divider	500	85 ns		
MUX 2:1 (16-bit data words)	16	3 ns		
MUX 4:1 (16-bit data words)	32	4 ns		
MUX 8:1 (16-bit data words)	64	5 ns		
16-bit logic operators	16	3 ns		

32-bit Adder / Subtractor	32	6 ns
64-bit Adder / Subtractor	64	8 ns
32-bit Multiplier (simple)	500	28 ns
32-bit Multiplier (3-stage pipeline)	540	11 ns
64-bit Multiplier (simple)	2000	56 ns
64-bit Multiplier (6-stage pipeline)	2200	12 ns

Note: all answers must be properly justified, without which they will not be classified.

1. Consider the algorithm, with 8 inputs and 2 outputs, defined by the following pseudo-code:

$$y_1 = (x_1 \times x_2) \times (((x_3 + x_4) \times (x_5 - x_6)) + (x_7 + x_8))$$
$$y_2 = (x_5 - x_6) \times x_3$$

a) **[1 val]** Draw the corresponding data flow graph.

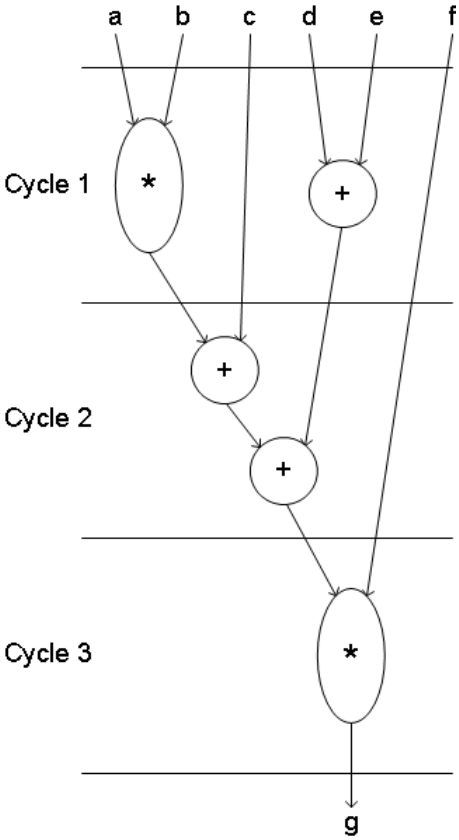
b) **[3 val]** Define a priority list using the critical path as metric.

Consider that the inputs are stored in independent registers and that the arithmetic operators available are 1 multiplier (simple) and 2 adder/subtractors. Consider the timing characteristics shown on table 1, for the registers, arithmetic operators and multiplexers.

Obtain a list scheduling using the priority list defined and considering a target clock period of **12ns**.

2. Consider the schedule defined in the figure, at the right.

   The circuit to be implemented must use only the minimum number of adders and the minimum number of multipliers required to implement this schedule.
   All input values are 16-bit integers in 2's complement representation. All circuit operators, buses and datapath registers are 16-bit wide, with the timing/area characteristics shown on Table 1. All circuit registers must be positive edge-triggered, fully synchronous and be synchronized with a unique global clock.



a) **[3.5 val]** Design the circuit datapath in order to achieve the minimum total area (given the constraints indicated).
   Indicate the number of registers and multiplexers required.
   Draw the block diagram of your datapath.
   Estimate the area occupied, the minimum clock period, and the latency for your design.
   Justify all the bindings and sharing of resources.

3. Consider that you want to design a circuit, ***in a Xilinx Artix-7 device***, to calculate the variance of a set of N elements, **as fast as possible**:

$$\sigma = \frac{1}{N} \sum_{i=1}^{N} (x_i - \mu)^2$$

Consider that N=2048=2^{11} , that the x_i are 18-bit integers stored in one RAMB36E1 block configured in ***simple dual-port mode (SDP)***, and that the mean value **μ** is a real number represented in fixed-point format Q18.11 and stored in one independent register.
All operators, buses and registers must be sized in order to avoid the existence of *overflows* and to avoid any loss of accuracy in the computations.
All registers must be positive edge-triggered and be synchronized with a unique global clock. Select the most adequate logic-arithmetic components listed on table 1, and use the timing characteristics indicated in the table to estimate the performance of your circuit (consider the timing characteristics of the most approximately sized operator).

a) **[1 val]** Briefly explain the advantage of configuring a RAMB36E1 block in ***simple dual-port*** mode (instead of configuring it in ***true dual-port*** mode).

b) **[1 val]** Define the fixed-point representation format required for the variance result **σ**. Justify.

c) **[3.5 val]** Define the port settings of the RAMB36E1 block where all x_i elements are stored, such that the total execution time of the circuit can be minimized. (This BRAM block is configured in ***simple dual-port mode***).

Design the circuit datapath in order to achieve the target performance. Draw the block diagram of the data unit.

Specify the dimensions of all operands, buses, and registers.

Indicate the clock period and the latency of the circuit you propose.

Justify.

4. **[2 val]** Specify the VHDL architecture of an arithmetic component to perform the operation $M = C \times (X + Y)$, where the operands are real numbers represented in fixed-point format, as indicated:

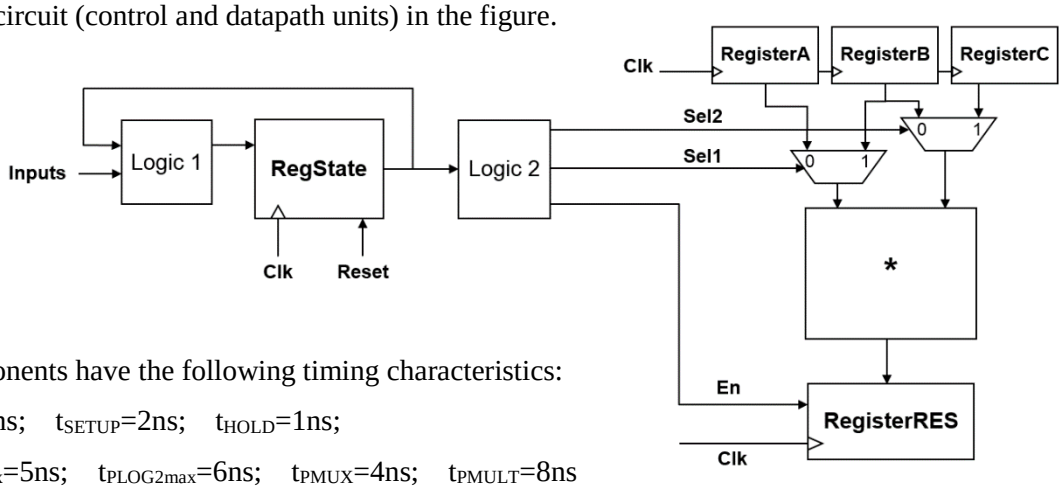
Signal	*Fixed-point format*
X	Q12.4
Y	Q2.14
C	Q5.13

Complete the specification below of its input / output ports, where ***mi*** is the integer part of ***m*** and ***mf*** is the fractional part of ***m***. Dimension all operators and buses in order to preserve maximum accuracy in the calculations and avoid overflows.

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity addm is
  Port ( x  : in  std_logic_vector ( ??? );
         y  : in  std_logic_vector ( ??? );
         c  : in  std_logic_vector ( ??? );
         mi : out std_logic_vector ( ??? );
         mf : out std_logic_vector ( ??? ));
end addm;

architecture behavioral of addm is

???

begin

???

end behavioral;
```

5. **[1 val]** Consider now that the arithmetic operations of the component specified in 4. will be implemented using DSP48E1 blocks.
   How many DSP48E1 blocks will be required? Justify.

6. Consider the circuit (control and datapath units) in the figure.



The circuit components have the following timing characteristics:

t_{PREG}=3ns; t_{SETUP}=2ns; t_{HOLD}=1ns;

$t_{PLOG1max}$=5ns; $t_{PLOG2max}$=6ns; t_{PMUX}=4ns; t_{PMULT}=8ns

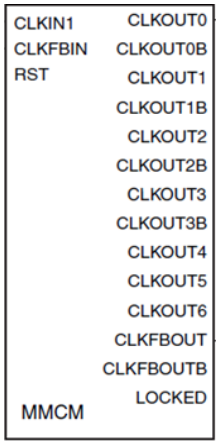
All circuit registers are synchronized by the same clock with a period T=25ns (f=40MHz).

a) **[1 val]** Estimate the WNS (Worst Negative Slack) of the circuit. Justify and indicate the critical logic path of the circuit.

b) **[1,5 val]** Is it possible to increase the clock frequency of the circuit by control pipelining, without modifying the datapath unit?

   If yes, explain how, what modifications are necessary and what will be the new minimum clock period. If not, explain the limitations.

c) **[1,5 val]** Explain how to configure a Mixed-Mode Clock Manager (MMCM), ***in a Xilinx Artix-7 device,*** to generate the circuit 40MHz clock, from a 100 MHz external oscillator (T_{ext} = 10ns), and how to connect it to the circuit. Justify.
   When defining the MMCM feedback, consider that there is no required phase relationship between the external and internal clocks.
   Explicitly specify the values that should be defined for the attributes:
   (M) Mult Counter: CLKFBOUT_MULT_F
   (D) Divide Counter: DIVCLK_DIVIDE,
   (O0) CLKOUT0 Divider: CLKOUT0_DIVIDE_F,
   Indicate the VCO frequency defined.



Symbol	Description	Speed Grade 1.0V -1	Units
MMCM_F_{INMAX}	Maximum input clock frequency	800.00	MHz
MMCM_F_{INMIN}	Minimum input clock frequency	10.00	MHz
MMCM_F_{VCOMIN}	Minimum MMCM VCO frequency	600.00	MHz
MMCM_F_{VCOMAX}	Maximum MMCM VCO frequency	1200.00	MHz
MMCM_F_{OUTMAX}	MMCM maximum output frequency	800.00	MHz
MMCM_F_{OUTMIN}	MMCM minimum output frequency	4.69	MHz
MMCM_$T_{LOCKMAX}$	MMCM maximum lock time	100.00	µs