Exercise 1 (40 min, 4 pt).

For the following digital circuit described in VHDL:

1. Complete the sensitivity lists for processes P1 and P2.
2. Complete the declarations for the signals in the architecture.
3. Draw the hardware resulting of P2 logic synthesis. How many flip-flops are required?
4. Complete the TestBench below. It should generate a Clk with 20ns Period and activate the input signal ***NewSampleIn*** during a Clk cylcle with 120 ns between activations.

```
--------------------------------------------------------------------------
-- DESIGN
--------------------------------------------------------------------------
library IEEE;
   use IEEE.std_logic_1164.all;
   use IEEE.numeric_std.all;
--------------------------------------------------------------------------
entity ADDR_GEN_MEM is
  port (
    Clk          : in  std_logic;
    Reset        : in  std_logic;
    NewSampleIn  : in  std_logic;
    MemAddrBus   : out unsigned(7 downto 0);
    SamplesCount : out unsigned(7 downto 0);
    EndOfMemory  : out std_logic);
end ADDR_GEN_MEM;
--------------------------------------------------------------------------
architecture Behavioural of ADDR_GEN_MEM is
  signal sMemAddressRow    :
  signal sMemAddressColumn :
  signal sSamplesCount     :
begin
  SamplesCount <= to_unsigned(sSamplesCount/2,8);
  MemAddrBus   <= sMemAddressRow & sMemAddressColumn;
  EndOfMemory  <= '1' when (sMemAddressColumn AND sMemAddressRow)= "1111" else
'0';

  P1: process (                  )
  begin
    if Reset = '1' then
      sSamplesCount <= (others => '0');
    elsif Clk'event and Clk = '1' then
     if NewSampleIn = '1' then
       if sSamplesCount = 255 then
            sSamplescount <= 0;
         else
           sSamplesCount <=  sSamplesCount + 1;
         end if;
     end if;
    end if;
  end process P1;
```

```
  P2: process(                    )
  begin
    if Reset = '1' then
      sMemAddressRow      <= (others => '0');
      sMemAddressColumn   <= (others => '0');
    elsif Clk'event and Clk = '1' then
      if NewSampleIn = '1' then
        if sMemAddressColumn = "1111" then
       sMemAddressColumn <= (others => '0');
       if sMemAddressRow = "1111" then
         sMemAddressRow <= (others => '0');
       else
            sMemAddressRow <= sMemAddressRow + 1;
       end if;
        else
          sMemAddressColumn <= sMemAddressColumn + 1;
        end if;
     end if;
    end if;
  end process P2;
end Behavioural;
--------------------------------------------------------------------------
-- TEST BENCH
--------------------------------------------------------------------------
library IEEE;
   use IEEE.std_logic_1164.all;
   use IEEE.numeric_std.all;
--
entity Ej1_tb is
end Ej1_tb;
--
architecture Behavioural of ej1_tb is
  component  ADDR_GEN_MEM is
  port (
    Clk          : in  std_logic;
    Reset        : in  std_logic;
    NewSampleIn  : in  std_logic;
    MemAddrBus   : out unsigned(7 downto 0);
    SamplesCount : out unsigned(7 downto 0);
    EndOfMemory  : out std_logic
   );
  end component;
  signal Clk           : std_logic;
  signal Reset         : std_logic;
  signal NewSampleIn   : std_logic;
  signal MemAddrBus    : unsigned(7 downto 0);
  signal SamplesCount  : unsigned(7 downto 0);
  signal EndOfMemory   : std_logic;
begin
  UUT: ADDR_GEN_MEM
  port map(
    Clk           => Clk,
    Reset         => Reset,
    NewSampleIn   => NewSampleIn,
    MemAddrBus    => MemAddrBus,
    SamplesCount  => SamplesCount,
    EndOfMemory   => EndOfMemory);




end Behavioural;
```

Problem 2 (50 min, 6 pto)

We want to design a digital circuit that processes data according to the following equation:

$$A = \frac{1}{2W}\sum_{i=0}^{w-1}(l_0 * cnt_{i+1} + l_1 cnt_i)^2,$$

Where W corresponds to the number of samples to be processed in each iteration, in this case 128 samples, the input data (cnt) is represented in natural binary with 8 bits and l0 and l1 correspond to adjustment coefficients with values 0.65 and 0.39.

The system has a 100 kHz clock (Clk) to perform the operations, an asynchronous initialization signal (rst), and an 8-bit data input (Data_in) where a new input data *cnt* enters with each clock cycle. The circuit output Data_out represents the final value of A after processing W samples. The R_ready output indicates the presence of a new valid output data. The circuit entity is as follows:

Translated with DeepL.com (free version)

```
entity problem2 is
  generic(W:integer:=128);
  port (
    Clk        : in  std_logic;
    rst      : in  std_logic;
    Data_in    : in  unsigned(7 downto 0);
    R_ready    : out std_logic;
    Data_out   : out unsigned(X downto 0));
end problem2;
```

Answer the following questions in a reasoned way:

1. Express so that it can be implemented using integer numbers and using divisions by powers of 2. Required coefficients are to be implemented with 8-bit unsigned numbers. Justify your answer.
2. Calculate the number of bits required at the output to accurately represent the result. Justify your answer.
3. A serial architecture must be implemented. Draw it at the Register Transfer Level (RTL), indicating the number of bits for each signal. Name the intermediate signals in this diagram and use the same names in the VHDL description in the following section as required.
4. Describe in VHDL the proposed architecture, including:
    a. Describe the declaration of signals needed to store partial and final results.
    b. Describe the sequential process that registers the input data.
    c. Describe the intermediate operations using concurrent assignments or combinational processes.
    d. Describe the accumulator needed to store the result during W samples.
    e. Describe the counter that allows us to keep track of the W samples.
5. The following synthesis results have been obtained for three different circuit architectures: parallel, serial, and pipeline. Identify each of them by comparing each parameter presented in the table and justify your answer.

Arquitecture	A	B	C
LUTs	5615	5630	4000
FFs	330	200	310
Tclk (ns)	9.8	20	11.2