IMPERIAL COLLEGE LONDON

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING
EXAMINATIONS 2014

EEE/EIE PART II: MEng, BEng and ACGI

Corrected Copy

DIGITAL ELECTRONICS 2

Monday, 2 June 2:00 pm

Time allowed: 2:00 hours

There are THREE questions on this paper.

Answer ALL questions.
Q1 carries 40% of the marks. Questions 2 and 3 carry equal marks (30% each).

Any special instructions for invigilators and information for candidates are on page 1.

Examiners responsible First Marker(s) : P.Y.K. Cheung
Second Marker(s) : W. Dai

Information for Candidates:

The following notation is used in this paper:

1. Unless explicitly indicated otherwise, digital circuits are drawn with their inputs on the left and their outputs on the right.
2. Within a circuit, signals with the same name are connected together even if no connection is shown explicitly.
3. The notation X2:0 denotes the three-bit number X2, X1 and X0. The least significant bit of a binary number is always designated bit 0.
4. Signed binary numbers use 2's complement notation.

1. (a) *Figure 1.1* shows the schematic diagram of a binary counter with a clock signal CLK, a synchronous reset input R, and a synchronous select input S. The select input S is low if the counter is a 10-bit counter, and high if it is a 12-bit counter. The counter has one output signal TO, which goes high for one clock cycle when the counter reaches a value of 1023 or 4095 if S is 0 or 1 respectively. Specify this counter circuit in Verilog HDL so that it can be synthesized. Estimate and justify the number of Logic Elements (LEs) required to implement this design on a Cyclone III FPGA.

[8]

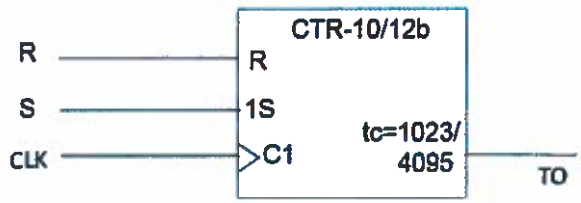


Figure 1.1

(b) Explain the principle of operation of a 12-bit pulse-width modulation (PWM) digital-to-analogue converter (DAC). Design in Verilog HDL a PWM DAC with an interface as defined in *Figure 1.2*.

[8]

```
module pwm_dac (clk, data_in, pwm_out);

   input          clk;      // system clock
   input [11:0]   data_in;  // input data for conversion
   output         pwm_out;  // PWM output
```

Figure 1.2

(c) *Figure 1.3* shows the timing diagram of a shift register based control circuit with a clock signal CLK and a trigger input signal TRIGGER, which changes on the falling edge of CLK and is a pulse lasting for one period of CLK. Design in schematic or Boolean equations the control circuit that generates output signals X and Y as shown in the timing diagram using shift registers and logic gates.

[8]

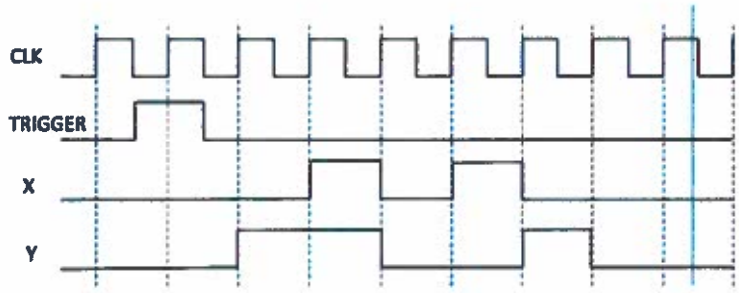


Figure 1.3

(d) *Figure 1.4* shows a circuit with two D-flipflops FF1 and FF2 with setup and hold times of 3 ns and 2 ns respectively, and a clock-to-Q output delay of 2 ns. The clock signal CLK has a 1:1 mark-space ratio. The datapath is driven by logic_A, which has a propagation delay between 2 ns and 5 ns. The clock path is driven by logic_B, which has a propagation delay between 1 ns and 10 ns. Derive the maximum frequency of the signal CLK for reliable operation of this circuit.

[8]

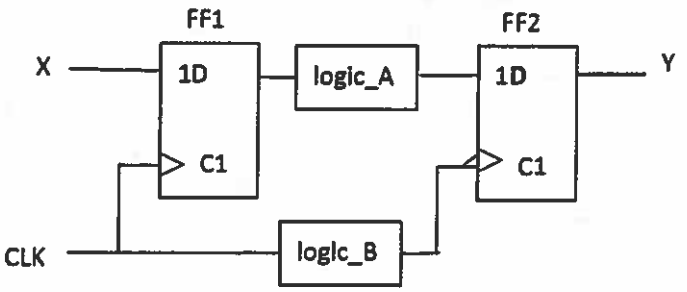


Figure 1.4

(e) A microprocessor system with an 18-bit memory address bus has a memory map as shown in *Figure 1.5*. The system consists of two banks of RAMs, RAM_1 and RAM_2, one bank of ROM, and a 32-location space for input and output. The two banks of RAMs are respectively 32k and 16k in size, and occupy consecutive address spaces starting from address 0. The ROM is 64k in size and occupies the address space starting from 18'h10000. The input/output space starts from 18'h3FF80. Design in the form of Boolean equations the address decoder circuit which produces the RAM_1, RAM_2, ROM_1 and INPUT_OUTPUT enable signals.

[8]

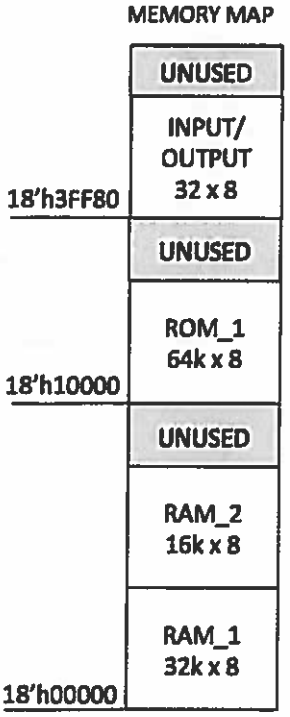


Figure 1.5

2. A finite state machine controlled by the clock signal CLK has two inputs P and Q, and two outputs X and Y. All signal transitions occur shortly after the rising edge of CLK. The signal P contains high pulses lasting for exactly one clock cycle that are separated by one or more clock cycles. The signal Q is similar to P, but pulses on P and Q never occur together, i.e. P and Q are never high simultaneously. *Figure 2.1* shows a typical sequence of the input signals P and Q.

The finite state machine produces the output X, which detects two or more consecutive pulses occurring on P without being cancelled by a pulse on Q. For example, referring to *Figure 2.1*, pulse "a" on P is cancelled by pulse "b" on Q, and X stays low. However, pulses "c" and "d" are two consecutive pulses on P. Therefore X goes high at the next rising edge of CLK, producing the output pulse "e" on X. Pulse "g" on Q causes X to reset to zero on the next rising edge of CLK. Similarly Y goes high whenever two or more pulses arrive on Q and is not cancelled by a pulse on P. For example, pulses "g" and "h" cause Y to go high at "j". Pulse "i" on P resets Y to zero. Pulses "k" and "m" on Q cause Y to go high at n, which is then reset by pulse "o" on P.

(a) Draw a state diagram for a finite state machine that implements the above specification. Remember that P and Q are never high together.

[15]

(b) Specify the design of your finite state machine in Verilog HDL using one-hot state encoding.

[15]

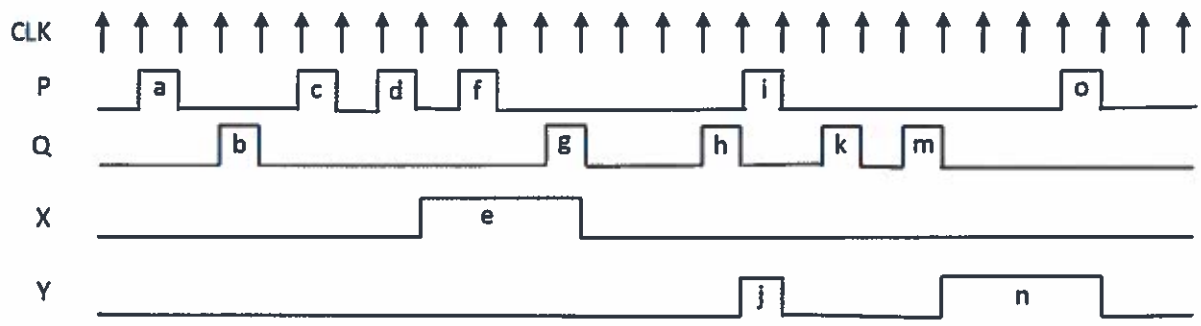


Figure 2.1

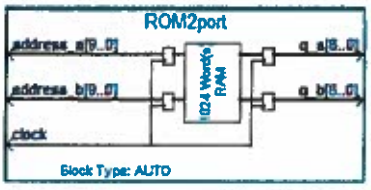
3. A dual-port 1024 x 9-bit ROM shown in *Figure 3.1* is used to store the coefficient of one cycle of a sinewave in 2's complement form. It is required to produce a pair of sinusoidal signals exactly $\pi/2$ radians apart (they are known as quadrature signals).

a) Design in block diagram form a circuit that produces these quadrature signals with the signal frequency as close to 1kHz as possible. You may assume that a clock signal at 50MHz is available and the dual-port ROM is already provided.

[15]

b) Design in Verilog HDL a module to produce two 9-bit output signals *sine_sig* and *cosine_sig* which are the quadrature signals at approximately 1kHz. State to 4 significant digits the frequency of your outputs.

[15]




```
module ROM2port (
   address_a,
   address_b,
   clock,
   q_a,
   q_b);

   input [9:0]  address_a;
   input [9:0]  address_b;
   input   clock;
   output  [8:0]  q_a;
   output  [8:0]  q_b;
```

Figure 3.1