

Sistemes Digitals

Examen Final. 14 de juny de 2018

Temps per a la resolució: 2 hores.

1 Problema: Temporització d'una aspiradora (50%)

L'objectiu d'aquest exercici és dissenyar un sistema de temporització de l'aspiradora d'una estació de rentat. Els usuaris poden pagar amb monedes d'un euro i tenen un minut de servei per cada euro introduït. Per tal d'evitar cues, es restringeix el temps màxim d'ús a 10 minuts.

El sistema compta amb una entrada que es manté a '1' durant uns quants mil·lisegons cada vegada que l'usuari introdueix una moneda. En el moment en que es detecta una moneda s'ha d'activar a '1' la sortida de control de l'aspiradora i iniciar la temporització corresponent. En cas que s'introdueixin monedes una vegada iniciat l'aspirat, s'incrementarà el temps de servei.

El sistema ha de disposar d'una sortida (vector de 10 bits) que indiqui quants segons queden per finalitzar el servei i una sortida d'1 bit que emeti un to de freqüència 100 Hz durant els últims 5 segons de servei.

Una vegada finalitzat el temps d'aspirat, les sortides s'han de mantenir a '0' fins que es torni a introduir una moneda.

Per implementar el sistema es disposa d'una entrada de rellotge d'1 kHz.

Dissenyu un mòdul VHDL que ofereixi el funcionament descrit. Les entrades han de ser `moneda` i `clk` i les sortides `aspiradora`, `segons` i `timbre`.

Per resoldre l'exercici responeu a les següents preguntes:

- Descriviu la forma correcta de gestionar l'entrada `moneda` si pot canviar de forma asíncrona al sistema.
- Observeu que la freqüència del rellotge és molt elevada en comparació a la temporització de minuts que es demana. Proposeu una manera per temporitzar el sistema sense utilitzar senyals interns de més de 10 bits.
- Definiu el diagrama de blocs del sistema complet.
- Escriviu l'`entity` i `architecture` del sistema (ha de ser sintetitzable i síncron). Definiu acuradament cadascun dels senyals interns del vostre disseny.
- Dibuixeu un cronograma detallat que verifiqui el correcte funcionament del sistema.

2 Test (30 %)

Les respostes errònies descompten 0.1 punts.

1. Indiqueu quina de les següents sentències és falsa respecte el llenguatge VHDL:

- a) La sentència `with ... select` es pot escriure dins d'un `process`
- b) L'ordre amb què s'escriuen les sentències dins d'un `process` és important
- c) Un `process` és una sentència concurrent

d) La sentència `wait` no és sintetitzable

2. Indiqueu quina de les següents afirmacions és certa si escrivim les següents sentències concurrents:

```
sA <= '0';  
sA <= 'H';
```

- a) El senyal `sA` valdrà '0' durant uns instants de temps i després prendrà el valor 'H'
- b) El senyal `sA` es mantindrà a '0'
- c) El senyal `sA` prendrà un valor indeterminat
- d) L'ordre de les sentències concurrents és important

3. Indiqueu quina de les següents afirmacions és falsa respecte el llenguatge VHDL:

- a) Un senyal del tipus `signed` de 8 bits pot pendre el rang de valors de -128 a 127
- b) Per ampliar el nombre de bits (i mantenir el valor) d'un senyal del tipus `unsigned` n'hi ha prou amb afegir-hi zeros a l'esquerra
- c) El complement a 2 s'utilitza per codificar valors negatius amb representació binària
- d) A l'hora de realitzar sumes binàries amb signe, necessitarem un bit més de precisió quan la operació tingui `carry` de sortida

4. Indiqueu quina de les següents afirmacions és certa respecte el codi VHDL de sota:

```
architecture behav of cronos is  
  signal flag1, flag2 : std_logic := '0';  
  signal Q : unsigned(1 downto 0) := "00";  
  
  process (clk) is  
  begin  
    if rising_edge(clk) then  
      Q <= "00";  
      Q <= Q + 1;  
      flag1 <= '1';  
      if Q = "10" then  
        Q <= "00";  
        flag1 <= '0';  
      end if;  
    end if;  
  end process;  
  flag2 <= '1' when (Q > "01") else '0';  
end behav;
```

- a) `flag2` valdrà sempre '0'
- b) Sempre que `flag2` està actiu ('1'), `flag1` també ho està
- c) El comptador `Q` es reinicia cada 4 cicles de `clk`
- d) `flag1` valdrà sempre '1'

5. Indiqueu quina de les següents afirmacions és certa respecte els sistemes digitals:

- a) El GHDL és una eina específica per sintetitzar sistemes digitals

- b) El camí crític és aquell que presenta un major retard combinacional entre dos registres
- c) El VHDL és un llenguatge de descripció de software
- d) El temps de hold d'un flip-flop és el retard que hi ha des de que arriba el flanc de pujada del rellotge fins que s'actualitza la sortida

6. Indiqueu quina de les següents afirmacions és falsa respecte el Mini AVR:

- a) La unitat de control s'encarrega de decodificar les instruccions i activar els senyals de control cap a la resta de mòduls
- b) El registre d'estat s'utilitza per emmagatzemar l'estat de l'última operació de la ALU
- c) L'escriptura als registres és síncrona
- d) La instrucció *Add with Carry* (ADC) s'anomena així perquè permet activar el bit de carry del registre d'estat

7. Indiqueu quina de les següents afirmacions és certa respecte les instruccions del Mini AVR:

- a) La instrucció LDI permet carregar el valor d'una constant k a una adreça de la memòria de dades (RAM)
- b) La instrucció ST permet carregar el valor d'una constant k a una adreça de la memòria de dades (RAM)
- c) La instrucció OUT permet carregar el valor d'una constant k a una adreça dels ports d'entrada / sortida
- d) La instrucció LD permet carregar el contingut d'una adreça de la memòria de dades (RAM) a un registre

8. Donat el següent codi ensamblador, indiqueu quina de les següents afirmacions és falsa:

```
LDI r16, x00
IN r17, x01
EOR r17, r16
BRNE -3
```

- a) La instrucció BRNE -3 saltarà en cas que el resultat de la EOR sigui diferent de 0
- b) Si substituïm la instrucció EOR per una OR no hi haurà cap canvi en l'execució del codi
- c) Es realitzarà un bucle mentre pel port d'entrada 1 hi arribi un valor diferent de 0
- d) La instrucció BRNE -3 permet saltar a la instrucció LDI

9. Donat el següent codi ensamblador, indiqueu quina de les afirmacions és certa:

```
LDI r26, x01
LDI r27, x00
LDI r16, x02
ST X, r16
```

- a) En l'adreça 1 de la memòria RAM s'hi ha carregat el valor x02
- b) El codi seria equivalent si en comptes de la instrucció ST utilitzessim la MOV
- c) El codi seria equivalent si en comptes dels registres r26 i r27 utilitzessim r20 i r21
- d) El registre r16 acaba prenent el valor x01

10. Indiqueu quina de les següents afirmacions és certa respecte el timer del Mini AVR:
- a) El comptador de programa del Mini AVR manté el seu valor mentre el timer està comptant
 - b) Cada vegada que modifiquem el `timer_limit` es reinicia el comptador del timer
 - c) El prescaler es modifica a través de la instrucció `OUT`
 - d) Per llegir el `timer_state` s'accedeix a l'adreça `x02` de la memòria RAM

3 Qüestions (20 %)

11. The Mini AVR was applied in the last lab exercise in order to implement a melody by controlling four synthesizers. Describe in detail the main ROM program of your design. (English: 100 %, Catalan: 50 %).
12. The Mini AVR has some instructions for jumping a specific number of ROM positions. Explain them and describe their differences. (English: 100 %, Catalan: 50 %).