



מערכות תוכנה להנדסאים - הנדסת אלקטרוניקה

הנחיות לנבחנים

- א. משך הבחינה : ארבע שעות
- ב. מבנה השאלון ומפתח הערכה : בשאלון 10 שאלות, יש לענות על 5 שאלות בלבד בהתאם לפירוט הבא :
חלק א' - שאלות 1-4 : יש לענות על 2 שאלות בלבד.
חלק ב' - שאלות 5-10 : יש לענות על 3 שאלות בלבד.
כל השאלות שוות בערך - 20 נקודות.
- ג. חומר עזר מותר בשימוש :
אין להעביר חומר עזר בין הנבחנים.
- ד. הוראות מיוחדות :
1. יש להשאיר את העמוד הראשון במחברת הבחינה ריק.
בגמר הבחינה יש לרשום בעמוד זה את מספרי השאלות אותן ברצונך שמעריך הבחינה יבדוק. לא יבדקו שאלות עודפות על הנדרש.
2. אם צוין בשאלון הבחינה שאלות המכילות השלמת טבלה יש לרשום את הטבלה במחברת הבחינה.
- ה. הוראות כלליות :
1. יש לקרוא בעיון את כל השאלות.
2. את התשובות יש לכתוב במחברת הבחינה בכתב יד ברור ונקי. (גם בכך תלויה הערכת הבחינה).
3. יש לכתוב בעט בלבד.
4. יש להתחיל כל תשובה בעמוד חדש תוך ציון מספר השאלה /הסעיף. אין צורך להעתיק את השאלה/סעיף.
5. טיוטה תעשה במחברת הבחינה בלבד. יש לכתוב "טיוטה" ולהעביר קו על התשובה כדי שלא תבדק.
6. יש להציג פתרון מלא ומנומק הצגת תשובה סופית ללא דרך הפתרון לא תזכה לניקוד.
7. אם להערכתך חסרים נתונים, יש לציין זאת במחברת. הינך רשאי להניח הנחות, אך חובה עליך לנמק אותם.

בהצלחה !

חלק א': תוכנה C/C++

יש לענות על 2 שאלות בלבד מתוך 4 השאלות הבאות (שאלות 1-4).

שאלה מספר 1 (20 נקודות)

יש לכתוב מחלקה oddArray המכילה מערך של SIZE=100 מספרים שכולם אי-זוגיים (א) (7 נקודות) יש לרשום את הגדרת המחלקה. המחלקה צריכה להכיל את הפונקציות הבאות:

- בנאי ברירת מחדל (default constructor) – מאתחל את כל האברים ל 1.
- בנאי המקבל כקלט מערך בגודל SIZE.
- פונקציה המאפשרת שינוי ערך במיקום (index) מסויים.
- פונקציה המדפיסה את תכולת המערך למסך

בסעיף זה אין לממש פונקציות

(ב) (10 נקודות) יש לממש את הבנאי השני, המקבל כקלט מערך בגודל SIZE ומעתיקו למחלקה. כאשר הפונקציה פוגשת באיבר זוגי, היא תחלק אותו ב 2 עד שיהפך לאי-זוגי. אם המספר הוא 0, הפונקציה תציב 1.

(ג) (3 נקודות) יש לרשום מה תדפיס התוכנית בפונקציה הבאה:

```
main()
{
    int a[SIZE] = {4, 14, 15, 16};
    oddArray b(a);
    b.print();
    return 0 ;
}
```

שאלה מספר 2 (20 נקודות)

לפניך מספר מקטעי תוכניות בשפת C. i, j הם משתנים מסוג `int`. `str` הוא משתנה מטיפוס `char*` (באורך מספיק). בכל אחד מהסעיפים להלן יש לרשום באופן ברור את הפלט המתקבל בסיום הקטע. כמו כן יש לנמק בכל סעיף: כיצד הגעת לתשובה זו?
(א) (7 נקודות) מה יודפס בסיום הקטע הבא?

```
strcpy(str, "ABCD");  
*(++str)='\0';  
--str;  
cout<<str<<endl;
```

(ב) (6 נקודות) מה יודפס בסיום הקטע הבא?

```
j = 2;  
for(i=5; i<100; i=i*2 + 1){  
    cout<<i%j<<' ';  
    j=i;  
}  
cout<<endl;
```

(ג) (7 נקודות) מה יודפס בסיום הקטע הבא?

```
strcpy(str, "ABCD");  
i=str[3]-str[1];  
cout << (str+i) << endl;
```

שאלה מספר 3 (20 נקודות)

- כתוב לכל אחד מהסעיפים הבאים פונקציה (נפרדת!) ב ++c הבודקת / מדפיסה את המבוקש.
כל אחת מהפונקציות מקבלת כפרמטר מערך שלמים (int) ואת גודלו.
חובה לממש לפחות אחד מהסעיפים ברקורסיה (3 נקודות).
- א) (6 נקודות) הפונקציה תחזיר true אם יש במערך (המועבר כפרמטר) שני מספרים סמוכים שסכומם מתחלק ב 10 , אחרת תחזיר false.
- ב) (6 נקודות) פונקציה שתחזיר את סכום האברים המתחלקים ב 10 במערך.
- ג) (8 נקודות) יש להדפיס את כל המספרים שלפניהם יש מספר שלילי, ואחריהם מספר חיובי.

שאלה מספר 4 (20 נקודות)

נתונה המחלקה הבאה המייצגת מערך מספרים ממויין:

```
class SortedArray{
    int *arr;
    int s;
    void sort();
public:
    SortedArray(int *array, int size);
    void print()const;
    void DeleteDoubles();
};
void SortedArray::print() const {
    for (int i=0; i<s; ++i)
        cout<<arr[i]<<" ";
    cout<<endl;
}
void main(){
    int arr[]={-7, -4, -1, 2, 2, 3, 4, 9 };
    SortedArray a(arr,8);
    a.print();
    a.DeleteDoubles();
    a.print();
}
```

- א) (10 נקודות) יש לממש את הפונקציה DeleteDoubles המוחקת מן המערך את הכפילויות (כך שכל איבר ישאר פעם אחת בלבד). חשוב לזכור לעדכן את גודל המערך.
- ב) (5 נקודות) במה מועילה העובדה שהמערך ממויין? האם הפונקציה שכתבת תעבוד באותה המהירות כאשר המערך איננו ממויין?
- ג) (5 נקודות) מה תדפיס התוכנית?

VHDL : חלק ב'

יש לענות על 3 שאלות מתוך 6 השאלות הבאות (שאלות 5-10).

שאלה מספר 5 (20 נקודות)

נתונה התוכנית הבאה :

- (א) (10 נקודות) יש לצייר בפירוט את החומרה שתיווצר מהתוכנית הנתונה ולצרף הסברים מתאימים לתרשים. נא להקפיד על תרשים ברור!
- (ב) (10 נקודות) יש לכתוב ארכיטקטורה זהה פונקציונלית, ע"י פקודת Generate אחת ליצירת הדלגלים. יש להניח קיומו של דלגל מסוג D ולכן אסור לכתוב קוד ל- DFF. חובה להשתמש באותם שמות משתנים אך מותר להשתמש בוווקטורים במקום בסיביות בודדות.

```
library ieee; use ieee.STD_LOGIC_1164.all;
library ieee; use ieee.STD_LOGIC_ARITH.all;
library ieee; use ieee.STD_LOGIC_UNSIGNED.all;

entity test5A_Process is
    generic(size: integer:=3);
    port(d,clk,rst: in std_logic;
         q: out std_logic_vector(size-1 downto 0));
end;

architecture to_hardware of test5A_Process is
    signal d0,d1,d2: std_logic;
    signal q0,q1,q2: std_logic;
begin
    process(rst,clk)
    begin
        if rst='1' then
            q0<='0'; q1<='1'; q2<='1';
        elsif rising_edge(clk) then
            q0<=d0; q1<=d1; q2<=d2;
        end if;
    end process;
    d0<= d XOR q2;
    d1<= d0 XOR q0;
    d2<= d1 XOR q1;

    q<= q2 & q1 & q0; --! שורה זו בחומרה!
end;
```

אין צורך ל"מחש" שורה זו בחומרה! --

ענית על שאלה זו ? נא לרשום את מספרה באופן בולט בעמוד הראשון של המחברת

שאלה מספר 6 (20 נקודות)

נתונה התוכנית הבאה, כאשר בכל הסעיפים נתון: `input = "1000"`. יש להשלים את הטבלאות להלן וגם לצייר את צורות הגלים של הפלט `div_by` ביחס ל-`count`. ניתן להניח שניתן איפוס קצר לפני הופעת מחזור השעון הראשון!

```
library ieee;
use ieee.STD_LOGIC_1164.all;
use ieee.STD_LOGIC_ARITH.all;
use ieee.STD_LOGIC_UNSIGNED.all;
entity div is
    generic (size: NATURAL:= 4);
    port (clk: in BIT; rst: in BIT;
          input: in std_logic_vector(size-1 downto 0);
          div_by: out std_logic);
end div;
architecture div of div is
    signal count: std_logic_vector(size-1 downto 0);
begin
    process (rst, clk)
    begin
        if rst = '1' then
            count<=(others=>'0');
            div_by<='1';
        elsif clk'event and clk = '1' then
            if count = input - 1 then
                count<=(others=>'0');
                div_by<='0';
            else
                if count > SHR(input,"01")-1 then
                    div_by<='0';
                else
                    div_by<='1';
                end if;
                count<=count+1;
            end if;
        end if;
    end process;
end;
```

(א) (10 נקודות) עבור התוכנית לעיל כמו שהיא, יש לצייר טבלה במחברת ע"י רישום ערכו של `div_by` ביחס ל-`count` וגם לצייר את צורות הגלים:

count	div_by
0	
1	
2	
3	
4	
5	
6	
7	

(ב) (10 נקודות) משנים את השורה המודגשת בתוכנית, ובמקומה רושמים את השורה הבאה:

```
if count > conv_integer(input)/2 then
```

מומלץ לרשום לעצמך שורה זו במילים, כדי לוודא ששמת לב לשינויים!
יש לצייר טבלה חדשה במחברת ע"י רישום ערכו של `div_by` ביחס ל-`count` וגם לצייר את צורות הגלים:

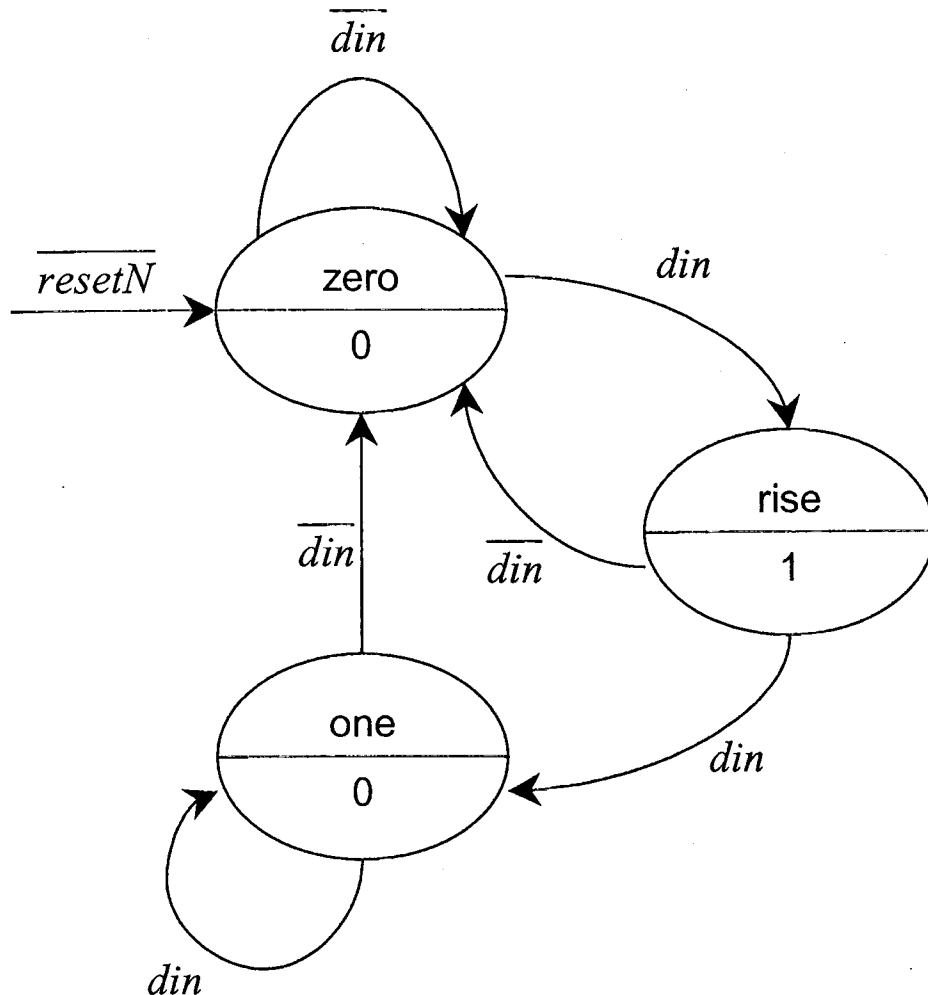
count	div_by
0	
1	
2	
3	
4	
5	
6	
7	

הערות:

1. הפונקציה `SHR` קיימת בספריות המצורפות לתוכנית ומבצעת הזזה ימינה.
2. הפונקציה `conv_integer` קיימת בספריות המצורפות לתוכנית ומתרגמת מווקטור ל-`integer`. לידיעתך, מותר להשתמש בחילוק ב-2 עבור הסוג `integer`.
3. הרבה טועים בשאלה זו מאחר והם "שוכחים" שמדובר בדלגלים, בבקשה נא לא לשכוח!

שאלה מספר 7 (20 נקודות)

א) (10 נקודות) במכונת מצבים זו, הכניסה resetN היא כניסה אסינכרונית הפעילה בנמוך, והמוצא dout לקרא מצבים של המכונה:



יש לתאר את מכונת המצבים הנ"ל באמצעות שני תהליכים. תהליך אחד יתאר את הדלגלים (State Registers) ותהליך נוסף יתאר את המערכת הצירופית של המכונה (לוגיקת העירור למצב הבא ולוגיקת היציאה).

יש לייצג את שמות המצבים בצורה סימבולית כמופיע בתרשים (Enumerated Data Type), ולכתוב את הקוד (הישות והארכיטקטורה) באופן מלא וברור.

(ב) (10 נקודות) יש להשלים את הממשק (Entity) להלן, כך שישמש עירור למכונת המצבים. הקוד שלך צריך להיות **רכיב עצמאי** שיפעיל את התוכנית לעיל.
הדרכה: יש לכתוב ארכיטקטורה עם תהליכים ו/או פקודות מקביליות, אך אסור לרשום חיווטים כלשהם ואסור להשתמש בפקודת Assert (יורדו נקודות)!

יש להסביר כיצד העירור שכתבת בודק את כל מצבי המכונה.

```
entity test7B is
    port(clk, resetN, din: OUT std_logic;
          dout: IN std_logic);
end;
```

שאלה מספר 8 (20 נקודות)

נא לקרוא לנתח ולהבין תוכנית זו.

(א) (5 נקודות) יש להסביר (עד 2 משפטים) מה תוכנית זו עושה, ולהדגים ע"י הסבר מפורט של 2 דוגמאות, שתבחר לערכי inp כרצונך.

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;
ENTITY test8B_process IS
    generic(n: integer:= 4);
    PORT(inp: IN STD_LOGIC_vector(n-1 downto 0);
          otp: OUT integer range 0 to 2**n -1);
END;

ARCHITECTURE arc OF test8B_process IS
BEGIN
    process(inp)
        variable var, temp: integer;
    begin
        var:= 0; temp:= 1;
        for i in 0 to inp'length -1 loop
            if inp(i)= '1' then
                var:= var + temp;
            end if;
            temp:= 2*temp;
        end loop;
        otp<= var;
    end process;
end;
```

ב) (5 נקודות) יש לכתוב פרוצדורה בתוך package body שתבצע בדיוק אותה פעולה שמבצע התהליך לעיל. נא להשתמש בקוד להלן!

```
library ieee;
use ieee.STD_LOGIC_1164.all;
use ieee.STD_LOGIC_ARITH.all;
use ieee.STD_LOGIC_UNSIGNED.all;

package pack8 is
    procedure vec2int(Signal vec: in std_logic_vector;
                     Signal int: out integer);
end;
package body pack8 is

end;
```

ג) (10 נקודות) כעת, יש לכתוב את הארכיטקטורה לעיל מחדש, כך שישתמש בפרוצדורה שכתבת. עליך האחריות שהתוכנית כולה תעבוד, ותעשה בדיוק אותו הדבר.

שאלה מספר 9 (20 נקודות)

(א) 10 נקודות) יש לצייר את החומרה שתיווצר מתוכנית זו:

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;
ENTITY test09A_What IS
    PORT(clk, rst, inp: IN STD_LOGIC;
          otp: OUT std_logic);
END;

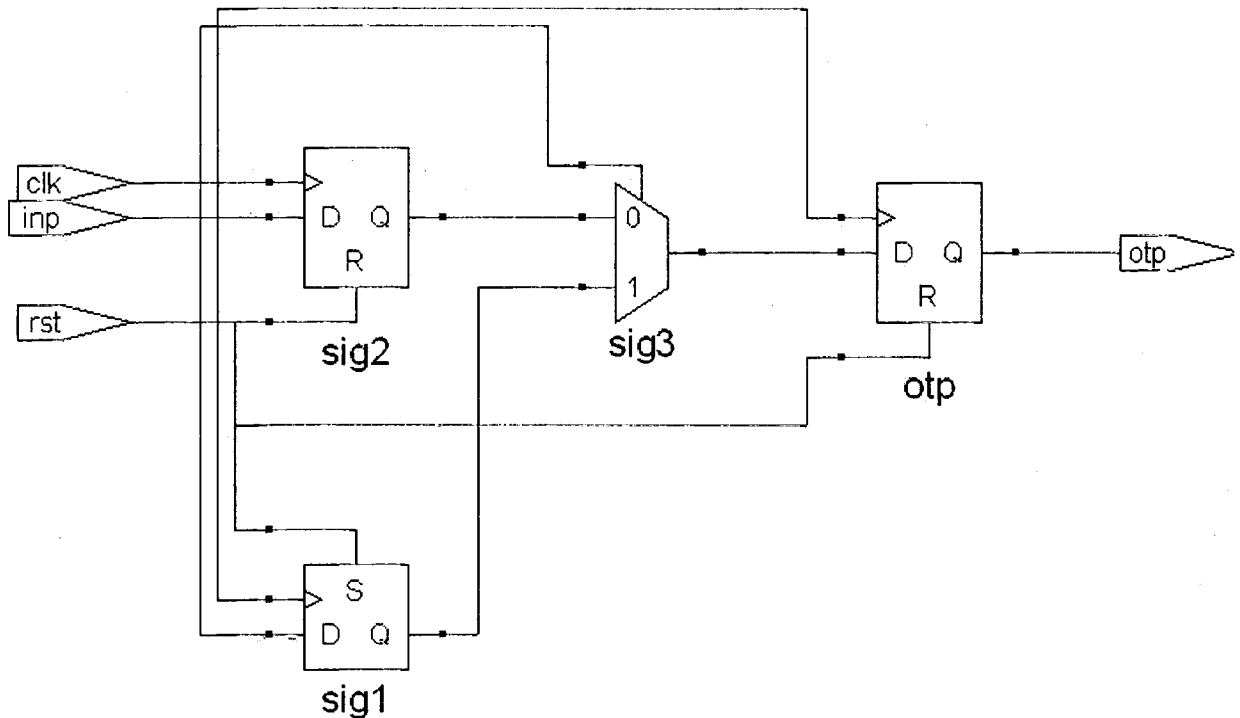
ARCHITECTURE arc OF test09A_What IS
    SIGNAL a, b: std_logic;
BEGIN
    otp<= a WHEN inp='1' ELSE
        b;
    process(clk, rst)
    begin
        if rst= '1' then
            a<= '0'; b<= '1';
        elsif rising_edge(clk) then
            b<= a; a<=inp;
        end if;
    end process;
end;
```

(ב) 10 נקודות) יש לצייר את החומרה שתיווצר מתוכנית זו:

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;
ENTITY test09B_What IS
    PORT(clk, inp1, inp2: IN STD_LOGIC;
          otp: OUT std_logic);
END;
ARCHITECTURE arc OF test09B_What IS
    SIGNAL a, b: std_logic;
BEGIN
    process(clk)
    begin
        if rising_edge(clk) then
            a<= inp2; otp<= b;
        end if;
    end process;
    b<= inp1 WHEN inp2= '0' ELSE
        a;
end;
```

שאלה מספר 10 (20 נקודות)

בשאלה זו, אסור לכתוב קוד המכיל חיווטים כלשהם.
(א) (10 נקודות) יש לכתוב קוד המכיל בדיוק תהליך מדורבן שעון אחד (Clocked Process), המתאר את התרשים הבא:



הערה חשובה: בדלגלים שבתרשים R=Reset, S=Set, ו-sig1, sig2, otp הם מוצאי הדלגלים, sig3 הוא מוצא המרבב. חובה להשתמש בסימונים שעל התרשים! ניתן להניח שכל האותות מסוג bit. המשתנה inp נכנס גם לדלגל שמוצאו sig2 וגם לדלגל שמוצאו sig1 וגם הבורר של המרבב הוא inp.

(ב) (10 נקודות) יש לכתוב תוכנית פשוטה כרצונך, ליצירת מרבב 4:1 ע"י פקודת With...Select, כאשר האותות כולם מסוג Bit.