**תרגיל בית 1 - יסודות המחשוב**

שימו לב:

**בכל השבבים שתממשו עליכם לשרטט ראשית את השבב ואחר כך לכתוב מימוש HDL. נא להוסיף לשרטוט את שמות החוטים הפנימיים-את השרטוטים צריך לצרף לקובץ WORD.**

חזרנו בקורס על הדרך שבה נעשה חיבור בינארי. ראינו שמבצעים חיבור של ספרות במיקומים מקבילים, כשבחיבור זה התוצאה יכולה להיות גדולה מאחת. כשהתוצאה גדולה מאחת מעבירים 2 לחיבור של הספרה הבאה שם ערכה יהיה 1. להעברה הזו קוראים carry. כיוון שניתן לחלק את משימת החיבור לתת משימות של חיבור כל ספרה במיקומים מקבילים בנפרד, ראינו שניתן לבנות מחבר על ידי בנייה של תת יחידות שכל אחת כזאת מקבלת שתי ספרות לחיבור, carry מחיבור הספרה מימין, ומוציאה תוצאה ו-carry לספרה משמאל.

להלן סירטון חזרה על הדרך שבה מתבצע חיסור בינארי: <https://www.youtube.com/watch?v=h_fY-zSiMtY>. בדומה לחיבור בינארי ניתן לבנות יחידת חיסור מתת יחידות שכל אחת מהן אחראית לחיסור בין שתי ספרות במיקומים מקבילים. כמו במחבר שבו היחידות מקבלות carry ומעבירות carry ליחידות הבאות, במחסר היחידות מקבלות בקשת borrow ומעבירות בקשת borrow ליחידה הבאה.

בתרגיל זה נתחיל בסעיף א ו-ב במימוש יחידת החיסור לספרה בודדת. בסעיף ג' נבנה יחידה שמאפשרת בחירה בין חיבור או חיסור עבור ספרה בודדת, ובסעיף ד', נבנה יחידה שמאפשרת חיבור או חיסור של מילה בגודל 8 ביט.

1. ממשו ב hdl צ'יפ בשם HalfSub (המזכיר את הצ'יפ HalfAdder) המוגדר באופן הבא:

CHIP HalfSub {

IN a,b;

OUT diff,bOut;

..

פעולת הצ'יפ הינה :(מדובר ברכיב **חצי מחסר** שמחסר בין שני ביטים)

diff=a-b  
bOut=BORROW(לווה)

התחילו מבניית טבלת האמת של הצ'יפ.  
שימו לב שאם a=0 ו- b=1 מקבלים a-b=0-1=-1, לכן צריך ללוות ומקבלים  
2+0-1=1. כלומר שמקבלים את היציאות diff=1,1=bOut.

1. ממשו ב hdl צ'יפ בשם FullSub המוגדר באופן הבא:

(מדובר ברכיב **מחסר מלא** שמחסר בין 3 ביטים)

CHIP FullSub {  
 IN a,b,bIn;  
 OUT diff,bOut;  
 ..

פעולת הצ'יפ הינה :

diff=a-b-bIn  
bOut=BORROW(לווה)

שימו לב למצבים בהם צריך ללוות, כלומר להוציא borrow, למשל אם ערכו של a הוא 0, וערכו של b הוא 0, וערכו של bInהוא 1, אז נקבל a-b-Bin=-1. כיוון שכך צריך ללוות ומקבלים 2+a-b-Bin=1. כלומר שמקבלים את היציאות diff=1,1=bOut.

**יש להשתמש בצ'יפ שמימשתם בסעיף א' על מנת לממש את הרכיב הזה.**

1. ממשו ב-hdl צ'יפ בשם AdderSub המוגדר באופן הבא:

CHIP AdderSub {  
 IN a,b,control,cIn;  
 OUT res,cOut;  
 …

אם סיבית ה control=0 אזי מתבצע res=a+b+cIn ו- cOut הוא הנשא של הסכום .

אם סיבית ה control=1 אזי מתבצע res=a-b-cIn ו- cOut הוא הלווה של ההפרש.   
יש להשתמש לצורך הבניה בצ'יפ שמימשתם בסעיף ב' ובצ'יפ FullAdder שפותח בתירגול.

1. ממשו ב-hdl צ'יפ בשם AdderSub8 המוגדר באופן הבא:

CHIP AdderSub8{

IN a[8],b[8],control;

OUT res[8], cOut, ifZero;

המקבל שני מספרים 8 ביט כל אחד a ו-b וביט control.  
אם סיבית ה-control=0 אזי מתבצע res=a+b ו-cOut הוא הנשא של הסכום.  
אם סיבית ה-control=1 אזי מתבצע res=a-b ו-cOut הוא הלווה של ההפרש.

ממשו בנוסף דגל ifZero שערכו יהיה שווה 1 אם התוצאה שווה ל-0. (כלומר אם res=00000000 אז הדגל ifZero ערכו יהיה 1. אחרת, ערכו של ifZero יהיה 0.

לדוגמא:   
עבור: a=01110110, b=10110001, control=0.  
נקבל: res=00100111, cout=1, ifZero=0.

עבור אותו a,b ,ו- control=1,  
נקבל res=11000101, cOut=1, ifZero=0.

**הוראות הגשה:**

ההגשה **בזוגות בלבד**. ההגשה במודל לפי התאריך המעודכן בתיבת ההגשה. יש להגיש קובץ hw1.zip אשר מכיל:

1. קבצים למימוש ה CHIP-ים הבסיסיים (מהתירגולים):  
   Not.HDL, And.HDL, Or.HDL, Xor.HDL, Mux.HDL, DMux.HDL, Not16.HDL, And16.HDL, Or16.HDL, Mux16.HDL, Or8Way.HDL, Mux4Way16, Mux8Way16.HDL, DMux4Way.HDL, DMux8Way.HDL
2. קבצים למימוש חיבור (מהתירגולים):  
   HalfAdder.HDL, FullAdder.HDL, Add16.HDL, Inc16.HDL
3. קבצים למימוש השבבים הנ"ל (מתרגיל זה):  
    HalfSub.hdl , FullSub.hdl , AdderSub.hdl , AdderSub8.hdl

כל קובץ אשר תרצו להוסיף לשימוש בפתרון , חייב להיות מצורף קובץ hdl שלו .

1. יש לבדוק את הקבצים שלכם עם קבצי TST ו-CMP מוכנים שנמצאים בתיקייה .
2. קובץ WORD עם שמות + ת.ז. של המגישים ושרטוט של כל השבבים מסעיפים א' עד ד' שימו לב – **שני הסטודנטים מתבקשים להגיש את הקובץ**

**שאלות נא להפנות למרצה האחראית על התרגיל מרינה במייל** [**marin.minishin@braude.ac.il**](file:///D:\יסודות%20המחשוב\2024B\תרגילי%20בית\marin.minishin@braude.ac.il)**.**

**עירעורים על בדיקת התרגיל: בהמשך תעודכן כאן הדרך לערער על בדיקת התרגיל.**

**בהצלחה מכל צוות הקורס!**