



מעבדת ארכי- מעבדה 3

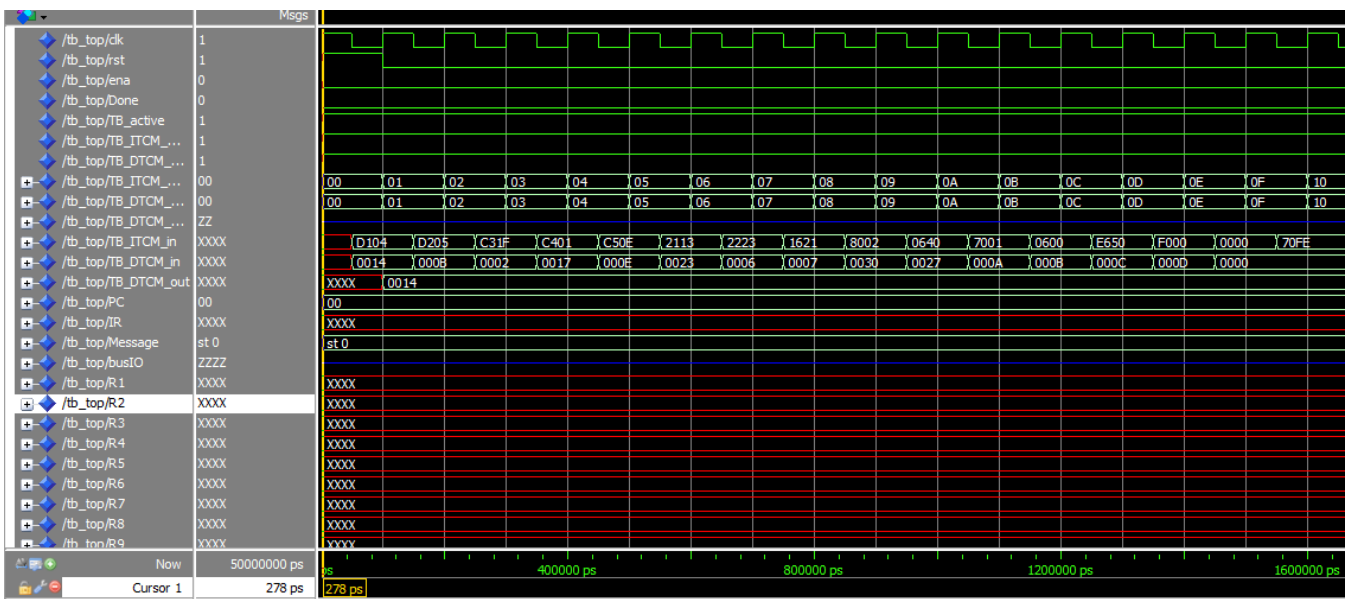
מגישים: יעקב קוזמינסקי 206511966 אור יעקבי 206827164

The system

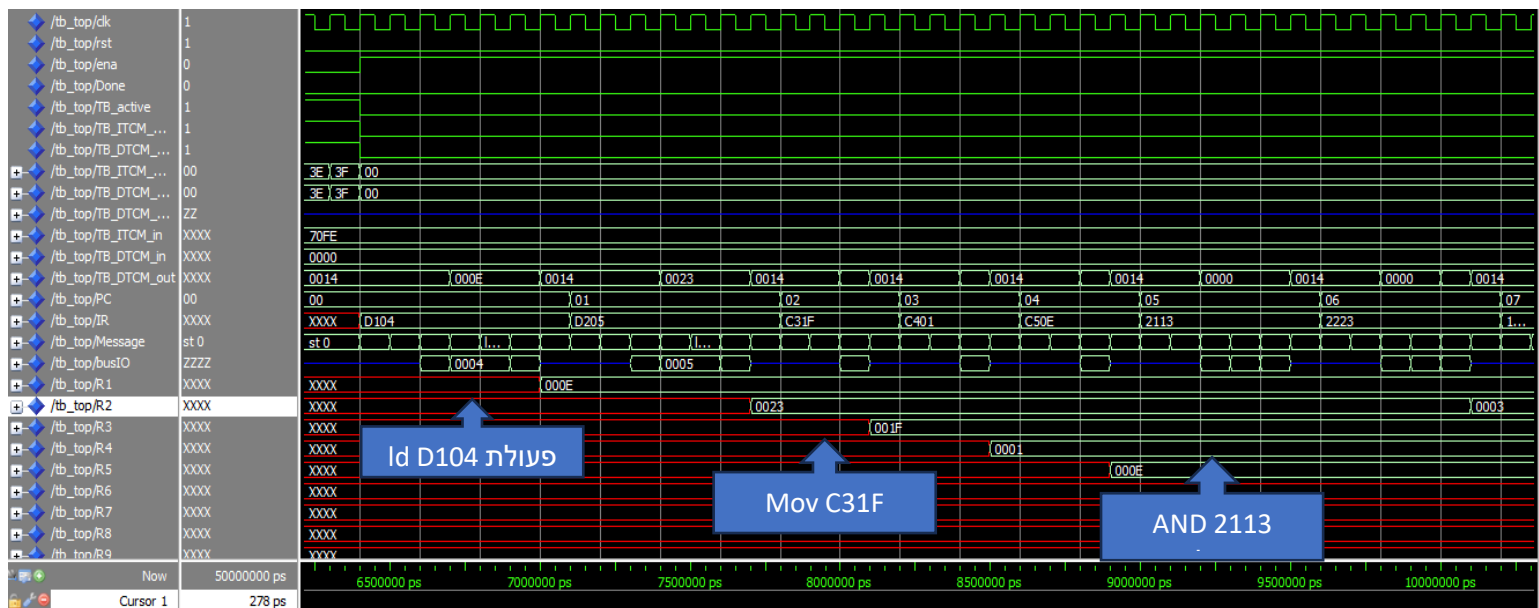
המערכת Top: כוללת 2 מערכות כנדרש Datapath ו CONTROL. המערכת קולטת בעזרת קובץ TB את התכנית מקובץ txt ואת הנתונים ל RAM בסיום טעינת הקבצים המערכת מודיעה על סיום טעינת הקבצים ומפעילה את control unit

2 המערכות עובדות בצורה כזו ש Control (המוח) אחראי על המערכת הסינכרונית השולחת קווי בקרה ל Datapath (הגוף) שמפעיל את המרכיבים שלו בהתאם לקווי בקרה אלה, ושולח חזר ליחידת control סטטוסים ומצבים עדכניים.

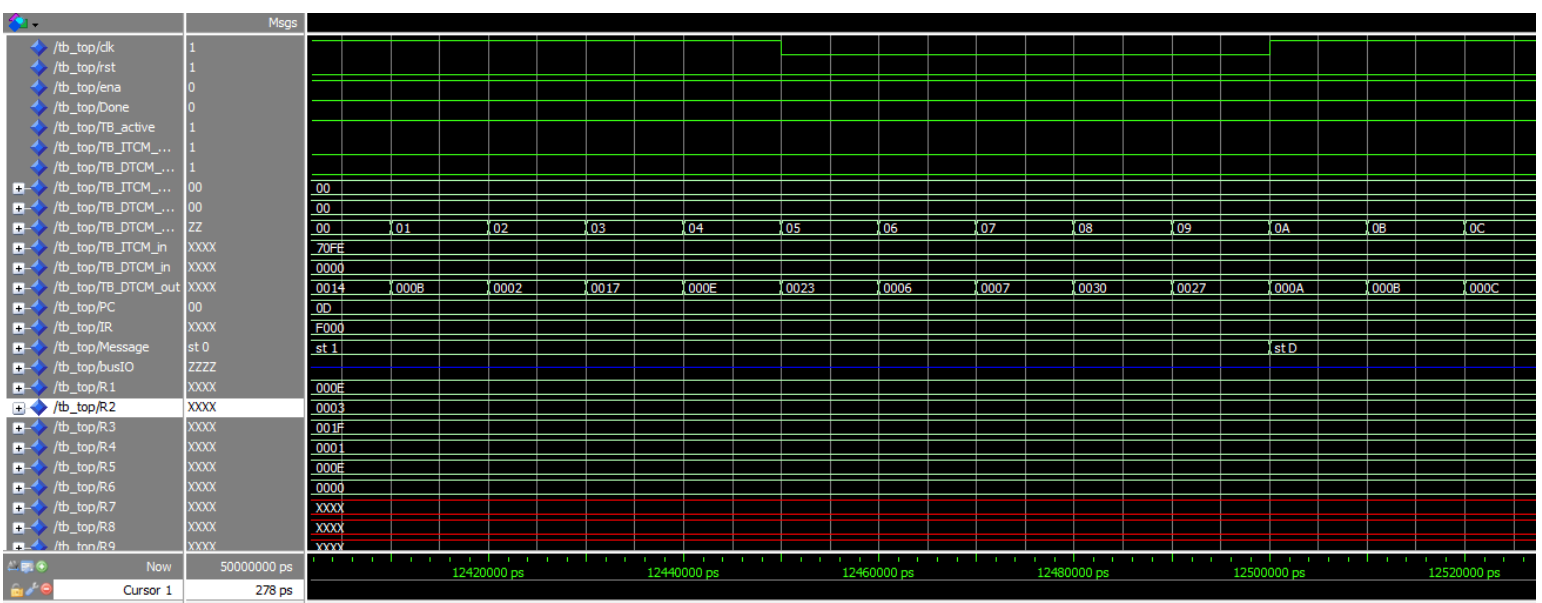
שלב טעינת הקבצים זיכרון תכנית וזיכרון נתונים:



שלב ריצת התכנית (חלק ממנה) :



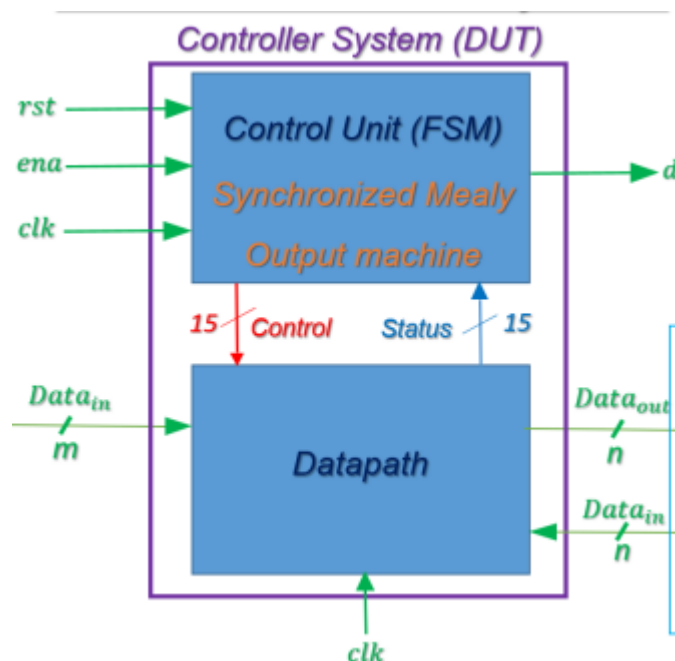
שלב כתיבת הפלט:



נשים לב שלב כתיבת הפלט נעשה בצורה א-סינכרונית, שכן אין סיבה שהכתיבה אשר מועתקת מהזיכרון לקובץ הטקסט תתבצע ע"י שעון.

בנוסף נשים לב כי הפעולה מתחילה רק כאשר $IR = F000$ כלומר הבקר מבצע את פעולת done. לאחר מכן כפי שכתוב בקוד הנתון במעבדה הפעולה נכנסת ללולאה אין סופית. אך הכתיבה מתבצעת רק בפעם הראשונה שהתבצעה פעולת done

* מגבלה נובעת מכך – בכל תכנית יכולה להיות רק פעולת done אחת כלומר רק פעם אחת נכתב פלט לקובץ הטקסט. שכן מדמה מערכת הפעלה המוציאה output באופן חד-חד ערכי לכל הרצה. (למשל לא יכול להיות מצב שנכתב ה output לטקסט ואז מתבצע שינוי בזיכרון ונדרוס את קובץ הטקסט באותה הרצה)

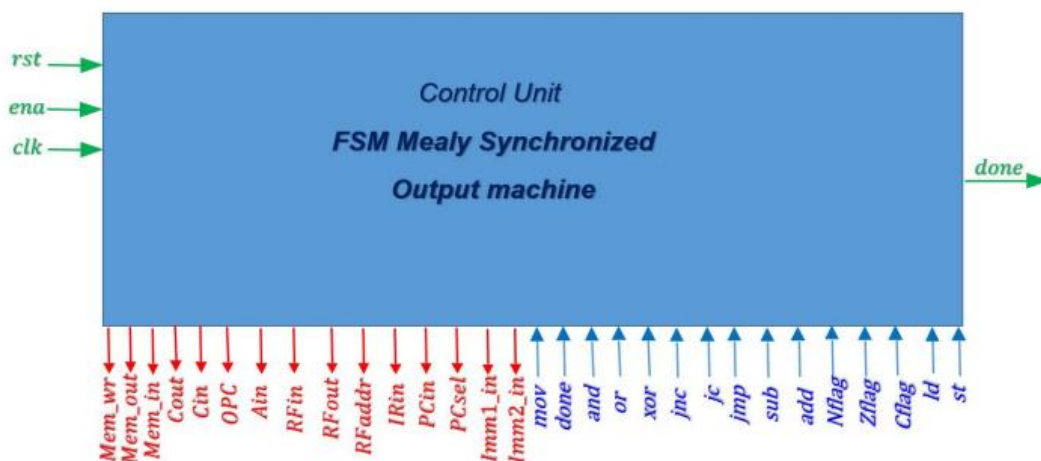


את קווי הבקרה הירוקים מבצעת "מערכת ההפעלה" צריבת הקוד על הבקר והוצאת הפלט, כאן את תפקיד זה מבצע TB TOP. ולו קווי הבקרה של הנתונים השעון enable & reset

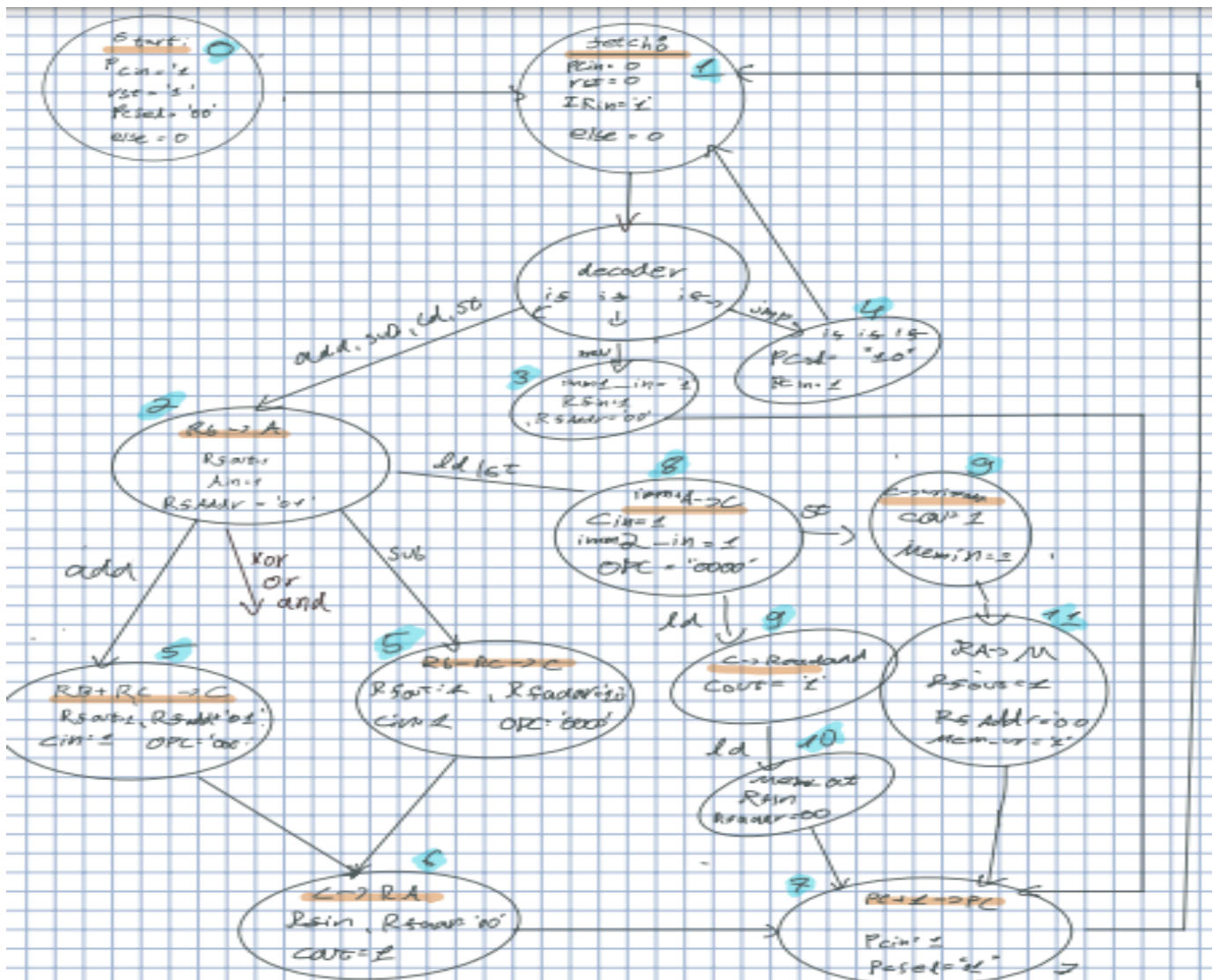
המערכת control : מערכת ה control מכילה בתוך מכונת מצבים סינכרונית FFSM אשר מדליקה את קווי הבקרה בהתאם לתוכנית שנקראה ב OPdecoder שנמצא במערכת datapath



בעזרת TB דימינו את שליחת הסטטוסים ל control ובדקנו האם ה control שולח את קווי הבקרה הנכונים לפי מערכת ה FFSM שבנינו.
ניתן לראות בקו הבקרה message את המצב בו נמצא ה control בצורה סינכרונית.
יש בדיקה כזו לכל הפעולות אותן נדרשנו לממש.



מערכת הFFSM בתיאור גרפי:



states חשובים:

State0 – מצב התחלתי בו אנו מהאפסים את PC.

State1 – IR מקבל את הפקודה (fetch)

Decode – state זה בנוי מפקודות if הבודק איזה פקודה הגיעה מהdecoder ומסעפת את מכונת המצבים לפי פקודות ה RTL שלהם.

כך למשל State 2

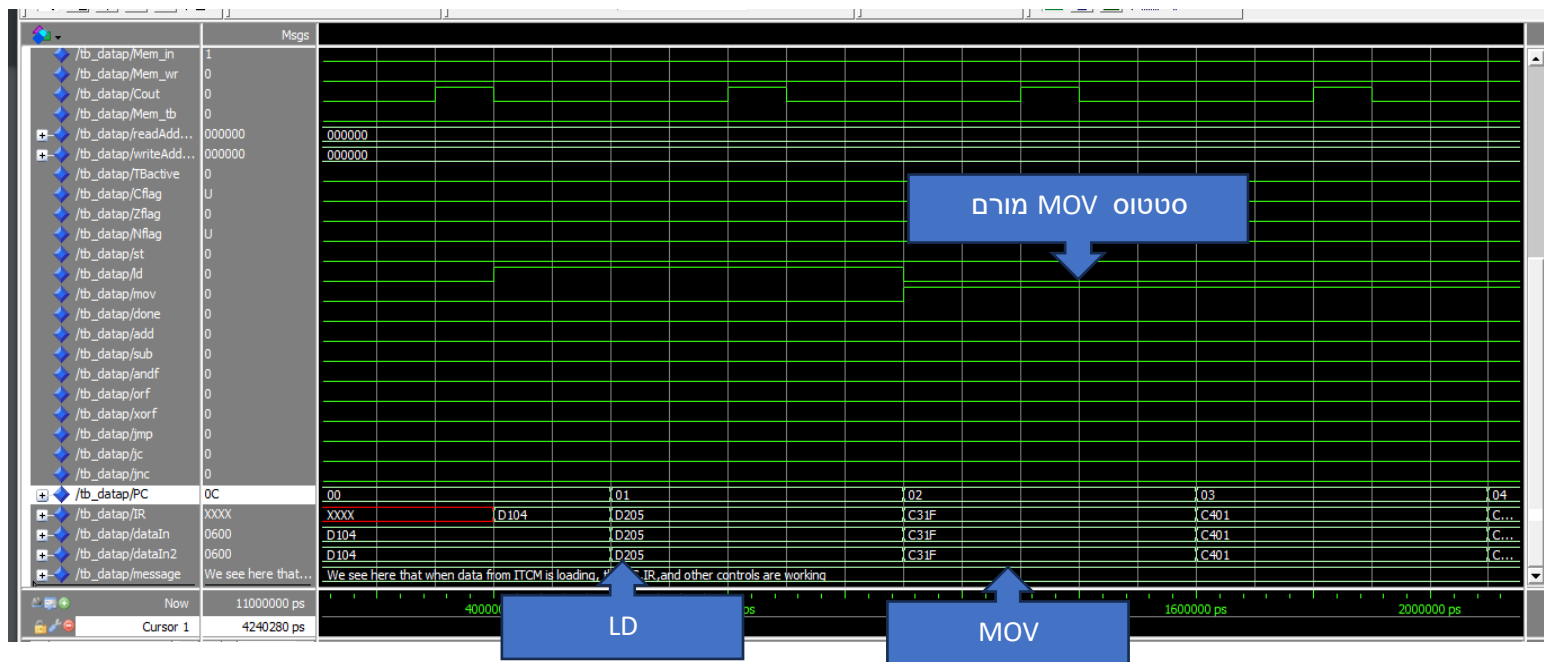
מתאר את השורה הראשונה בRTL של הפעולות האריתמטיות ופעולות LD ו ST שהיא העברת regA אל הרגיסטר A של הALU.

State 7 – state זה אנו מבצעים קידום של PC ב1 וחוזרים חזרה לשלב הfetch רוב הפעולות יגיעו למצב זה למעט פעולות הסתעפות שמגיעות לstate4 שם מתבצע חישוב ערך הPC הבא וחזרה לfetch.

מערכת DataPath :

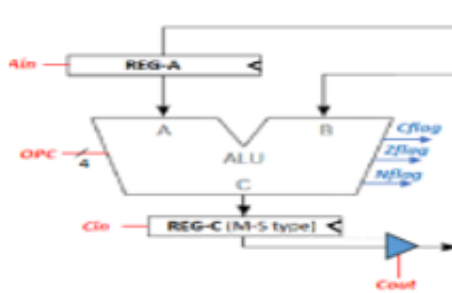
מערכת זו מכילה בתוכה את הרכיבים השונים של הבקר למשל כמו ה ALU יחידות זיכרון קובץ הרגיסטרים והבס.

מערכת זו מעבירה סטטוסים על מצב התכנית דרך ה decoder ל control ובחזרה מקבלת ממנו את קווי הבקרה המאפשרים העברת מידע בצורה נכונה בין הרכיבים השונים בהתאם לפקודה ומכונת המצבים.

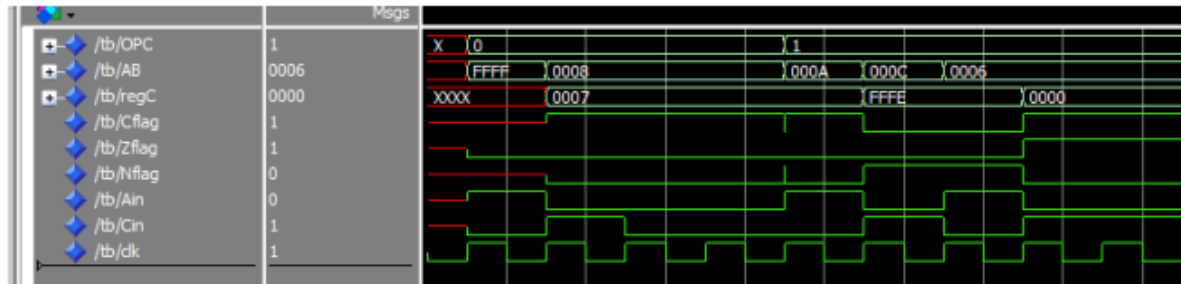


בדיאגרמת הגלים זו ניתן לראות את טעינת הקבצים ישירות ל IR באופן ידני (לא מתוך הקובץ טקסט) והרמת הסטטוסים המתאימים כך לשמל פקודה D104 ו D205 נטענות בזו אחר זו וסטטוס ld מורם. מתבצעת בדיקה שהמערכת מזהה נכון את הפקודה שנכנסה לבקר.

*הפקודה אינה קוראת את הנתונים מקובץ הטקסט לכן בפקודת ld לא נכנסו ערכים ולא הצגנו את הרגיסטרים והבס כפי שהוצגו במערכת TOP למעלה.

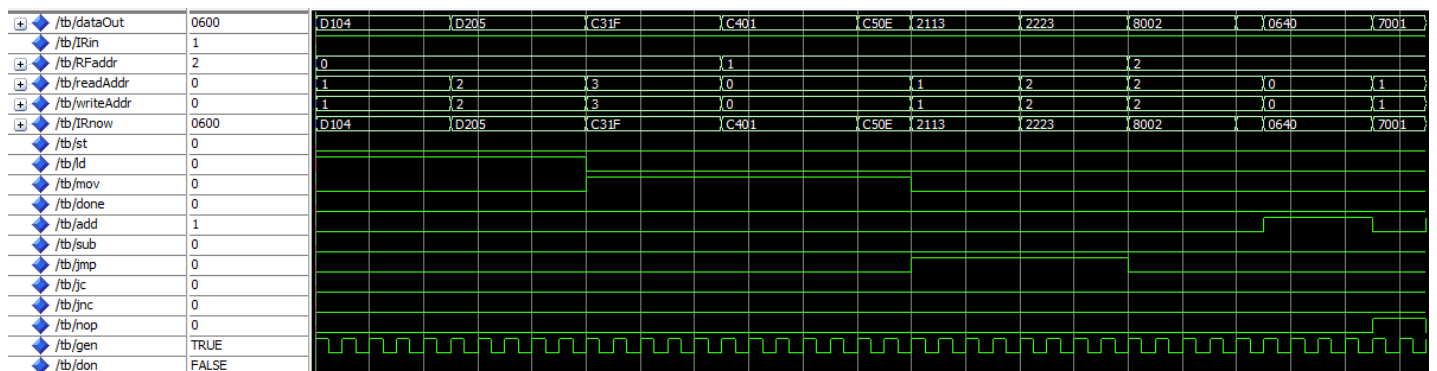
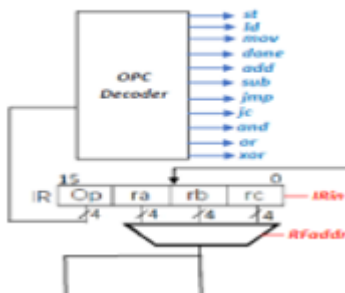


ALU - יחידת החישוב האריתמטי מורכבת גם היא מ-2 רכיבים, האחת האחראית על חיבור חיסור והשנייה על חישובים לוגים שהתבקשו לממש.



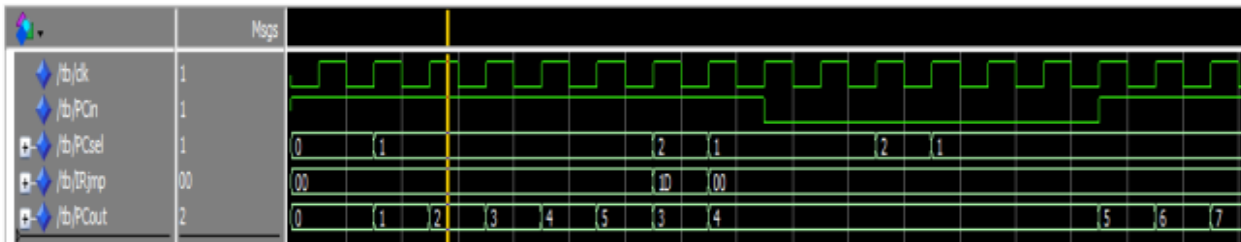
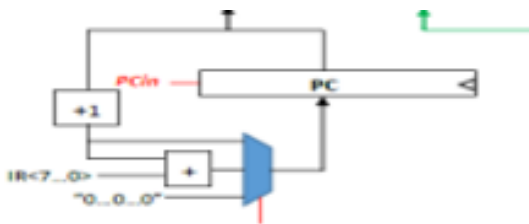
דוגמא לביצוע חיבור חיסור.

Opdecoder – יחידה זו האחראית לפיענוח ה OP CODE ושליחת הסטטוס המתאים מכילה בתוכה את ה IR וה mux לבחירת ה regA regB בהתאמה



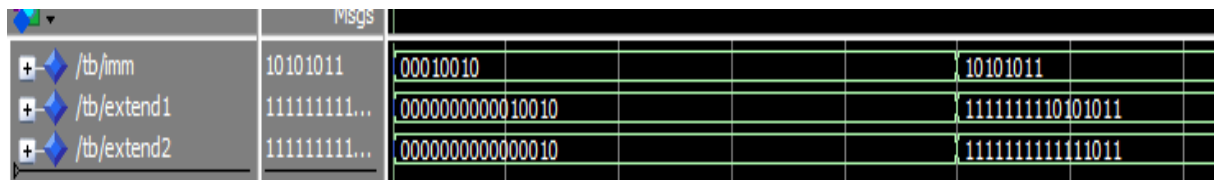
הרצת התכנית הנתונה ושליחת הסטטוסים ל datapath הרכיב שמעליו בהיררכיה.

PC-Unit : היחידה הזו מקבלת את קווי הבקרה pcini pcsel ובהתאם לכך מקדמת את ה PC בערוץ הרצוי



אפשרויות קידום ה PC : השארות במקום , קידום באחד , וקפיצה

Signextesion - יחידה זו מקבלת כ קלט את 8 הביטים ה MSB של IR ומרחיבה אותם ל16 ביט לפי הביט האחרון. כפי שהוגדר למעלה לרכיב זה אין קווי בקרה. הוא תמיד קורא את 8 הביטים של IR ומאפשרת כתיבה לבס בעזרת קווי בקרה imm1 ו imm2.



שאר ה components היו נתונים בהגדרת המשימה ולכן לא בוצע עבורם TB.