ADVANCED CPU ARCHITECTURE AND

HARDWARE ACCELERATORS LABRATORY

Final Project Report

Dvir Zaguri 315602284

Elad Hubashi 313612038

**רקע**

מטרות פרויקט זה היא לממש תצורה של מעבד מסוג MIPS SINGLE CYCLE , רכיבי I/O ויכולת קבלת אינטרפטים ולשלבן יחד, בנוסף להבין את השוני בין CPU כיחידה לבין MCU אשר מתקשר עם יחידות חיצוניות ומבנה הזיכרון של רכיב FPGA.

בפרויקט זה מימשנו ארכיטקטורת MIPS (MIPS ISA) עם יכולת עבודה עם שני זיכרונות CHACHE אחד לפקודות המערכת והשני לזיכרון.

בנוסף מימשנו רכיבי פריפריה כגון, מחלק שלמים, רכיב שירות לרכיבי GPIO ורכיב שעון BTIMER.

את המערכת מימשנו בעזרת שפת VHDL וקמפלנו בעזרת MODELSIM ו QUARTUS לסינתזה וקימפול.

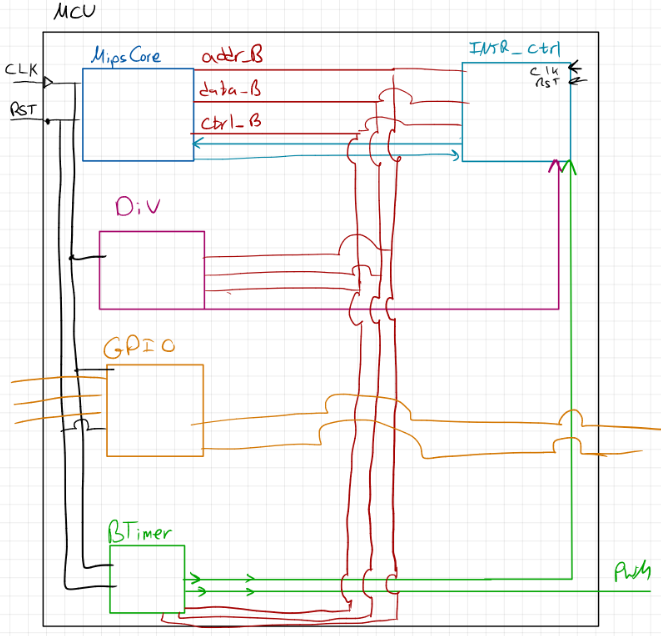
חילקנו את המערכת למספר קומפוננטות ונפרט עליהן בהמשך, כן ניתן תיאור היררכי של היחידות המרכזיות.

**פירוט ה-ENTITIES הקיימים במערכת**

* **TOP (MCU.vhd) –** קובץ העוטף את כל המערכת, מקבל את הכניסות ומוציא את היציאות הדרושות, מחבר בין כל הרכיבים הרשומים מטה.
* **MIPSenv (MIPSenv.vhd) –** קובץ הליבה של הCPU בארכיטקטורה הרצויה, יודע לתקשר עם הזיכרונות השונים ולבצע את ההוראות המתקבלות ולהפעיל את שאר המערכת כולל שליחת קווי בקרה.
* **DIV\_env (DIVenv.vhd) –** מאיץ חומרה שמבצע חלוקת שלמים לא מסומנים.
* **INTERRUPT\_env (INTRenv.vhd) –** רכיב היודע לתכלל את סוגי הפסיקות ולהעבירן לCPU ולקבל ממנו קווי בקרה המאפשרים פסיקות אליו.
* **GPIO (GPIO.vhd) –** רכיב התומך ביציאות וכניסות חיצוניות למיקרו מעבד.
* **BTimer\_env (BTimerEnv.vhd) –** רכיב הממש טיימר פנימי של המיקרו מעבד, הרכיב יודע להוציא אות ריבוע מסוג PWM.
* **רכיבי מימוש פנימיים**

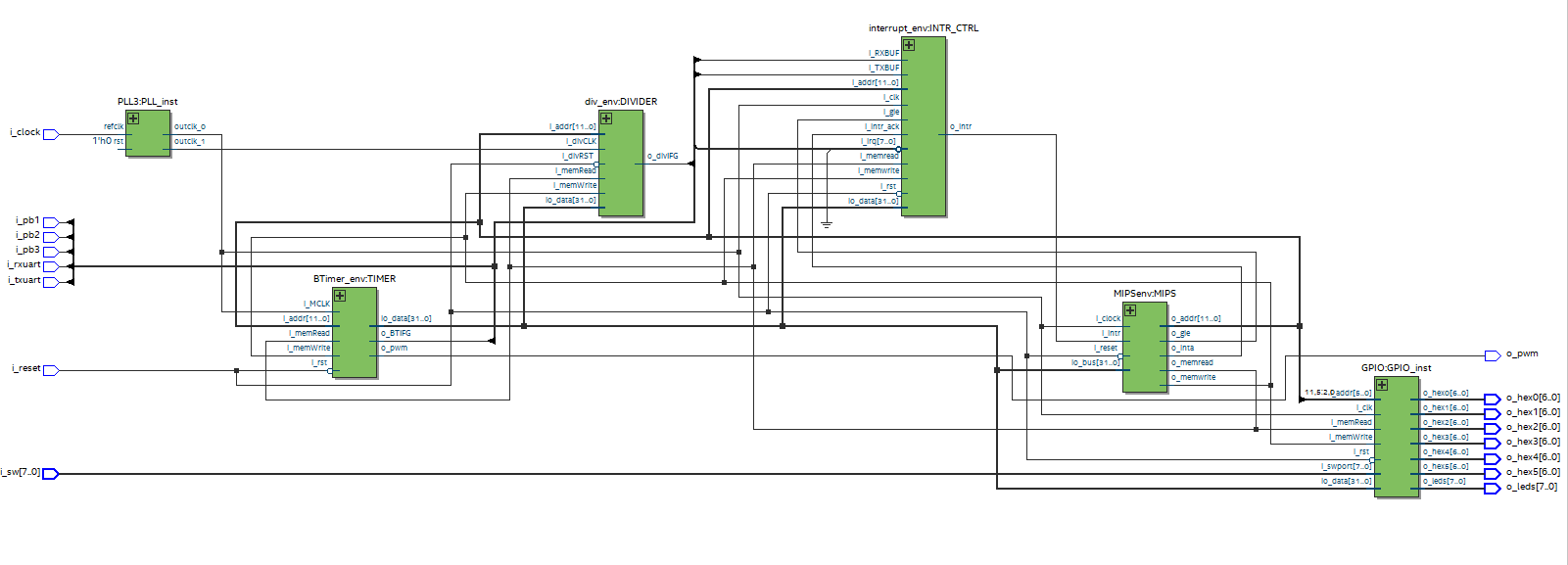
**על כל הקומפוננטות קיים הסבר מפורט בקובץ readme.txt**

תיאור סכמטי שצויר על ידי רב אמן בדיוק מירבי:



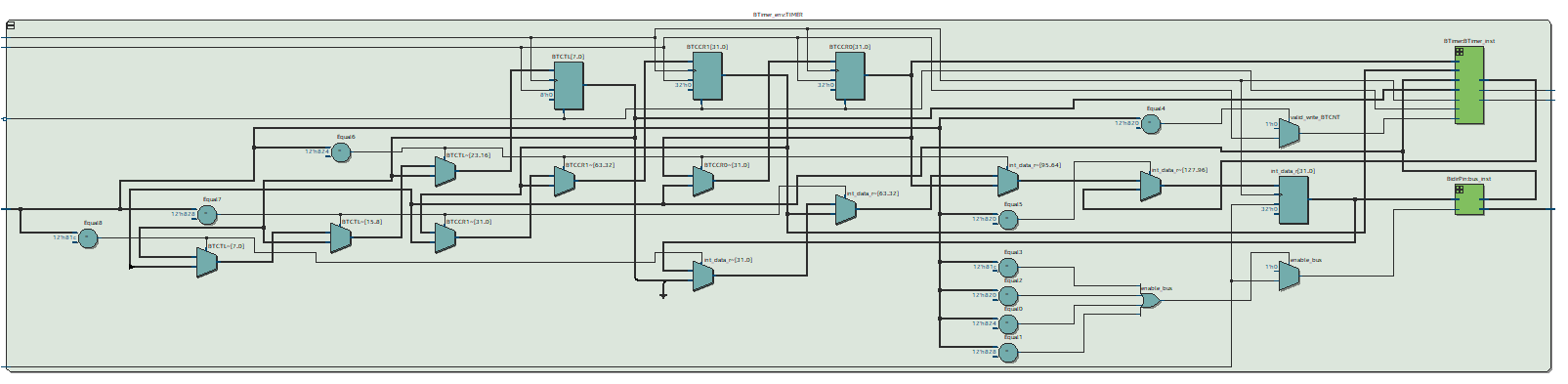
איור 1 איור סכמטתי של המערכת בתצוגת TOP

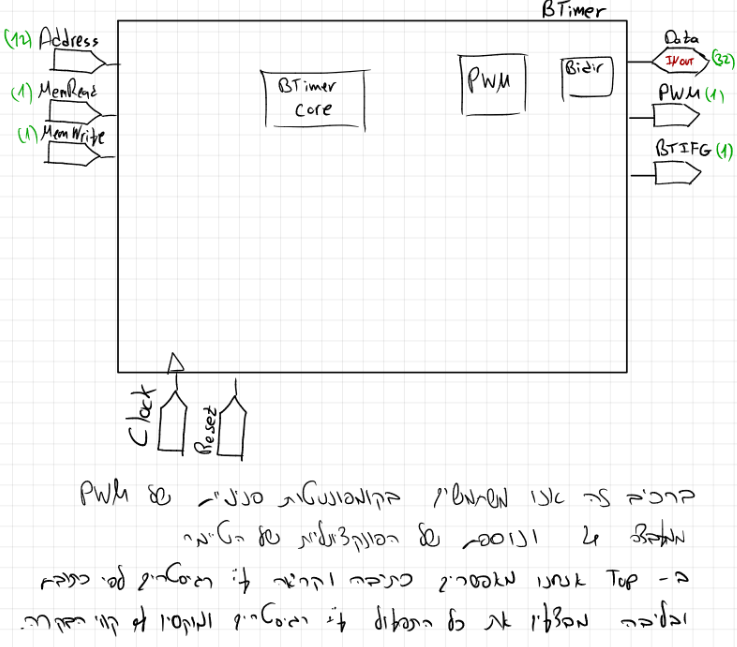
כעת נעבור לתיאור ב QUARTUS של כל רכיב:

****

איור 2 : RTL עבור TOP, נקרא MCU

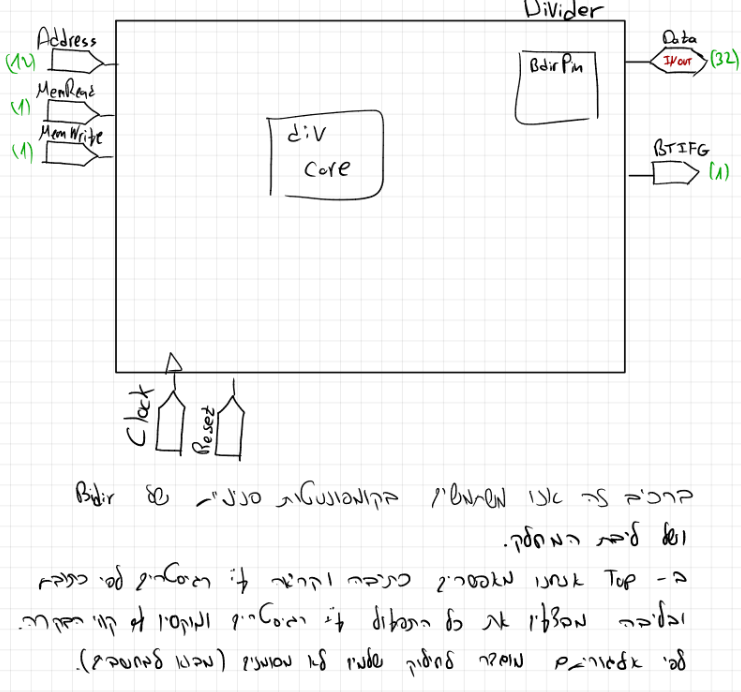
רכיבים נוספים:



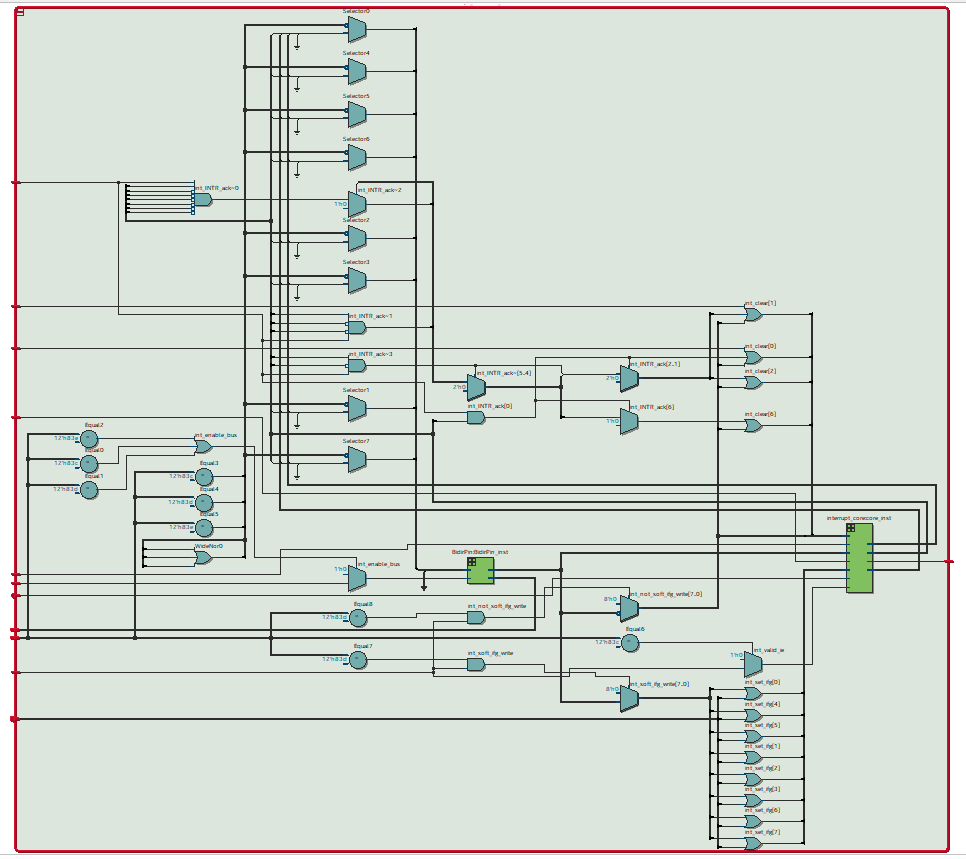
איור 3 רכיב הטיימר BTIMER

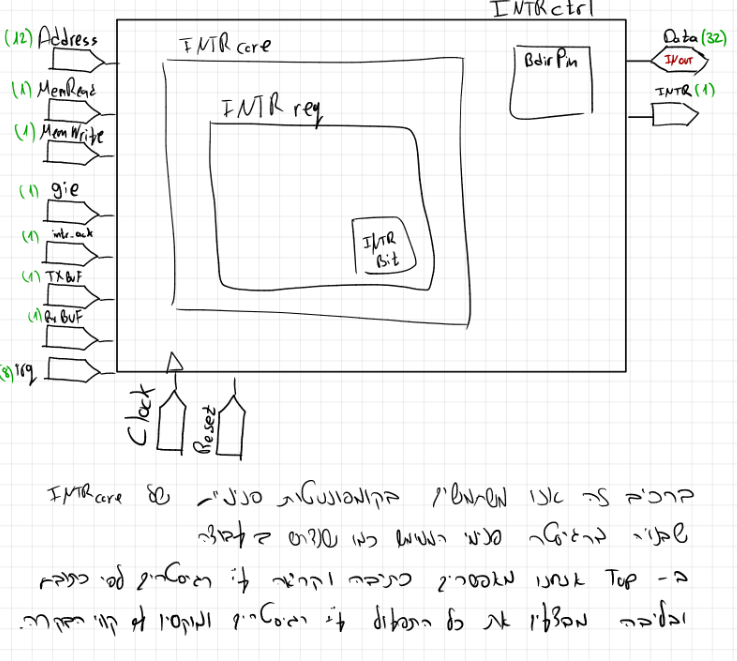
תמונה שמכילה טקסט, מקביל, תרשים, מלבן

התיאור נוצר באופן אוטומטי



איור 4 : רכיב מחלק DIVIDER



איור 5 רכיב מטפל פסיקות INTR CTRL

תמונה שמכילה טקסט, תרשים, כתב יד, קו

התיאור נוצר באופן אוטומטי

תמונה שמכילה תרשים, תוכנית, קו, שרטוט טכני

התיאור נוצר באופן אוטומטי

איור 6 ליבת המעבד MIPS

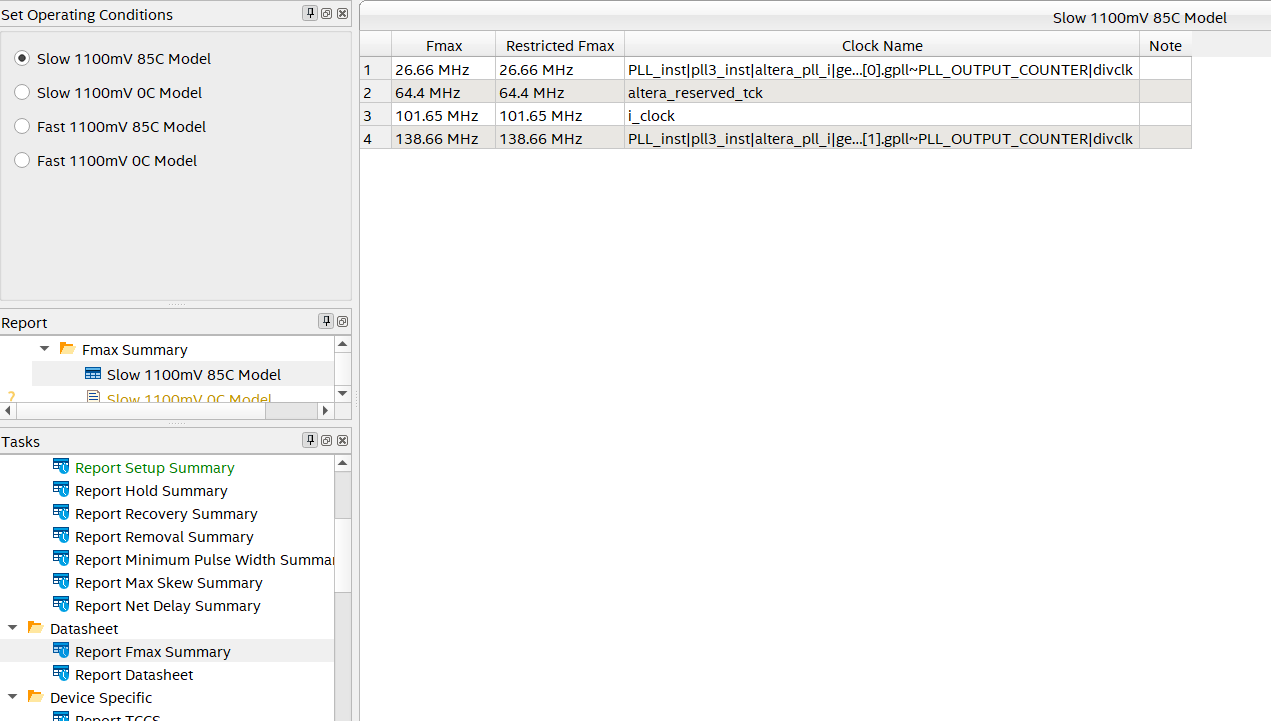
תמונה שמכילה טקסט, תרשים, כתב יד, צילום מסך

התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט, תרשים, מקביל, צילום מסך

התיאור נוצר באופן אוטומטי

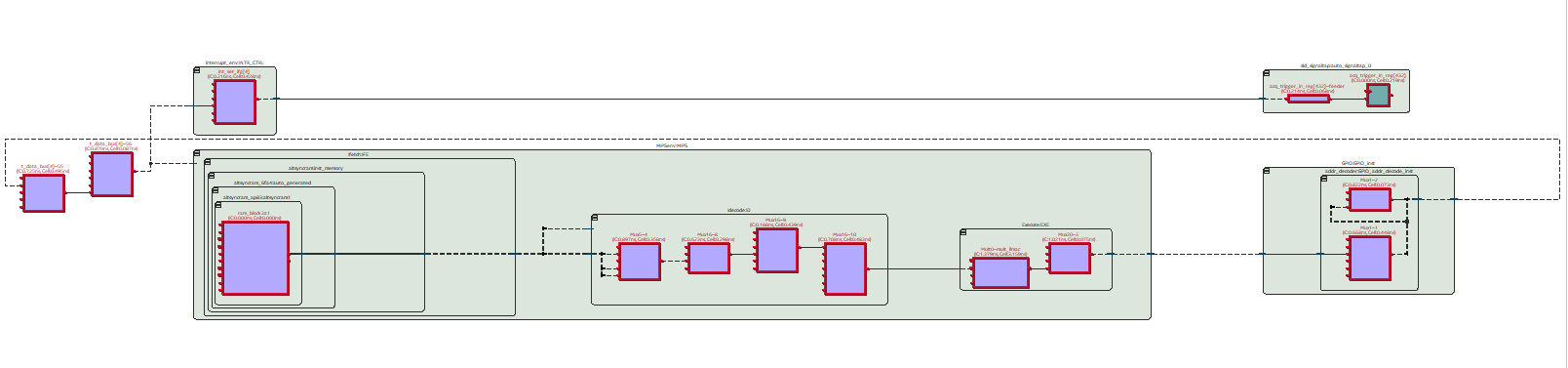
איור 7 רכיב מטפל ברכיבי GPIO

כעת נעבור לתיאור זמני המערכת,   
תיאור עבור תדר מקסימלי



איור 8 תדר מקסימלי עבור שעוני המערכת

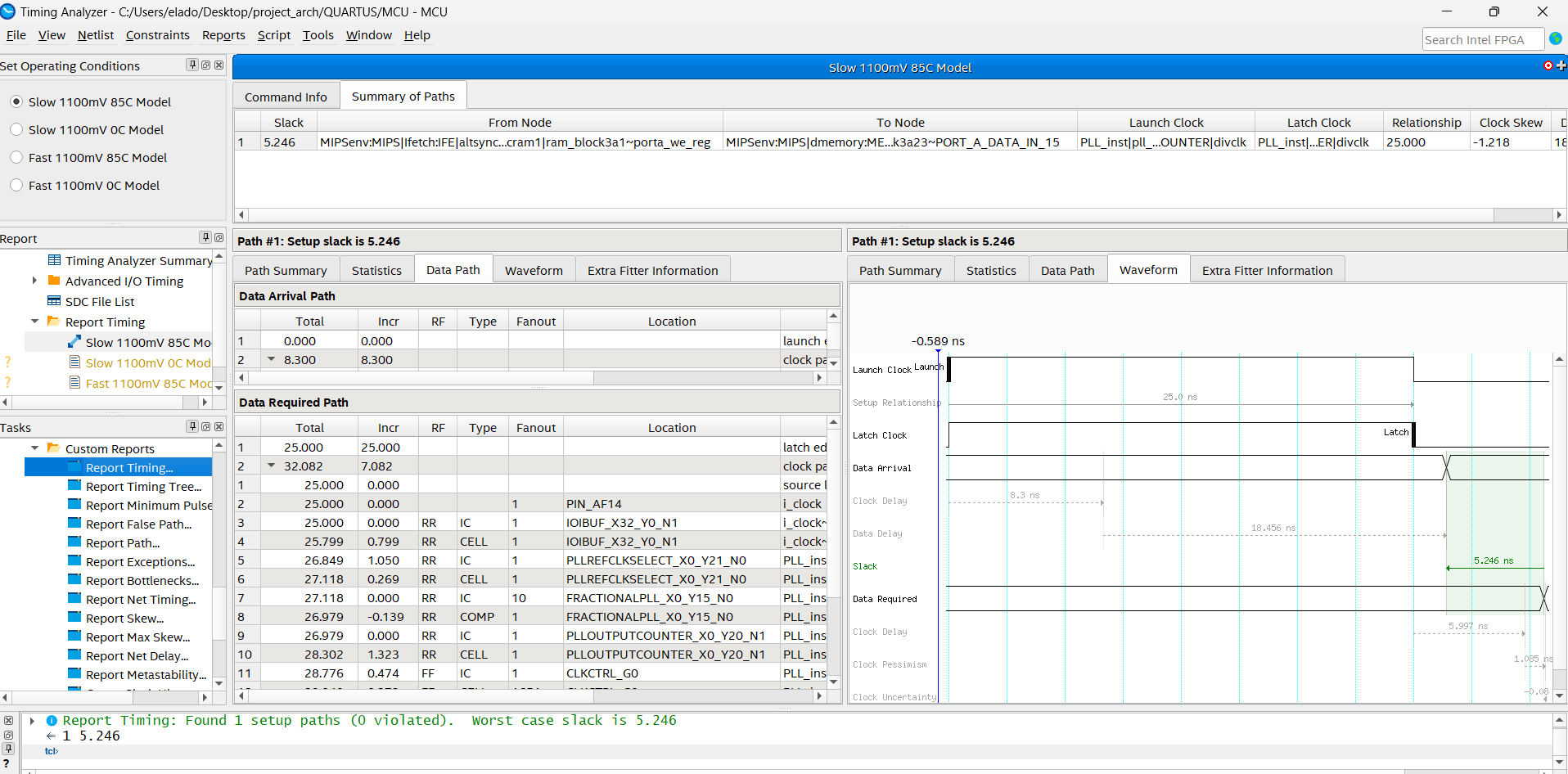
נתיב קריטי במערכת,



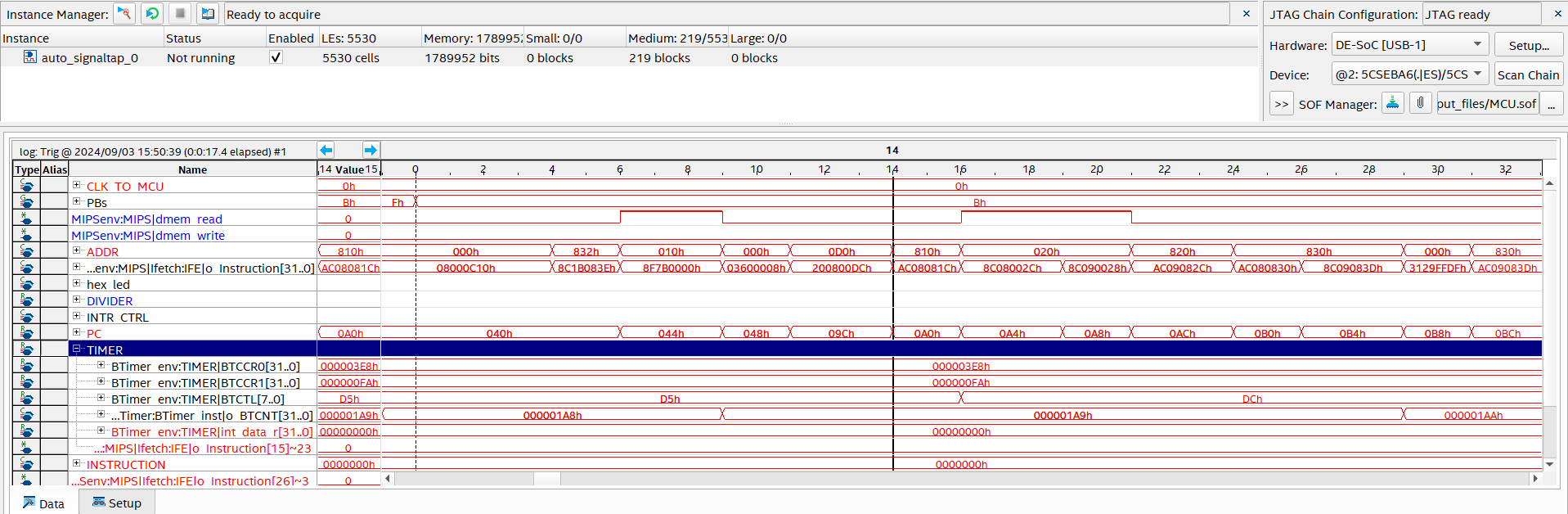
איור 9 נתיב קריטי

דיון לגבי נתיב קריטי:

ניתן לראות כי הנתיב הקריטי נמצא בחלקו של ליבת המעבד, ובפרט במעבר דרך הזיכרון. זהו משהו שיכלנו לצפות מכיוון שאנחנו יודעים כי כתיבה לזיכרון היא דבר "יקר" שלוקח זמן בגלל המרחק שלו מהמעבד עצמו, באופן עקרוני אין כל כך דרך לשפר את קצבי הכתיבה / קריאה מהזיכרון אך כן ניתן לשפר באופן מסויים את הנתיב אחריהן במידה ונעשה אופטימיזציה לביצועי המעבד. אם היה לנו זמן היינו משקיעים את רובו בעבודה על של ה EXE של המעבד ולנסות לבצע בו קיצורים וגם את שלב ה DECODE . תדר השעון עבור הזיכרון הוא בערך 60 מגה הרץ ועבור כל המערכת בערך 20 מגה הרץ כך שככל הנראה ניתן לשפר את זמן המערכת הכולל.



הוכחת פעילות עבור המערכת על ידי single tap ,



שינוי בשעון BTIMER

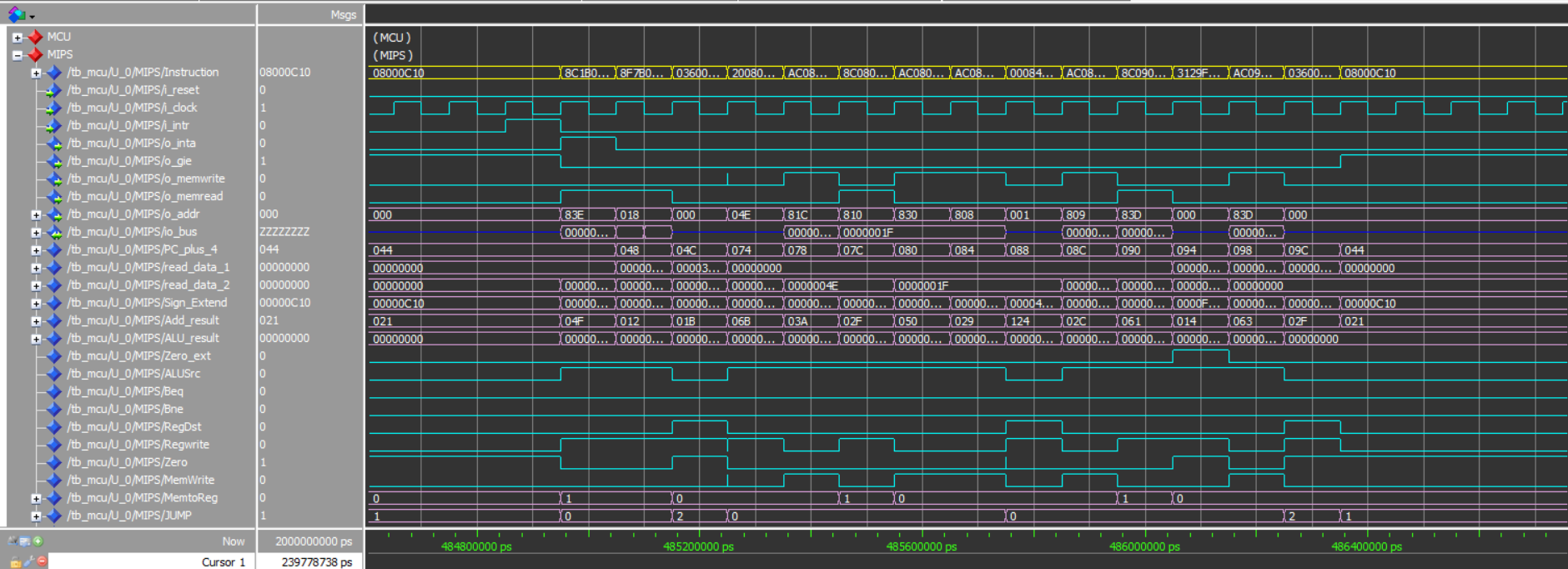
שינוי בהוראות ההפעלה של המערכת לפי פסיקה מלחצן 3 בטסט מספר 4

שינויים בכתובות בניהול ה MIPS

לחיצה על PB3  
והפעלת טריגר

שינוי בכתיבה לרגיסטר בקרה של הטיימר בכתובת 81C

תיאור גלים באפליקציית מודל סים,

עבור MIPS

עבור TIMER

תמונה שמכילה תכונות מולטימדיה, תוכנה, תוכנה גרפית, טקסט

התיאור נוצר באופן אוטומטי

מסקנות,  
תחילה נאמר כי היה מאוד מעניין ומלמד לעבוד על הפרוייקט ההנדסי, למדנו איך לקרוא SPEC ולהמיר אותו לרכיבים, למדנו איתך לקשר בין רכיבים ולסנכרנן בניהם, למדנו איך לעבוד עם תוכנות חשובות בשוק ואיך להפיק מהן מידע מהימן לטובת ורפיקציה, למדנו כיצד ניתן למקסם את היכולות והוכחנו את זה באופן פרקטי. למדנו על ISA ועל ארכיטקטורות שונות, בנוסף למדנו על איך ניתן להאיץ חומרה ולתקשר עם מעבד כולל פרוטוקול פסיקות עם עדיפות.