**הטכניון - מכון טכנולוגי לישראל**

**הפקולטה להנדסת חשמל**



מעבדה 1

פרויקט סיום

תבנית דוח מסכם

גרסה 1.3

קיץ 2018

מחברים: אברהם קפלן, דודי בר-און

|  |  |  |  |
| --- | --- | --- | --- |
| מועד | ביצוע עד סעיף | שם המדריך בפועל | תאריך |
| ביצוע הניסוי |  |  |  |
| השלמת חלקים חסרים -1 |  |  |  |
| השלמת חלקים חסרים -2 |  |  |  |

|  |  |  |
| --- | --- | --- |
| סטודנט | שם פרטי | שם משפחה |
| 1 | ברק | זן |
| 2 | בועז | טייטלר |

|  |  |
| --- | --- |
| שם הפרויקט | Piano Hero |

תוכן עניינים – פרויקט

Contents

[1 נספח מנהלתי 3](#_Toc522550687)

[2 הקדמה 3](#_Toc522550688)

[2.1 צילום של הפרויקט 3](#_Toc522550689)

[2.2 הנחיות כלליות 3](#_Toc522550690)

[3 אפיון הפרויקט 3](#_Toc522550691)

[3.1 הדרישות המקוריות מהפרויקט -(כמו במצגת) 3](#_Toc522550692)

[3.2 החלק היצירתי 3](#_Toc522550693)

[4 ארכיטקטורה 4](#_Toc522550694)

[4.1 תפקיד היחידות: 4](#_Toc522550695)

[5 סכמת מלבנים פנימית 4](#_Toc522550696)

[רשימת מכלולים (מלבנים) עיקריים, תפקידם וסדר ביצועם 4](#_Toc522550697)

[5.1 פרוט ארבעת המודולים העיקריים 6](#_Toc522550698)

[5.1.1 [שם המודול] 6](#_Toc522550699)

[5.2 בחירת המודולים למצגת סופית 6](#_Toc522550700)

[יש להגיש חלק זה למעבדת פרויקט VGA 6](#_Toc522550701)

[6 שלבים במימוש הפרויקט 7](#_Toc522550702)

[6.1 סיפתח 7](#_Toc522550703)

[6.2 פתיחת PIPE 7](#_Toc522550704)

[7 תיאור מפורט של שני מודולים -(כמו במצגת) 7](#_Toc522550705)

[7.1 [שם המודול] - [שם הסטודנט האחראי] 7](#_Toc522550706)

[7.1.1 דיאגרמת מלבנים (תהליכים) 7](#_Toc522550707)

[7.1.2 דיאגרמת מצבים bubble diagram 7](#_Toc522550708)

[7.1.3 פרט את המצבים העיקריים - 8](#_Toc522550709)

[7.1.4 מסך(י) סימולציה 8](#_Toc522550710)

[7.2 [שם המודול] - [שם הסטודנט האחראי] 9](#_Toc522550711)

[7.2.1 דיאגרמת תהליכים 9](#_Toc522550712)

[7.2.2 דיאגרמת מצבים 9](#_Toc522550713)

[7.2.3 מסך(י) סימולציה 9](#_Toc522550714)

[יש להגיש חלק זה למעבדת פרויקט 2 9](#_Toc522550715)

[8 Signal Tap (S.T.) 10](#_Toc522550716)

[9 מימוש ההירארכיה עליונה 10](#_Toc522550717)

[9.1 שרטוט 10](#_Toc522550718)

[9.2 צריכת משאבים 11](#_Toc522550719)

[10 סיכום ומסקנות 11](#_Toc522550720)

[11 המלצות לשנה הבאה 11](#_Toc522550721)

[12 נספחים: דפי נתונים, דפי מידע שונים בהם השתמשת. 11](#_Toc522550722)

# נספח מנהלתי

|  |  |  |  |
| --- | --- | --- | --- |
| תיאור | תאריך | שם המדריך | הערות ומסקנות |
| דיון בהגדרת הפרויקט |  |  |  |
| סכמת מלבנים סיפתח |  |  |  |
| סכמת מלבנים PIPE |  |  |  |
| מכונת מצבים של כל הפרויקט |  |  |  |
| הגדרת שני המכלולים העיקריים |  |  |  |
| CODE REVIEW |  |  |  |
| דיונים על בעיות |  |  |  |

# הקדמה

## צילום של הפרויקט

|  |
| --- |
|  |

## הנחיות כלליות

* מטרת הדוח לתעד בצורה מלאה את פרויקט הסיום שבצעתם.
* יש לכתוב בצורה מלאה וברורה, כך שנתן יהיה בעתיד על סמך קריאת הדוח, להבין את הפרויקט.
* יש לוודא שכל השרטוטים, הסכמות, האיורים, הגרפים, התמונות וכו' ברורים ומובנים. שרטוט מ QUARTUS ע"י: סימון השרטוט, העתק, הדבק, ולא Print-Screen.
* בכל אחד מפרקי הדוח, יש לציין את החלק השייך לתוספת היצירתית.

# אפיון הפרויקט

## הדרישות המקוריות מהפרויקט -(כמו במצגת)

|  |
| --- |
| דרישות בסיסיות   * סולם 12 תווים של פסנתר המוצג על המסך (סולם יחיד) * שיר אחד\*, מוכן מראש שמתנגן ברקע * לוח נגינה דו-ממדי ציקלי, שנע מחלקו העליון של המסך לחלקו התחתון באופן סינכרוני לשיר * על הלוח מופיעים ה"תווים" אותם צריך לנגן * בעת נגינה התו המתאים על גבי הפסנתר נצבע בצבע * בכל פעם שמנגנים את התו בזמן הנכון מקבלים נקודה. כשמפספסים מופיע צליל פספוס * ניתן לשנות את מהירות הניגון   דרישות מתקדמות   * בחירה מבין מספר שירים * הקשות הדורשות לחיצה ארוכה |

במידה וחסרו פרטים בהגדרת בפרויקט, הוסף את ההנחות שלך לפיהם פעלת.

|  |
| --- |
| * תווים אשר ינוגנו בעתיד (פרק זמן קצר קדימה) יופיע על המסך כמלבנים שירדו למטה * כשלוחצים על אחד מקלידי הפסנתר, יתנגן התו של הקליד הקלוונטי |

## החלק היצירתי

הדרישות הנוספות מהפרויקט כתוצאה מהחלק היצירתי שהוספת.

|  |
| --- |
| * מלבני הקלידים מעוצבים * להוסיף תמיכה לכמה סולמות במקביל |

# ארכיטקטורה

היחידות מהן בנוי הפרויקט (כרטיסים, אמצעי קלט/פלט וכו') וזרימת הנתונים דרכן.

שרטוט המבנה והסבר תפקידה של כל יחידה. – *העזר ברכיבים מהמצגת ואל תגיש שרטוט בעפרון*

|  |
| --- |
|  |

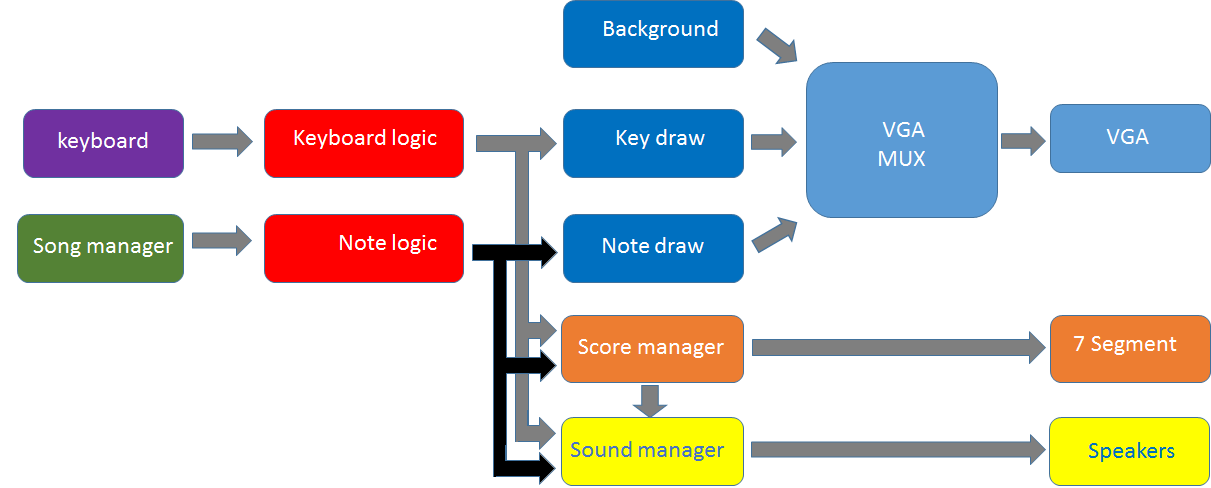
## תפקיד היחידות:

|  |  |
| --- | --- |
| שם | תקציר פעולתה |
| כרטיס DE10 | מחשב את לוגיקת המשחק. אחראי על הרצת מכונת המצבים, אכסון זיכרון עבור השירים, חישוב התגובה במקרה של קבלת מקש מהמקלדת, התקדמות במשחק, שליחת התמונה המתאימה למסך ושליחת הצלילים לרמקול.  בנוסף, על הכרטיס ניתן לראות את הניקוד והזמן על צג ה-seg7. |
| מסך (אמצעי פלט) | מציג את תמונת המשחק על המסך. מקבל את התמונה הרלוונטית מכרטיס ה-DE10 בעזרת חיבור VGA. |
| רמקול (אמצעי פלט) | משמיע את צלילי המשחק. צלילי המשחק מוגדרים בכרטיס ה-DE10 ומתקבלים ברמקול בעזרת חיבור aux. |
| מקלדת (אמצעי קלט) | אמצעי אשר בעזרתו המשתמש בוחר על איזה "תו" ברצונו לנגן. תווי המשחק מיוצגים ע"י מקשי המקלדת אשר נשלחים לכרטיס ה-DE10 בעזרת חיבור ps/2 |

# סכמת מלבנים פנימית

חלוקת הפרויקט למודולים פונקציונליים והקשרים ביניהם.

שרטוט ***סכמת המלבנים הכללית (VISIO או ( PPT***



## רשימת מכלולים (מלבנים) עיקריים, תפקידם וסדר ביצועם

פרט בטבלה את כל המכלולים העיקריים.

רצוי להתחיל עם ליבת הפרויקט (החלק הקשה/הארוך/המסובך של הפרויקט)

* בתפקיד מנוון רשום מה תעשה לפתיחת ה-PIPE
* לכל יחידה פרט את הסיבוכיות שתידרש לדעתך למימושה (קל בינוני כבד) \
* החלט מהו סדר המימוש שבחרת

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| מודול מס | שם | תפקיד | תפקיד מנוון PIPE | סיבוכיות התכן | סדר ביצוע |
| 1 | keyboard | אחראי לקבל לחיצות מקש מהשתמש | זיהוי תו ספציפי | קל | 1 |
| 2 | Song manager | אחראי על קצב השיר ובחירת התווים | שיר קצר עם כל צליל פעם אחת | בינוני | 11 |
| 3 | Keyboard logic | ממיר בין מקש מהמקלדת לבין תו על הפסנתר | מיפוי בין מקש ספציפי לתו ספציפי | קל | 2 |
| 4 | Note logic | אחראי על לוגית התווים, קידום תווים בזמן והתראה של מתי לנגן תו מסויים | ניהול ציור של תו בודד | בינוני | 10 |
| 5 | Background | אחראי לרקע של המשחק | ציור רקע אפור ואוקטבה אחת ללא ציור לחיצה | קל | 7 |
| 6 | Key draw | מצייר את הקליד שמייצג את לחיצת מקש המקלדת | צובע בצבע מתאים תו לחוץ | קל | 8 |
| 7 | Note draw | מצייר את התווים יורדים | ציור של תו אחד יורד | קשה | 9 |
| 8 | Score manager | אחראי לבדוק האם נתקבל לחיצה מהמקלדת על התו המתאים לשיר ולעדכן את הניקוד. אחראי גם להוספת אפקטים כמו ADSR וכו. | הוספת נקודה כאשר יש פגיעה | בינוני | 12 |
| 9 | Sound manager | אחראי לנגן את התווים | ניגון של תו בודד ללא ADSR | קשה | 4 |
| 10 | VGA mux | מנהל אובייקטים לציור: תווים, מקשי המקלדת ורקע | ניהול רק של רקע ולחיצות | קל | 6 |
| 11 | VGA | אחראי על תפעול המסך | קיבלנו במעבדה | קל | 5 |
| 12 | 7 Segment | אחראי על תפעול צג הספרות על כרטיס ה-DE10 | קיבלנו במעבדה | קל | 13 |
| 13 | speaker | אחראי על תפעול השמע | קיבלנו רכיב מוכן מהמעבדה | קל | 3 |

## פרוט ארבעת המודולים העיקריים

### Sound manager

|  |  |
| --- | --- |
| תפקיד מפורט | מקבל מערך בינאי של תווים ומנגן בצורה נעימה לאוזן את התווים הלחוצים |
| מימוש מצומצם PIPE)) | מקבל תו ומשמיע צליל בתדר המתאים ללא ADSR |
| אופן המימוש | מיצרים סינוס לכל תו בתדר המתאים כמו שעשינו במעבדת MSS, מכניסים לADSR ע"מ לקבל צליל נעים לאוזן, מחברים את כל התווים ומשמיעים |
| כניסות עיקריות | מערך של תווים לחוצים |
| יציאות עיקריות | יציאה 16 ביט של סאונד |

### Song manager

|  |  |
| --- | --- |
| תפקיד מפורט | אספקת מערכים של תווים שצריך לנגן\לצייר כרגע, לפי מנגינה שנקבעה מראש |
| מימוש מצומצם PIPE)) | ניגון שיר קצת עם כל תו פעם אחת |
| אופן המימוש | השירים נשמרים במערך בו כל תא בגודל 24 ביטים, כאשר 1 אומר שצריך לנגן את התו המתאים לתא זה במשך עשירית שנייה |
| כניסות עיקריות | שעון, ריסט, מקשים לבחירת שירים |
| יציאות עיקריות | שני מערכים, אחד לתווים שיש לנגן כרגע ואחד לתווים שיש לצייר כרגע |

### note manager

|  |  |
| --- | --- |
| תפקיד מפורט | ציור התווים שיש לנגן והורדתם בהדרגתיות על פני המסך |
| מימוש מצומצם PIPE)) | ציור של תו בודד שיורד לפי הקצב |
| אופן המימוש | לכל תו השתמשנו בשיפט רגיסטר שמקבל מתי צריך להתחיל לצייר תו זה והוא "דואג" להוריד את התווים לפי הקצב (להוריד = שיפט ימינה) |
| כניסות עיקריות | מערך של תווים לצייר כרגע, X, Y |
| יציאות עיקריות | Draw request מתאים |

### KBD logic

|  |  |
| --- | --- |
| תפקיד מפורט | קבלת KEY, MAKE ו-BREAK והוצאת מערך של תווים לחוצים ובנוסף אותות נפרדים לכל לחצן שאנו משתמשים בו |
| מימוש מצומצם PIPE)) | הוצאת מערך של 7 תווים |
| אופן המימוש | כאשר מקבלים תו וMAKE הביט המתאים הופך ל-1, כאשר מקבלים תו וBREAK הביט המתאים יורד ל-0 |
| כניסות עיקריות | keycode, make, break |
| יציאות עיקריות | מערך של 24 תווים, ויציאה של ביט לכל לחצן שהשתמשנו בו |

|  |  |
| --- | --- |
| מודול | Sound manager |
| סטודנט | ברק |
| למה הוא חשוב | מייצר צליל שנשמע כמו פסנתר |
| מה נציג | את הADSR |

## בחירת המודולים למצגת סופית

|  |  |
| --- | --- |
| מודול | Song manager |
| סטודנט | בועז |
| למה הוא חשוב | אחראי על המנגינה, הקצב והשירים השונים |
| מה נציג | את הsong player |

# שלבים במימוש הפרויקט

בגלל המורכבות של הפרויקט יחסית למה שתכננתם עד היום, וכדי שהפיתוח יעשה בצורה חלקה, ביצוע הפרויקט נעשה בשלושה שלבים, מהקל לכבד.

1. סיפתח – ביצוע פריט אחד או שניים הקשורים לממשקים של הפרויקט: תצוגה על מסך VGA וצליל.
2. PIPE – ביצוע מסלול שלם ומנוון של הפרויקט הדורש שיתוף מכלולים עיקריים שלו.
3. הפרויקט השלם.

חובה לבצע את כל השלבים בסדר שלמעלה וכל שלב יש לו חלק בציון על הפרויקט.

כל שלב הוא חלק מדוח הכנה בהתאם ללו"ז המופיע במודל.

## סיפתח

בגלל מורכבות הפרוייקט והקשר היחסית מועט בין החלק של הסאונד לחלק של הציור והניקוד – בתחילת הפרויקט הTOP שלנו היה רק הsound manager וזה השרטוט המקורי שעל פיו מימשנו אותו

|  |
| --- |
|  |

## פתיחת PIPE

תאר מה יעשה ה PIPE,

|  |
| --- |
| הפייפ הראשוני הכיל רק את הSOUND FLOW, הוא קיבל תווים מהמקלדת, הפך את הלחיצות השונות לתווים שיש לנגן, יצר את הצליל הבסיסי לכל תו, הפך את הצילילם השונים לנעימים לאוזן ובסוף הכל התחבר ויצא ליציאת הסאונד. |

# תיאור מפורט של שני מודולים -(כמו במצגת)

שימו לב שיש להקפיד לשים מודול אחד לכל סטודנט- (שיהיה תכנון שלו ועליו הוא יסביר)

**יש לקחת מודולים מסובכים**, רצוי כאלה המכילים המכילים מכונת מצבים , ולא קוד טרוויאלי

לכל מודול יש לבצע את הסעיפים שלהלן.

## Sound manager - ברק

### דיאגרמת מלבנים (תהליכים)

תאר את המודול כתהליך אחד או יותר.

|  |
| --- |
|  |

### דיאגרמת מצבים bubble diagram

לתהליכים אותם מימשת בעזרת מכונת מצבים, צייר את דיאגרמת המצבים

|  |
| --- |
| ADSR |

### פרט את המצבים העיקריים -

|  |  |  |
| --- | --- | --- |
| **שם המצב** | **פעילות עיקרית** | **לאיזה מצב עוברים מהמצב הנוכחי ובאילו תנאים** |
| Idle | Amplifier מאופס ומחכים ללחיצה של התו | **עוברים** ל**- attack** **עם** קבלת לחיצה |
| attack | הגברה הדרגתית אך מהירה של ההגבר | **עוברם ל** decay כאשר הamplifier יורד מ 32000 |
| decay | ירידה הדרגתית אך מעט מהירה של ההגבר עד חצי העוצמה | **עוברים ל** releaseReleased כאשר הלחצן משוחרר  **עוברים ל** sustain כאשר הamplifier יורד מתחת ל16000 |
| sustain | ירידה איטית של ההגבר עד רבע העוצמה | **עוברים ל** releaseReleased כאשר הלחצן משוחרר  **עוברים ל** releasePressed כאשר הamplifier יורד מתחת ל8000 |
| releasePressed | ירידה איטית של ההגבר עד רבע העוצמה | **עוברים ל** releaseReleased כאשר הלחצן משוחרר  **עוברים ל** idle כאשר הamplifier יורד ל0 |
| releaseReleased | ירידה איטית של ההגבר עד רבע העוצמה | **עוברים ל** idle כאשר הamplifier יורד ל0 |

### מסך(י) סימולציה

## ניתן לראות בסימולציה הדגמה של לחיצה ארוכה ומעבר בין כל המצבים עד לאיפוס היציאה ואז לחיצה קצרה שהיציאה דועכת.song manager - בועז

### דיאגרמת מלבנים (תהליכים)

תאר את המודול כתהליך אחד או יותר.

|  |
| --- |
|  |

### דיאגרמת מצבים bubble diagram

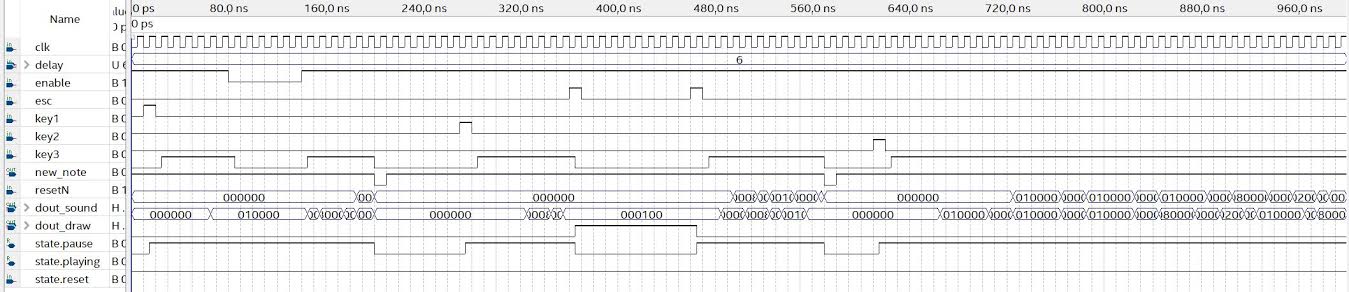
לתהליכים אותם מימשת בעזרת מכונת מצבים, צייר את דיאגרמת המצבים

|  |
| --- |
| Song player |

### פרט את המצבים העיקריים -

|  |  |  |
| --- | --- | --- |
| **שם המצב** | **פעילות עיקרית** | **לאיזה מצב עוברים מהמצב הנוכחי ובאילו תנאים** |
| play | מנגן שיר על פי לחצן שנבחר, כל פרק זמן קבוע מייצר מערך בן 24 ביטים שמכיל את כל התווים המנוגנים באותו רגע. | עוברים -Reset אם נלחץ קבלנו ResetN = ‘0’.  עוברים ל-Pause עם עליית שעון בתנאי שנלחץ esc |
| Pause | עוצר זמנית את ניגון השיר. | עוברים -Reset אם נלחץ קבלנו ResetN = ‘0’.  עוברים ל-play עם עליית שעון בתנאי שנלחץ esc |
| Reset | עוצר את ניגון השיר ומבצע ריסט לכל לבחירת השיר | עוברים ל-play עם עליית שעון בתנאי שנלחץ key1 או key2 או key3 |

### מסך(י) סימולציה



# Signal Tap (S.T.)

אם השתמשת ב .T.S לזהות באג בחומרה, צרף מסך של ה .T.S בו זיהית את הבאג. הסבר מה היה הבאג, כיצד זיהית אותו וכיצד תקנת אותו.

אם לא השתמשת ב .T.S לזיהוי באג בחומרה, צרף מסך של ה .T.S בו מתבצעת פעולה סינכרונית והסבר אותה.

|  |
| --- |
| בדגימה זו ניתן לראות את שלושת היציאות לשלושת נורות הניקוד. כאשר האות שמייצג את ספרת האחדות נמצא בערך 9 וצריך לעלות – הוא מתאפס והערך של ספרת העשרות עולה ב1. |

# מימוש ההירארכיה עליונה

## שרטוט

שרטוט מלבנים של ההירארכיה (העליונה של הפרויקט – מצויר מעל תדפיס הקוארטוס – ראה דוגמא



**דוגמה**

|  |
| --- |
| Top כללי של כל הפרויקט  Sound\_manager המודול גדול מדי מכדי שנוכל להכילו בתמונה אחת (מכיל 24 שורות של רכיבים), לכן הצגנו את הליבה שלו באופן מצומצם (4-7 שורות של רכיבים).  בנוסף, המודל ממומש בקוד ולכן נציגו בעזרת netlist view    Song\_manager    Score\_manager    Note\_draw\_manager  גם כאן המודול גדול מדי וממומש בעזרת קוד שמכיל הרבה מודולים קטנים, ולכן שוב נעזר ב-netlist view. נזכור כי מודול זה מכיל 24 תווים ולכן קיימות בו 24 שורות של note\_draw. נציג פה חלק מהן. |

## צריכת משאבים

|  |
| --- |
|  |

האם צריכת המשאבים (CELLS סבירה , לאן לדעתכם הלכו רב המשאבים?

לדעתינו צריכת המשאבים סבירה מאוד, לדעתינו (ולפי מה שראינו כאשר הארכנו את השירים והוספנו עוד תווים) רוב המשאבים הלכו לשמירת המערכים של השירים, יצירת גלי סינוס קבועים בשביל כל התווים שאנחנו מנגנים וגם על השיט רגיסטרים שמורידים את התווים.

# סיכום ומסקנות

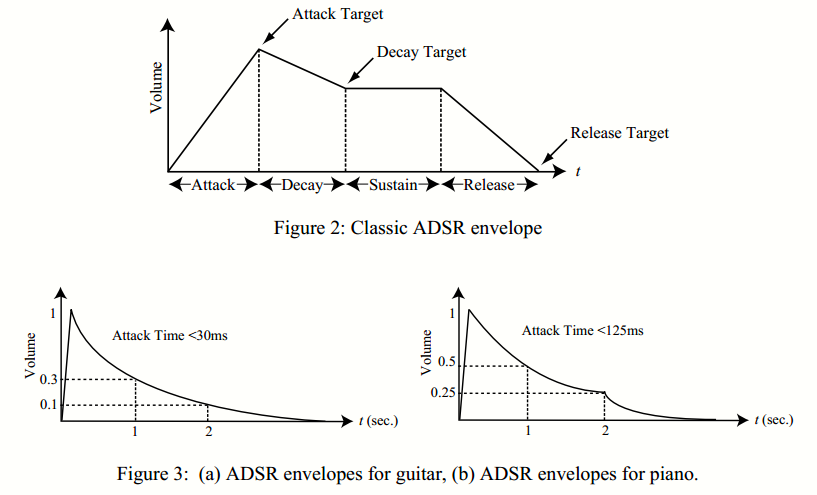
עמידה בדרישות, קשיים , פתרונות אחרים, שימוש בכלים, מסקנות.

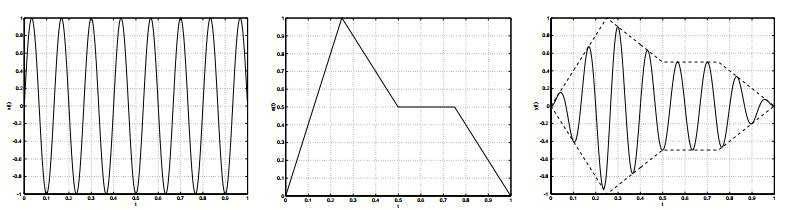
|  |
| --- |
| עמדנו בכל הדרישות שהוגדרו לנו, במהלך הדרך הוספנו עוד יכולות ותכולות למערכת שלא צפינו מראש שיתווספו בחלק היצירתי בהתחלה.  עבודה נקייה עם שמירה על סטנדרט אחיד לאורך הפרויקט (למשל אופן ייצוג מערך התווים) הקל עלינו כרצינו לממש מודולים חדשים ולהוסיף יכולות מבלי לפגוע בתכולות חדשות. קביעת mindset במהלך החשיבה על המערכת הקל לקרוא ולהכנס לקוד אחד של השני.  פיתוח רכיב הADSR היה מלווה בקשיים רבים ברמה ההבנתית, למידת נושא האפקטים לצלילים, יצירת envelop ספציפי עבור פסנתר שישמע טוב היו משימות לא פשוטות שדרשו חקר בנפרד על מנת להפיק צליל שישמע טוב.  הפרויקט פותח ונשמר בgit. כלי זה הקל עלינו מאוד בשיתוף ואיחוד הקוד מהאחד לשני, אך עם זאת, קושי נוסף שהועלה במהלך פיתוח הפרויקט מתבטא בכך שהתוכנה quartus מייצרת קבצים רבים הנלווים לפרויקט, אשר יכולים להשתמנות מקומפילציה אחת לאחרת. המצאות קבצים אלו הקשתה על שמירת סדר בעץ של הgit וגמרה לcommitים גדולים שיוצרים קונפליקטים רבים.  הקושי האחרון שהתגלה במהלך הפיתוח הוא זמני קומפילציה ארוכים. השתדלנו לפתח רק חלקים מינימלים עד לקראת סוף הפרויקט (למשל רק 5 תווים במקום 24, שיר אחד קצר במיוחד במקום 3 שירים). זמני קומפילציה ארוכים הקשו על הבדיקות – היו מספר בדיקות שניתן היה לבצע אך ורק על הבורד (כגון איכות שמע, תצוגת מסך ועוד)  מסקנות:  סדר מקל על הפיתוח – ביצירת תכולות חדשות, שמירה על תקינות תכולות קיימות והבנת הקוד על ידי השותף השני בצוות.  זכרון הינו משאב עדין בפרויקט. שימוש יתר עלול לגרום לזמני קומפילציה ארוכים אשר עשויים להקשות על הבדיקות מערכת. |

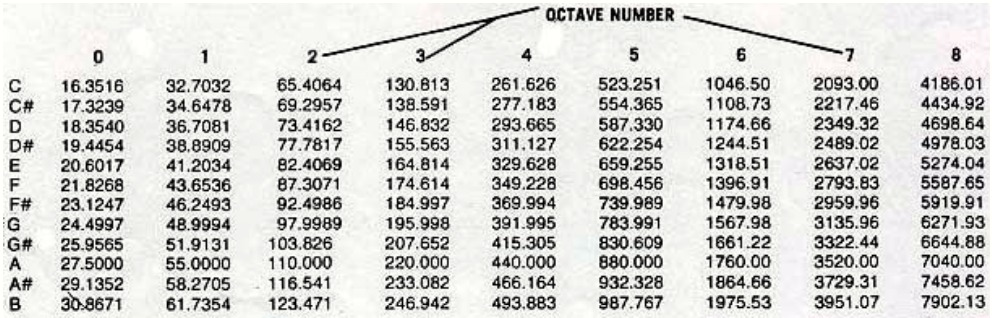
# המלצות לשנה הבאה

|  |
| --- |
| 1. לסדר מראש עבודה עם quartus שתשמר בgit. כנראה שקיימת דרך להסתדר עם כל "קבצי הזבל" שהתוכנה מייצרת. 2. לאפשר לחתום על מקלדות ps2 מהמעבדה. מתברר שזה זן מקלדות נכחד וקשה להשגה |

# נספחים: דפי נתונים, דפי מידע שונים בהם השתמשת.

השתמשנו בהרבה מקורות ומסמכים מהאינטרנט ע"מ ללמוד איך שDSR עובד, נצרף מספר תמונות של חלקים בהם השתמשנו הרבה:  






***לאחר שסיימת - לחץ על ה LINK ומלא בבקשה את השאלון המצורף***

|  |
| --- |
|  |
| |  | | --- | | [**מלא את הטופס**](https://docs.google.com/forms/d/e/1FAIpQLScKIokZgowmcDuf0l79Qzn_sybx6sq9v_V_CBx9J30Exvg08w/viewform?c=0&w=1) | |