המחלקה להנדסת חשמל ואלקטרוניקה

**עבודת גמר - תכן לוגי**

שם הפרויקט:

Programmable Interrupt Controller - PIC

בקר פסיקות דמוי Intel 8259

**מגישים:** ניב כהן 312142524

ולדימיר  סרוב 317569341

ולדיסלב ויזל 320854490

תאריך הגשה: 08.10.2020

תוכן עניינים

הצגת המערכת3

מבוא3

דרישות המערכת / מפרט פונקציונלי3

פעולת המערכת4

אופן הטיפול בפסיקה4

אופן תכנות והתקשרות בין הבקר למעבד4

מפרט טכני6

תרשים מלבנים כללי7

תהליכי תכנון של היחידות במערכת8

Interrupt Request Register - IRR8

Interrupt Mask Register - IMR10

Currently Handled Request Register - CHRR12

Priority Arbiter - PA14

Interrupt Request Priority Examinator15

List Mode Controller18

Writing Counter19

Address Counter21

Interrupt Sender23

Programmable Priority List26

Default Priority List26

Program Controller28

IR Vector to Interrupt Number List Decoder29

CHR Vector to Interrupt Number Decoder31

Interrupt Acknowledge Handler32

Priority Writing Inspector34

Internal BUS Controller40

חישוב תזמוני המערכת45

סימוכין46

הצגת המערכת

מבוא

במחשבים קיימים מספר התקני קלט/פלט המבקשים לתקשר עם המעבד לעיתים בו זמנית, מכיוון שהמעבד אינו יכול לתקשר עם יותר מהתקן אחד בו זמנית קיים הצורך ברכיב שיוכל לבקר על התהליך כך שבכל פעם ההתקן בעל העדיפות הגבוהה יותר מבין כל מבקשי השירות יקבל מענה. לצורך כך קיים בקר פסיקות הניתן לתכנות (PIC – Programmable Interrupt Controller).

ה-PIC מקבל את כל הבקשות לפסיקה מכלל התקני ה-I/O המחוברים, מעבד את הבקשות ומוצא מתוכן את הבקשה בעלת העדיפות הגבוהה ביותר, כאשר המעבד יודיע שהוא פנוי לקבל את הבקשה הטיפול יועבר אליו.

דרישות המערכת / מפרט פונקציונלי

* קבלה של בקשות פסיקה מ8 רכיבי I/O שונים לכל היותר.
* איתור הבקשה בעלת העדיפות המרבית ושליחת בקשת פסיקה למעבד.
* עבודה בשתי שיטות בוררות:
  + על פי רשימת ברירת מחדל הצרובה בזיכרון של הרכיב.
  + על פי רשימה ניתנת לתכנות ע"י המעבד.
* יכולת להתעלם מרכיבי I/O מסוימים על פי בקשת המעבד באמצעות מיסוך הבקשות.
* שליחת מספר הפסיקה המתאים של רכיב ה-I/O בעל העדיפות המרבית למעבד ברגע קבלת אישור לבקשת פסיקה.
* יכולת להעביר למעבד לפי בקשתו את כלל מספרי הפסיקות של רכיבי ה-I/O אשר מבקשים פסיקה באותו רגע נתון.
* זיהוי שגיאות בעת שליחת רשימת עדיפויות על ידי המעבד ופעולה מתאימה בהתאם:
  + כאשר הרשימה הגיעה תקינה תישלח הודעה מתאימה למעבד.
  + כאשר התגלתה שגיאה ברשימה תישלח הודעה מתאימה למעבד והמערכת תחזור לעבוד לפי רשימת ברירת מחדל.
* שמירה של האינדקס המתאים לרכיב ה-I/O המטופל כרגע על ידי המעבד עד סיום התהליך.

פעולת המערכת

אופן הטיפול בפסיקה:

1. מתקבלת לפחות בקשה אחת לפסיקה אל תוך אוגר הפסיקות והביט המתאים נדלק.
2. בקר הפסיקות מעבד את הבקשות שהתקבלו ומתוכן מוצא את הבקשה בעלת העדיפות הגבוהה ביותר ולאחר מכן שולח בקשת פסיקה (INT) למעבד (CPU), במידה והתקבלה בקשת פסיקה חדשה בקר הפסיקות מעבד את הבקשות מחדש, במידה ובקר הפסיקות כבר מצא בקשה בעלת עדיפות גבוהה ביותר והתקבלה בקשה חדשה הוא יוריד את בקשת הפסיקה ויעבד מחדש את הבקשות.
3. כאשר המעבד מוכן לקבל את הפסיקה הוא מחזיר פולס אחד של INT A (Acknowledge) למשך שעון אחד בחזרה לבקר פסיקות.
4. בקר הפסיקות מעביר את הבקשה שנמצאה בעלת עדיפות גבוהה ביותר אל אוגר פסיקה בפעולה (CHRR) ומוריד את בקשת הפסיקה (INT).
5. לאחר שעון אחד בדיוק מאז סיום הפולס הראשון של INT A המעבד שולח פולס נוסף של INT A למשך שעון אחד בלבד.
6. ברגע קבלת הפולס השני של INT A בקר הפסיקות שולח למעבד את מספר הפסיקה המתאים של ה-I/O שקיבל את העדיפות.
7. ברגע שהמעבד קיבל את מספר הפסיקה הוא מתחיל תהליך התקשרות עם רכיב ה-I/O המתאים.
8. לאחר סיום תהליך ההתקשרות בין המעבד לבין הרכיב המעבד מיידע את בקר הפסיקות כי התהליך הסתיים ואוגר הפסיקה בפעולה מתאפס.

אופן תכנות והתקשרות בין הבקר למעבד:

|  |  |  |  |
| --- | --- | --- | --- |
| פעולה | DATA | Write | Read |
| קריאת מידע מאוגר פסיקות (IRR) | 0xCC  1100 1100 | 0 | 1 |
| תכנות אוגר מיסוך הפסיקות (IMR) | 0xBB  1011 1011 | 1 | 0 |
| תכנות בורר עדיפויות (PA) | 0xDA  1101 1010 | 1 | 0 |
| איפוס אוגר הפסיקה בפעולה (CHRR) | 0xDB  1101 1011 | 1 | 0 |

1. קריאת מידע מאוגר הפסיקות:

שליחת המילה המתאימה לשעון אחד בלבד ובשעונים הבאים יתקבלו מספרי הפסיקות של כל הרכיבים המבקשים שירות, התהליך יסתיים בשליחת מספר הפסיקה 000x שהוא כאמור מספר פסיקה לא חוקי.

1. תכנות אוגר מיסוך הפסיקות:

שליחת המילה המתאימה לשעון אחד בלבד ולאחר מכן שליחת המסיכה בשעון הבא וסיום התהליך.

1. תכנות בורר עדיפויות:

שליחת המילה המתאימה למשך שעון אחד בלבד ולאחר מכן המשך לפי המצב הרצוי.

* מצב א: ברירת מחדל

שליחת המילה 0xDE - 1101 1110 לשעון אחד בלבד ולאחר מכן סיום התהליך.

* מצב ב: תכנות הרשימה

שליחת המילה 0xAA - 1010 1010 לשעון אחד בלבד ולאחר מכן יש לכתוב במשך 8 שעונים 8 מילים שונות וחוקיות. לאחר סיום הכתיבה תוחזר למעבד תשובה האם הכתיבה התבצעה בהצלחה – 0x11 או שהייתה טעות – 0xEE, במקרה שהייתה טעות הבורר יעבור למצב ברירת מחדל באופן אוטומטי.

המילים החוקיות: 0x01, 0x02, 0x04, 0x08, 0x10, 0x20, 0x40, 0x80.

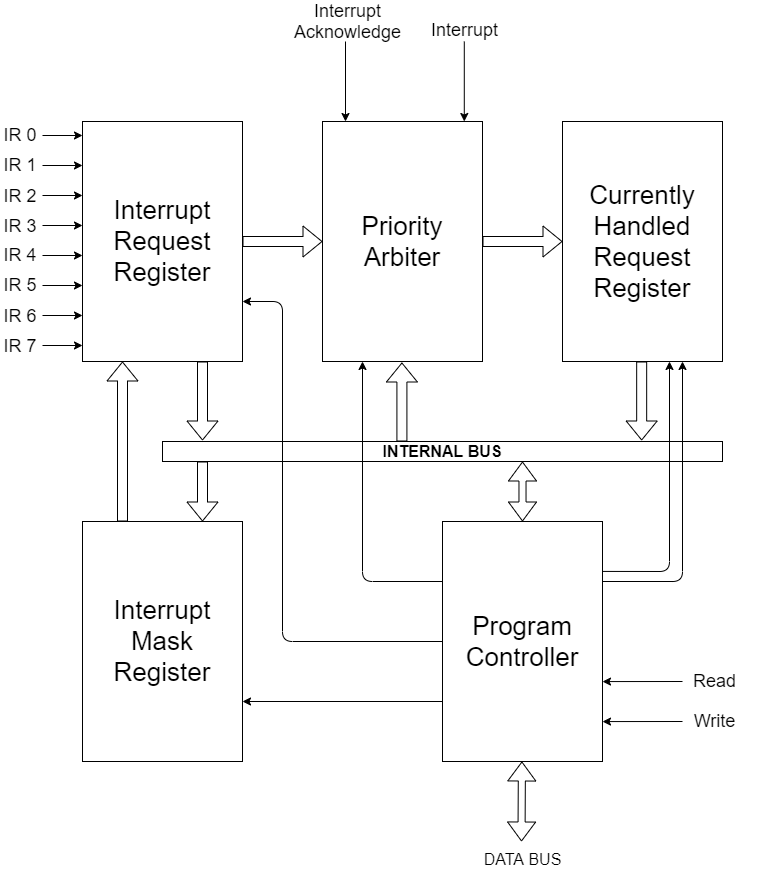
הערה: במידה ולא התקבלה מילה המתאימה לאחד המצבים הנ"ל המערכת תתעלם מהבקשה ותמשיך לפעול כרגיל.

1. איפוס אוגר הפסיקה בטיפול:

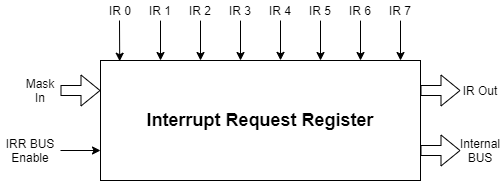
לאחר שהמעבד סיים לטפל בפסיקה הוא שולח לבקר הפסיקות את המילה 0xDB לשעון אחד כדי ליידע אותו שהפסיקה הסתיימה כך שהוא יוכל לאפס את אוגר הפסיקה בטיפול.

מפרט טכני

|  |  |  |
| --- | --- | --- |
| סעיף | מכלולי מערכת | תפקיד |
| 1 | Interrupt Request Register (IRR) | מאגד וקטור בקשות פסיקה (עד 8 בקשות) המתקבלות מרכיבי ה-I\O של המערכת. |
| 2 | Priority Arbiter (PA) | מקבל את וקטור בקשות הפסיקה מאוגר הפסיקות ומאתר את הבקשה בעלת העדיפות הגבוהה ביותר. לאחר מכן הוא שולח בקשת פסיקה למעבד. קיימים 2 מצבי עבודה – לפי רשימת ברירת מחדל או לפי רשימה מותאמת אישית. |
| 3 | Currently Handled Request Register (CHRR) | שומר את הפסיקה המטופלת על ידי המעבד בזמן אמת עד אשר הסתיים הטיפול ברכיב I/O המתאים. |
| 4 | Interrupt Mask Register (IMR) | ממסך בקשות פסיקה לפי מיקומם על גבי וקטור הפסיקות. |
| 5 | Program Controller (PC) | אחראי על ניתוב של הBUS- הפנימי של בקר הפסיקות, ועל כלל ההתקשרות עם המעבד בפרט קבלת בקשות מהמעבד ושליחת מידע מתאים. |

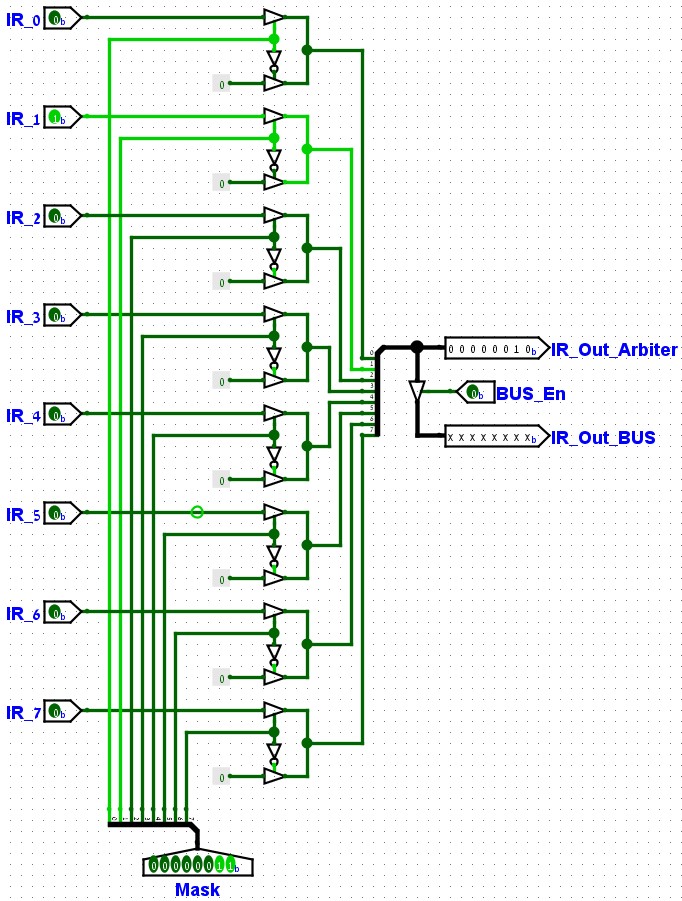
תרשים מלבני כללי

תהליכי תכנון של היחידות במערכת

Interrupt request register – IRR

תיאור כניסות ויציאות:

* IR\_0 עד IR\_7 – בקשות הפסיקה מרכיבי ה-I/O – קיימת בקשת פסיקה במצב לוגי גבוה, במידה ולא כל הכניסות מחוברות המסיכה תדאג לא לאפשר את הכניסה בביט המתאים.
* Mask In – ווקטור מיסוך בעל 8 ביטים כאשר כל ביט מאפשר בקשת פסיקה במצב לוגי גבוה עבור האינדקס אליו הוא שייך בהתאמה (LSB מתאים ל-IR\_0 וכך עד MSB המתאים ל-IR\_7).
* IR Out – ווקטור הפסיקות בעל 8 ביטים לאחר פעולת המיסוך, הווקטור מסודר כך ש-LSB מתאים ל-IR\_0 וכך עד MSB המתאים ל-IR\_7.
* Internal BUS – יציאה ל-BUS הפנימי של בקר הפסיקות של ווקטור הפסיקות.
* IR BUS Enable – מאפשר העברת ווקטור הפסיקות ל-BUS, מאופשר במצב לוגי גבוה.



Interrupt Mask Register – IMR

תיאור כניסות ויציאות:

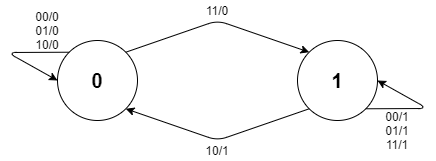
* Mask Out – יציאת ווקטור מיסוך בעל 8 ביטים השמור באוגר בתוך המערכת.

הערה: מצבו התחלתי של הווקטור הוא 0x00 עד אשר שונה.

* Internal BUS – כניסה מה-BUS הפנימי של בקר הפסיקות של ווקטור מיסוך בעל 8 ביטים לשמירה באוגר בתוך המערכת.
* IMR BUS Enable – אפשור כניסת ווקטור המיסוך מתוך ה-BUS, מאופשר במצב לוגי גבוה.

תכנון המערכת:

כניסות:

יציאות:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N.S. | | | | P.S.  Q |
|  |  |  |  |
|  |  |  |  | 0 |
|  |  |  |  | 1 |

טבלת מצבים:

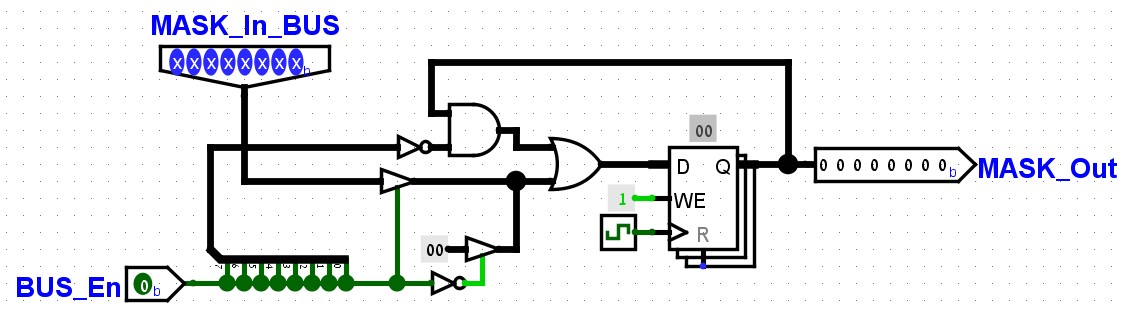
פונקציות המערכת:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 |

הכניסה Mask In היא כניסה "מתחלפת" כלומר תלויה ב-controlled buffer כך שהמידע מה-BUS הפנימי יועבר רק כאשר IMR BUS Enable עלה למעלה אחרת נכנס 0 קבוע, מתוך כך אין צורך בעוד שער AND כמו שמתואר בפונקציה למעלה מכיוון שהתוצאה זהה.

כל זאת בכדי לא להכניס מידע לא נכון מתוך ה-BUS וכדי שבכל זמן נתון תהיה כניסה תקפה שהיא או 0 או 1.

כמו כן המערכת מורכבת מ-8 מערכות mealy זהות עבור כל ביט בווקטור המסיכה.



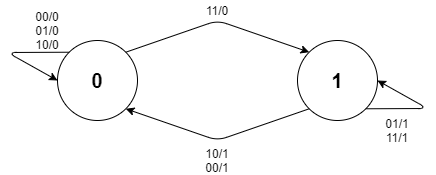
 Currently Handled Request Register – CHRR

תיאור כניסות ויציאות:

* CHR In – כניסת ווקטור הפסיקה בפעולה בעל 8 ביטים המכיל את המידע על איזה רכיב I/O קיבל את האישור לפסיקה לצורך שמירה באוגר בתוך המערכת, הביט המתאים לאינדקס של רכיב ה-I/O יהיה בגבוה והשאר בנמוך (למשך שעון אחד), בכל רגע אחר יתקבל ווקטור אפסים.
* Internal BUS – יציאה ל-BUS הפנימי של בקר הפסיקות של ווקטור הפסיקה בפעולה בעל 8 ביטים השמור באוגר בתוך המערכת.
* CHRR BUS Enable – אפשור יציאת ווקטור הפסיקה בפעולה אל ה-BUS, מאופשר במצב לוגי גבוה.
* Reset – כניסת איפוס אוגר הפסיקה בפעולה, פעיל בגבוה.

תכנון המערכת:

כניסות:

יציאות:

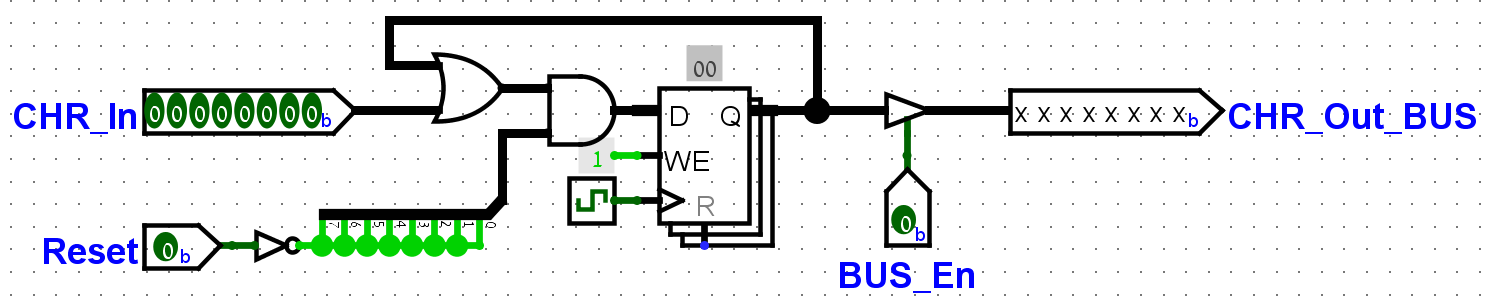
טבלת מצבים:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N.S. | | | | P.S.  Q |
|  |  |  |  |
|  |  |  |  | 0 |
|  |  |  |  | 1 |

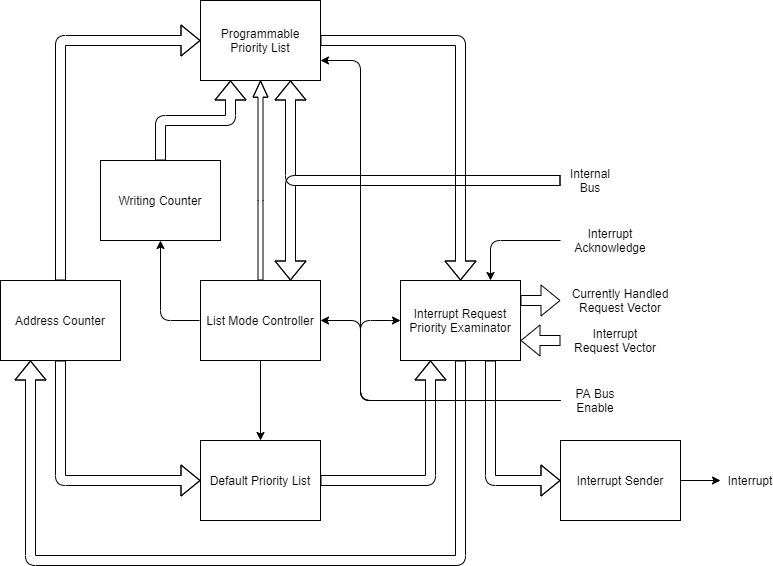
פונקציות המערכת:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 |

המערכת מורכבת מ-8 מערכות mealy זהות עבור כל ביט בווקטור הפסיקה בפעולה.



Priority Arbiter - PA

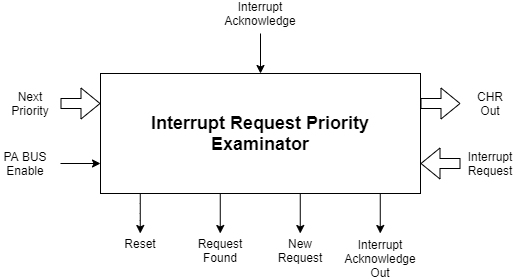
תרשים מלבנים:

אופן פעולה:

היחידה בעלת מספר פונקציות:

* קבלת וקטור פסיקות ומציאת הפסיקה בעלת העדיפות הגבוהה ביותר מתוכן.
* שליחת בקשת פסיקה למעבד בעת מציאת הפסיקה בעלת העדיפות הגבוהה ביותר.
* אפשרות עבודה לפי רשימת עדיפות ברירת מחדל או לפי רשימת בוררות מותאמת אישית.
* צריבת רשימת בוררות מותאמת אישית ע"י המעבד.
* העברת וקטור הפסיקה שבפעולה ל-CHRR בעת התחלת הטיפול בפסיקה ע"י המעבד.

תכנון המערכות:

Interrupt Request Priority Examinator

תיאור כניסות ויציאות:

* Next Priority – כניסת ווקטור בעל 8 ביטים המציין את הפסיקה הבאה לטיפול ע"י המעבד, הווקטור בנוי כך שהביט המתאים לאינדקס של רכיב ה-I/O יהיה בגבוה והשאר בנמוך.
* PA BUS Enable – כניסה של אפשור ה-BUS הפנימי של בקר הפסיקות, פעיל בגבוה.
* Reset – יציאת איפוס של מונה הכתובות.
* Request Found – יציאה המעידה על כך שנמצאה בקשת הפסיקה בעלת העדיפות הגבוהה ביותר.
* New Request – יציאה המעידה על קבלת בקשת פסיקה חדשה.
* Interrupt Acknowledge – כניסה של קבלת אישור לבקשת פסיקה מהמעבד.
* Interrupt Acknowledge Out – יציאה של קבלת אישור לבקשת פסיקה מהמעבד.
* CHR Out – יציאת ווקטור בעל 8 ביטים ליחידת ה-CHRR, ברגע קבלת אישור לבקשת פסיקה מהמעבד מועבר הווקטור המציין איזה רכיב I/O במערכת קיבל את האישור והמעבד התחיל לטפל בו.
* Interrupt Request – ווקטור בעל 8 ביטים המציין את כל בקשות הפסיקה מרכיבי ה-I/O.

זיהוי בקשות חדשות:

מערכת המשתמשת באוגר בעל 8 סיביות ומשווה האם וקטור הפסיקות שונה מווקטור הפסיקות שהיה בשעון הקודם, במידה והווקטור השתנה במוצא נקבל 1.

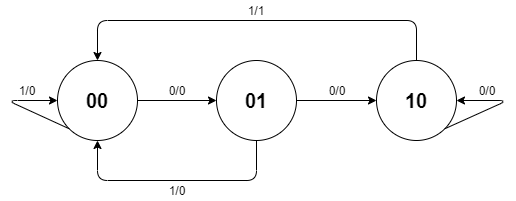
שליחת וקטור של הפסיקה בפעולה:

מעביר את הווקטור המתאים לעדיפות הגבוהה ביותר מבין כל הבקשות כאשר התקבל פולס ראשון של INT A מהמעבד.

בגלל אופן הפעולה של המערכת יש צורך במערכת המזהה שזהו הפולס הראשון ולא השני.

תכנון המערכת:

כניסה:

**יציאה:

טבלת מצבים:

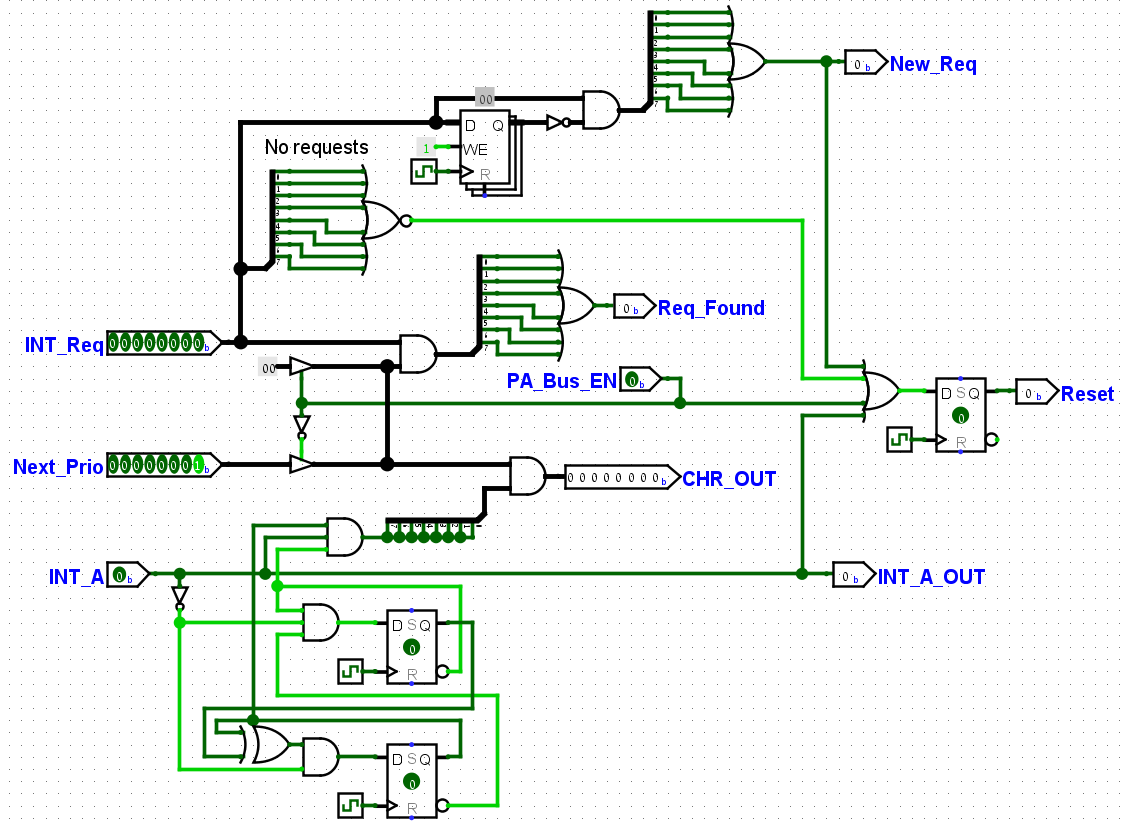
|  |  |  |
| --- | --- | --- |
| N.S. | | P.S. |
|  |  |
|  |  |  |
|  |  |  |
|  |  |  |

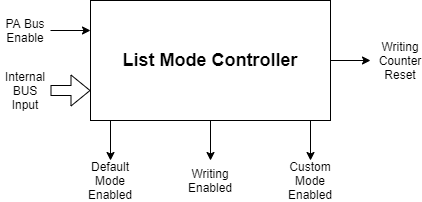
פונקציות המערכת:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 1 |  | 1 | 0 | 0 |
| 0 |  | 0 | 0 | 1 |

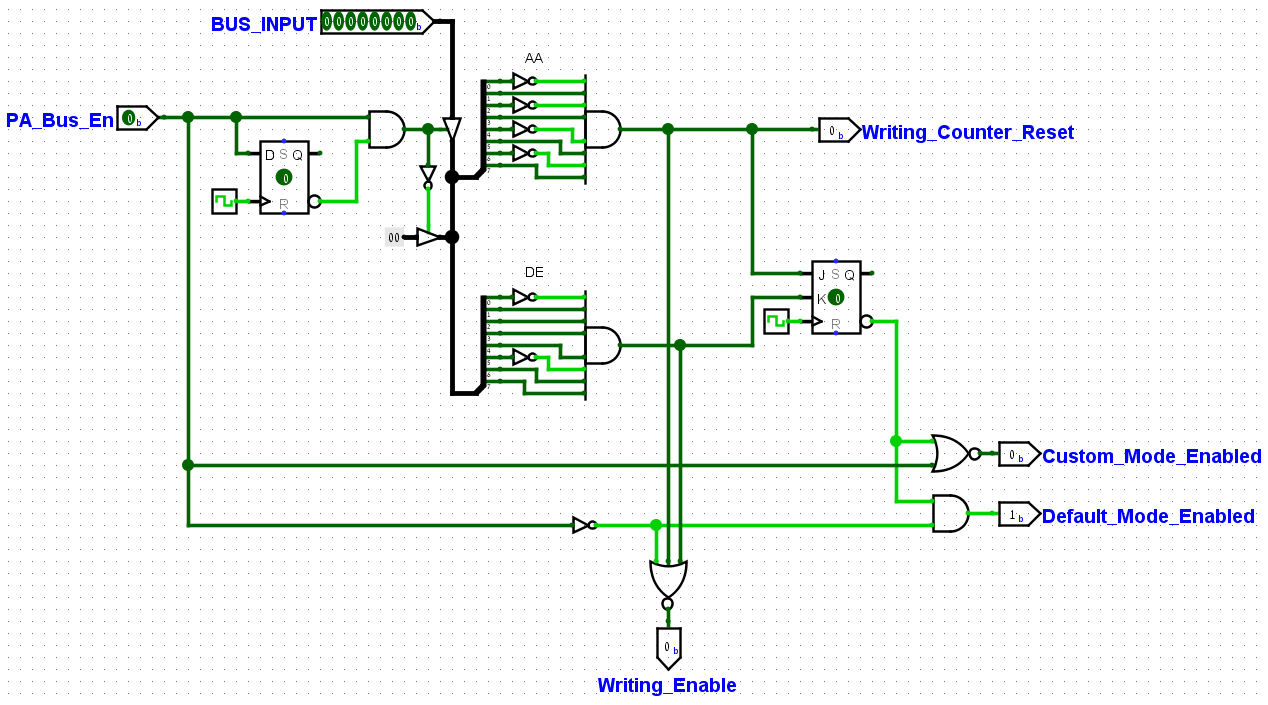
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 |  | 0 | 1 | 0 |
| 0 |  | 0 | 0 | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 |  | 0 | 0 | 0 |
| 1 |  | 0 | 0 | 1 |

**

List Mode Controller

תיאור כניסות ויציאות:

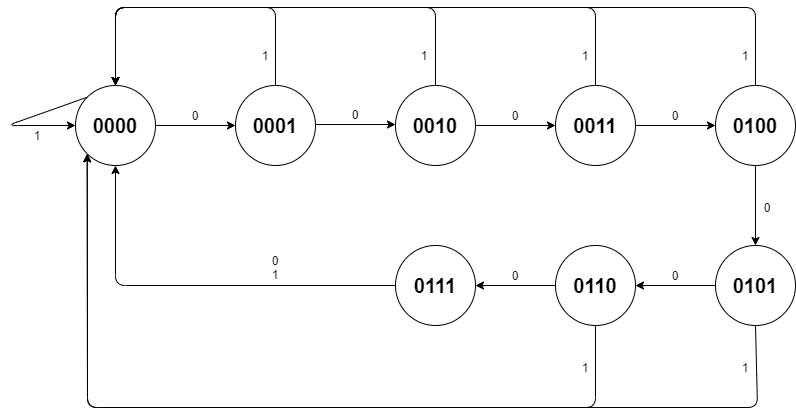
* PA BUS Enable – כניסה של אפשור ה-BUS הפנימי של בקר הפסיקות, פעיל בגבוה.
* Internal BUS Input – כניסה של המידע המתקבל מה-BUS הפנימי בבקר הפסיקות.
* Default Mode Enabled – יציאה המאפשרת עבודה במצב ברירת מחדל בעת קבלת DE בכניסת BUS Input.
* Writing Enabled – יציאה המאפשרת כתיבה לרשימת עדיפויות בעת קבלת AA בכניסת BUS Input.
* Custom Mode Enabled – יציאה המאפשרת עבודה במצב מותאם אישי לאחר סיום כתיבת רשימת העדיפויות.
* Writing Counter Reset – יציאה לאיפוס מונה הכתיבה.

Writing Counter

תיאור כניסות ויציאות:

* Reset – כניסת איפוס המונה, פעיל בגבוה.
* Address – יציאת כתובת לזיכרון.

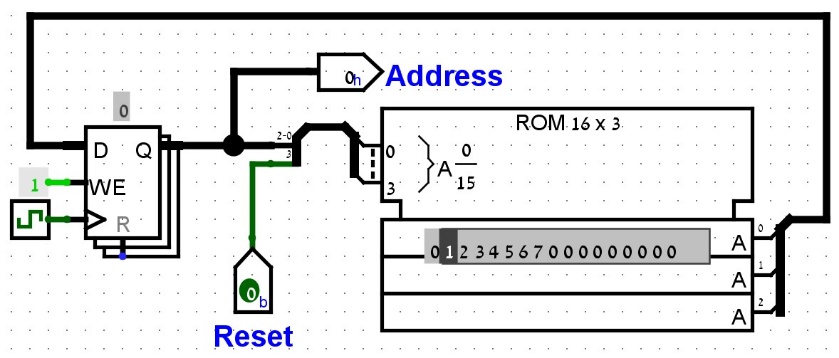
תכנון המערכת:

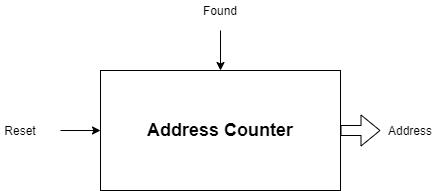
כניסה:

טבלת צריבה:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | |  | | | |  |
|  |  |  |  |  |  |  |
| 1 | 1 | 0 | 0 | *0* | *0* | *0* | *0* | *0* |
| 2 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 3 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 2 |
| 4 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 3 |
| 5 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 4 |
| 6 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 5 |
| 7 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 6 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 7 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 8 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 9 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | A |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | B |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | C |
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | D |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | E |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | F |

יציאת המערכת:



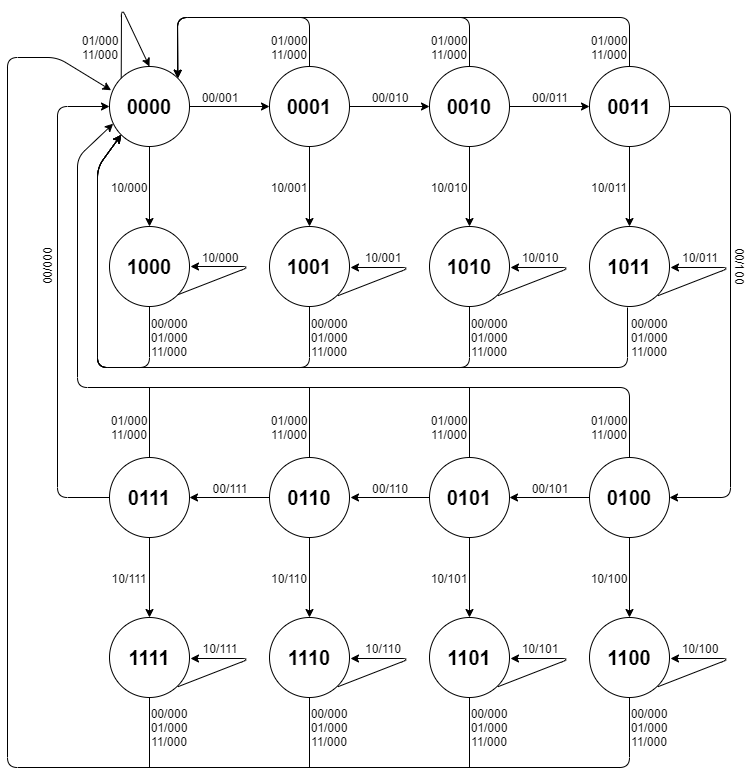
Address Counter

תיאור כניסות ויציאות:

* Reset – כניסת איפוס המונה, פעיל בגבוה.
* Found – כניסה העוצרת את המונה ונועלת את הכתובת ברגע שנמצאה בקשת פסיקה המתאימה.
* Address – יציאת כתובת לזיכרון.

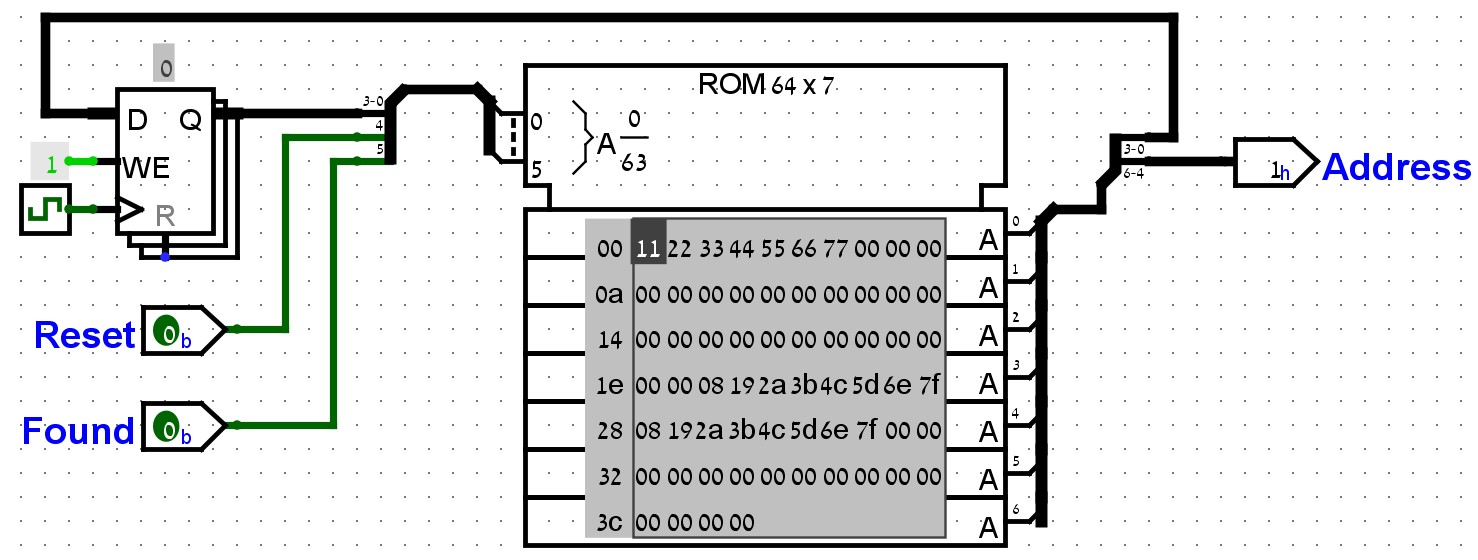
תכנון המערכת:

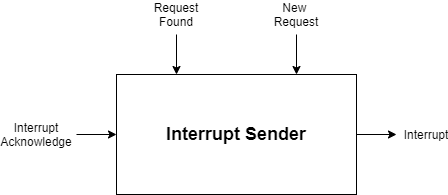
כניסות:

יציאה:

טבלת צריבה: מצורפת לקובץ זה.

**



Interrupt Sender

תיאור כניסות ויציאות:

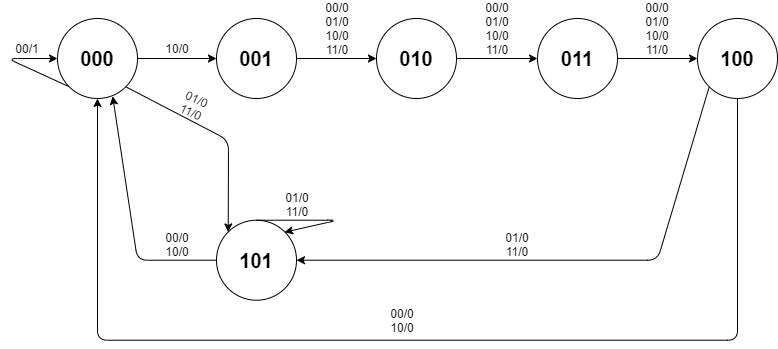
* Interrupt Acknowledge – כניסה של קבלת אישור לבקשת פסיקה מהמעבד.
* Request Found – כניסה המעידה האם נמצאה בקשת פסיקה מתאימה.
* New Request – כניסה המעידה האם התקבלה בקשת פסיקה חדשה.
* Interrupt – יציאה של שליחת בקשת פסיקה למעבד.

הערה: לאחר שליחת בקשת פסיקה וקבלת אישור מהמעבד לפסיקה הפסיקה תרד זמן מוגדר של 5 שעונים לפי דרישות המערכת.

תכנון המערכת:

כניסות:

יציאה:



טבלת מצבים:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N.S. | | | | P.S. |
|  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

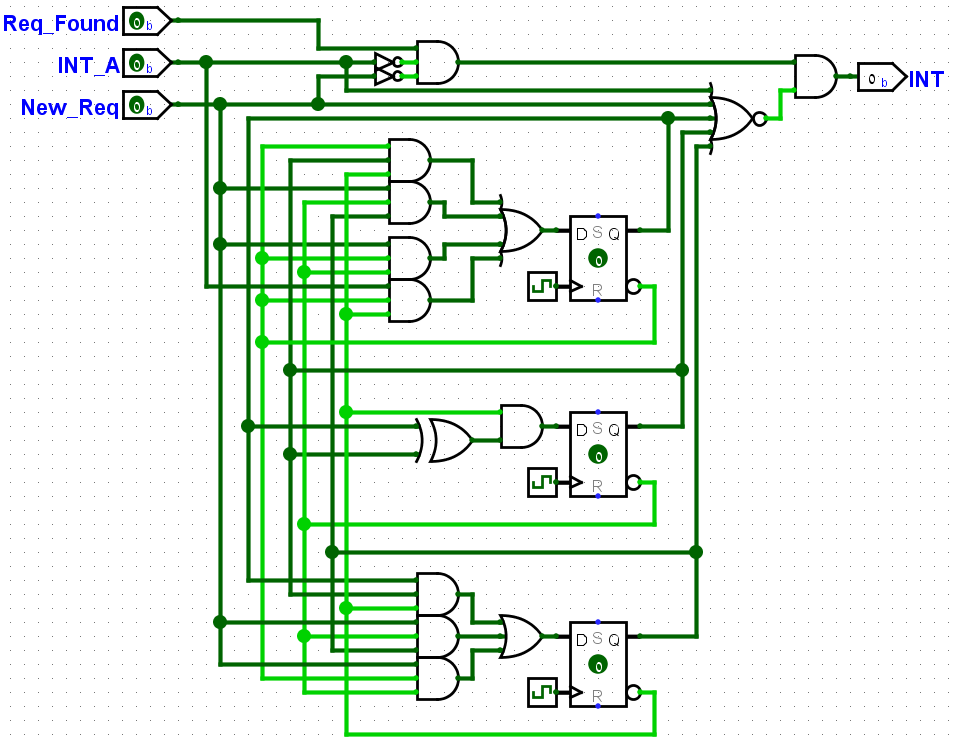
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| 0 | 0 |  |  | 0 | 0 | 0 | 1 |  |
| 0 | 0 |  |  | 0 | 0 | 0 | 0 |  |
| 0 | 0 |  |  | 0 | 0 | 0 | 0 |  |
| 0 | 0 |  |  | 0 | 0 | 0 | 0 |  |

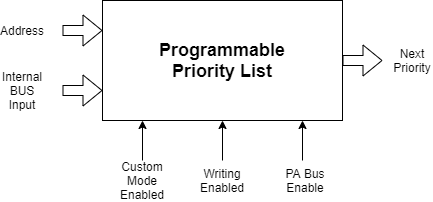
פונקציות המערכת:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| 0 | 0 |  |  | 0 | 1 | 0 | 0 |  |
| 1 | 1 |  |  | 0 | 1 | 0 | 1 |  |
| 1 | 1 |  |  | 0 | 1 | 0 | 1 |  |
| 0 | 0 |  |  | 0 | 1 | 0 | 0 |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| 0 | 0 |  |  | 1 | 0 | 1 | 0 |  |
| 0 | 0 |  |  | 1 | 0 | 1 | 0 |  |
| 0 | 0 |  |  | 1 | 0 | 1 | 0 |  |
| 0 | 0 |  |  | 1 | 0 | 1 | 0 |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| 0 | 0 |  |  | 1 | 0 | 0 | 0 |  |
| 1 | 1 |  |  | 1 | 0 | 0 | 1 |  |
| 1 | 1 |  |  | 1 | 0 | 0 | 1 |  |
| 0 | 0 |  |  | 1 | 0 | 0 | 1 |  |



Programmable Priority List

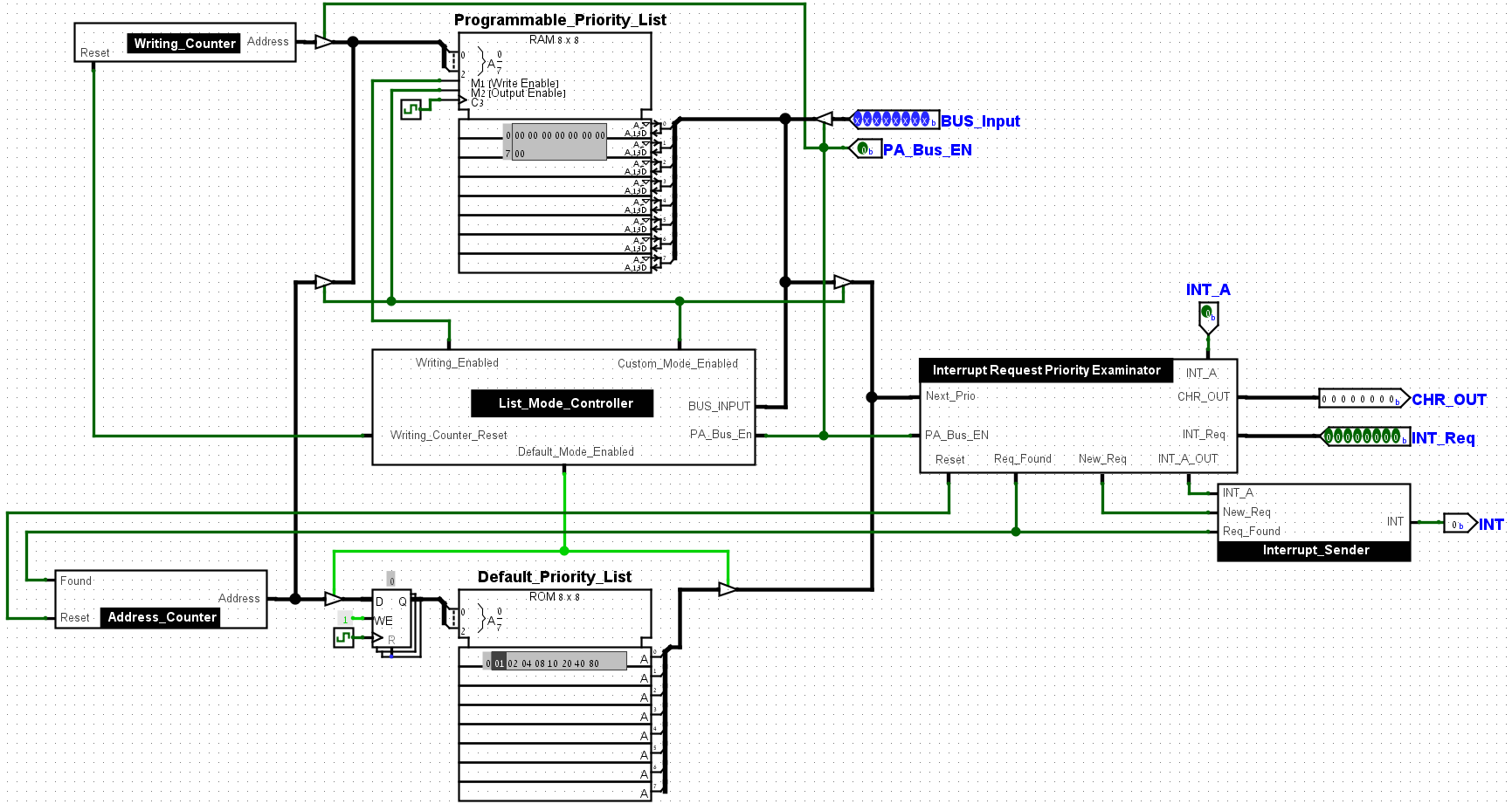
תיאור כניסות ויציאות:

* Address – כניסת הכתובת לזיכרון של רשימת העדיפויות לצרכי קריאה או כתיבה.
* Internal BUS Input – כניסת המידע מה-BUS הפנימי של בקר הפסיקות לצורך כתיבת הרשימה.
* Custom Mode Enabled – כניסה לאפשור עבודה במצב מותאם אישית, פעיל בגבוה.
* Writing Enabled – כניסה לאפשור כתיבה לזיכרון של רשימת העדיפויות, פעיל בגבוה.
* PA BUS Enable – כניסה של אפשור ה-BUS הפנימי של בקר הפסיקות, פעיל בגבוה.
* Next Priority – יציאה של ווקטור בעל 8 ביטים שבו מידע מהו רכיב ה-I/O שבקשת הפסיקה שלו נבדקת כרגע.

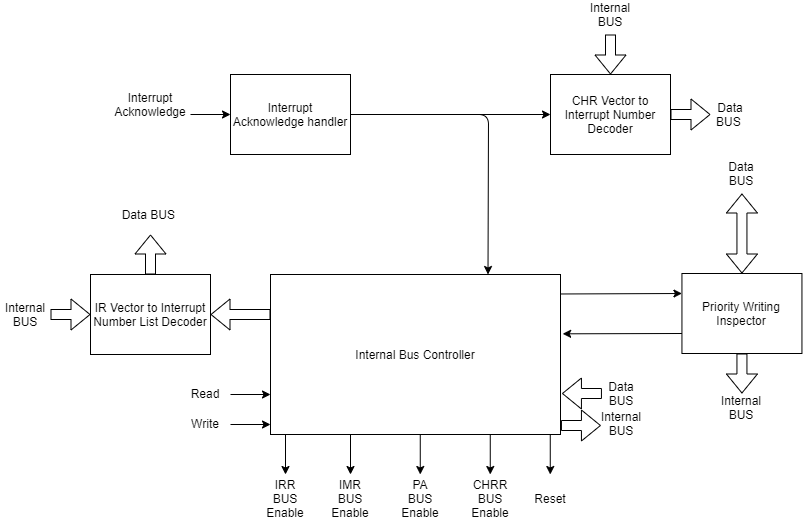
Default Priority List

תיאור כניסות ויציאות:

* Address – כניסת הכתובת לזיכרון של רשימת העדיפויות לצורך קריאה.
* Default Mode Enabled – כניסה לאפשור עבודה במצב ברירת מחדל, פעיל בגבוה.
* Next Priority – יציאה של ווקטור בעל 8 ביטים שבו מידע מהו רכיב ה-I/O שבקשת הפסיקה שלו נבדקת כרגע.



Program Controller – PC

תרשים מלבנים:

אופן פעולה:

היחידה בעלת מספר פונקציות:

* שליטה על הערוץ (BUS) הפנימי של המערכת.
* מטפלת בתחילת תהליך הפסיקה מול המעבד.
* אחראית על תיווך בין המעבד לכל אחת מהיחידות השונות במערכת לצרכי קריאה וכתיבה של מידע.
* מפקחת על תהליך כתיבת רשימת עדיפויות ל-PA ובודקת האם הרשימה כולה תקינה (מילים חוקיות ללא חזרות), במידה וקיימת שגיאה תיידע את המעבד ותעביר את ה-PA למצב ברירת מחדל, במידה ואין שגיאה תיידע את המעבד על כך.
* פיענוח וקטור הפסיקות מתוך IRR לרשימת מספרי פסיקה ושליחתה למעבד.
* פיענוח וקטור הפסיקה בפעולה למספר פסיקה המתאים והעברתו למעבד.

תכנון המערכות:

IR Vector to Interrupt Number List Decoder

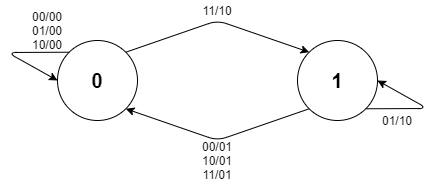
תיאור כניסות ויציאות:

* Internal BUS Input – כניסת ווקטור הפסיקות בעל 8 ביטים מה-BUS הפנימי של בקר הפסיקות.
* DATA BUS Output – יציאה של מספרי פסיקה של רכיבי ה-I/O ל-BUS החיצוני המחובר למעבד.
* Read – כניסת בקרה מהמעבד למצב קריאה מבקר הפסיקות, פעיל בגבוה.
* IRR BUS Enable – כניסה המעידה האם יש מידע המגיע מ-IRR דרך ה-BUS הפנימי, פעיל בגבוה.

במערכת קיימת מכונת מצבים לשליטה על העברת המידע וסיום התהליך:

תכנון המערכת:

כניסות:

יציאות:

טבלת מצבים:

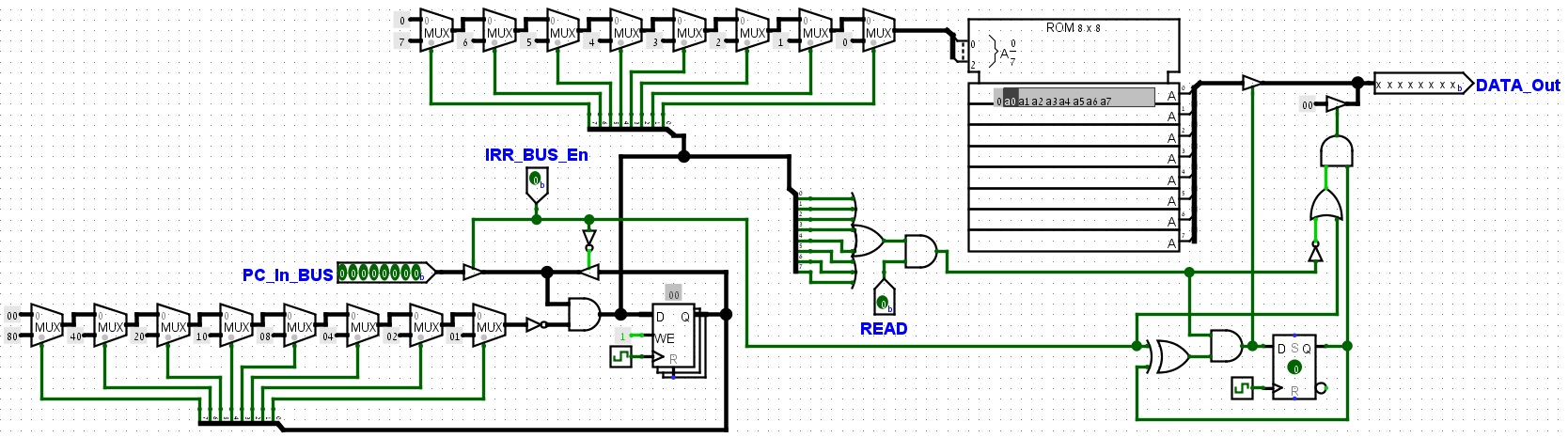
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N.S. | | | | P.S. |
|  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

פונקציות המערכת:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |

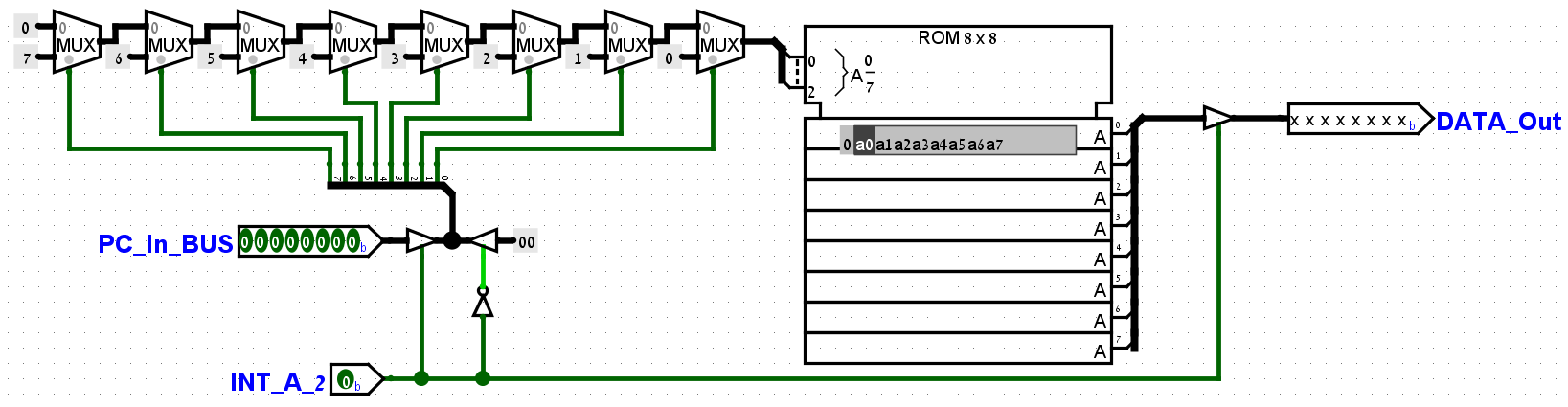
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |

**

CHR Vector to Interrupt Number Decoder

תיאור כניסות ויציאות:

* Internal BUS Input – כניסת ווקטור הפסיקה בפעולה בעל 8 ביטים מה-BUS הפנימי של בקר הפסיקות.
* DATA BUS Output – יציאה של מספר הפסיקה של רכיב ה-I/O המתאים ל-BUS החיצוני המחובר למעבד.
* 2nd Interrupt Acknowledge Pulse – כניסה של המידע האם התקבל פולס שני של אישור מהמעבד, פעיל בגבוה.

**

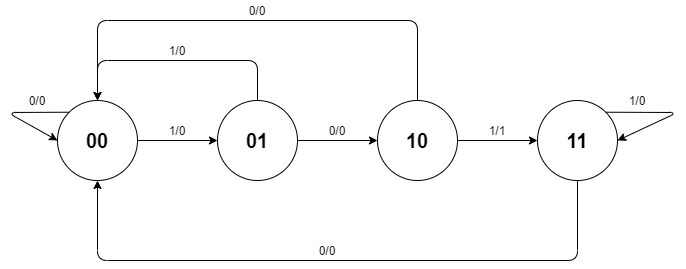
Interrupt Acknowledge Handler

תיאור כניסות ויציאות:

* Interrupt Acknowledge – כניסה של קבלת אישור לבקשת פסיקה מהמעבד, פעיל בגבוה.
* 2nd Interrupt Acknowledge Pulse – יציאה של המידע האם התקבל פולס שני של אישור מהמעבד.

תכנון המערכת:

כניסה:

**יציאה:

טבלת מצבים:

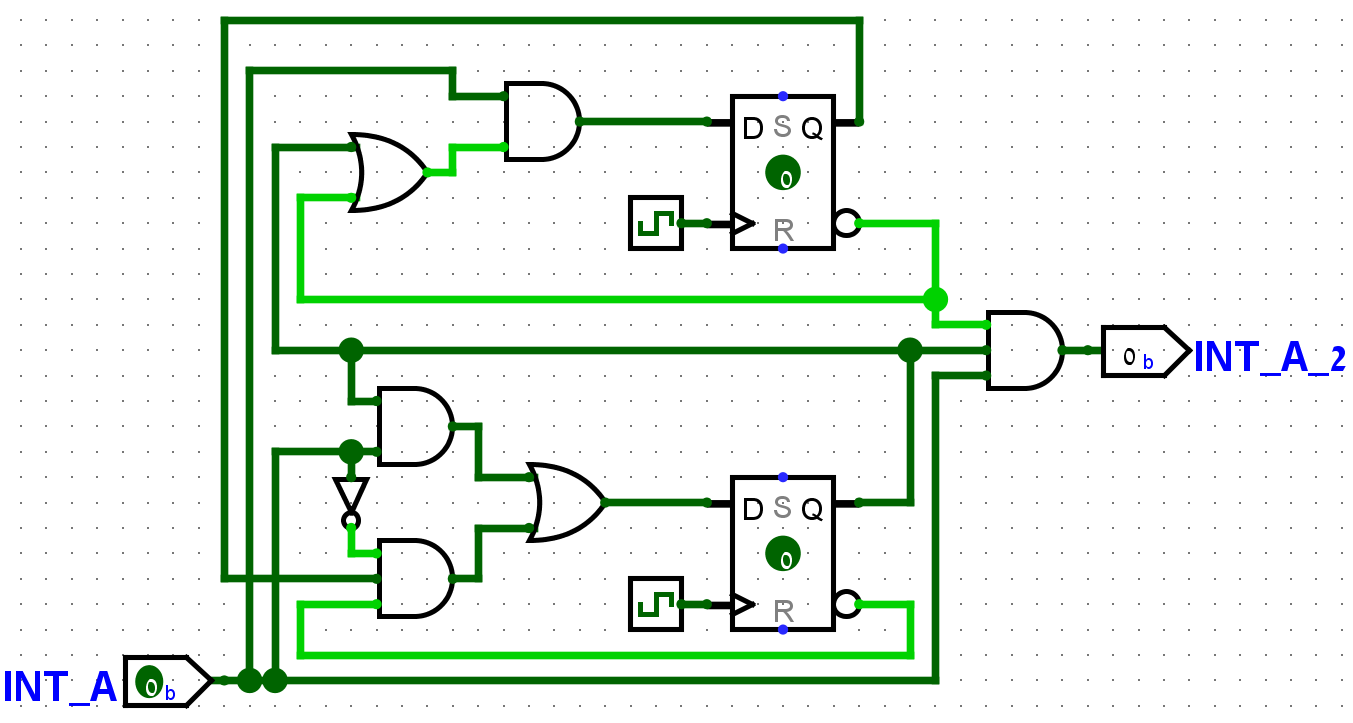
|  |  |  |
| --- | --- | --- |
| N.S. | | P.S. |
|  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

פונקציות המערכת:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |



Priority Writing Inspector

תיאור כניסות ויציאות:

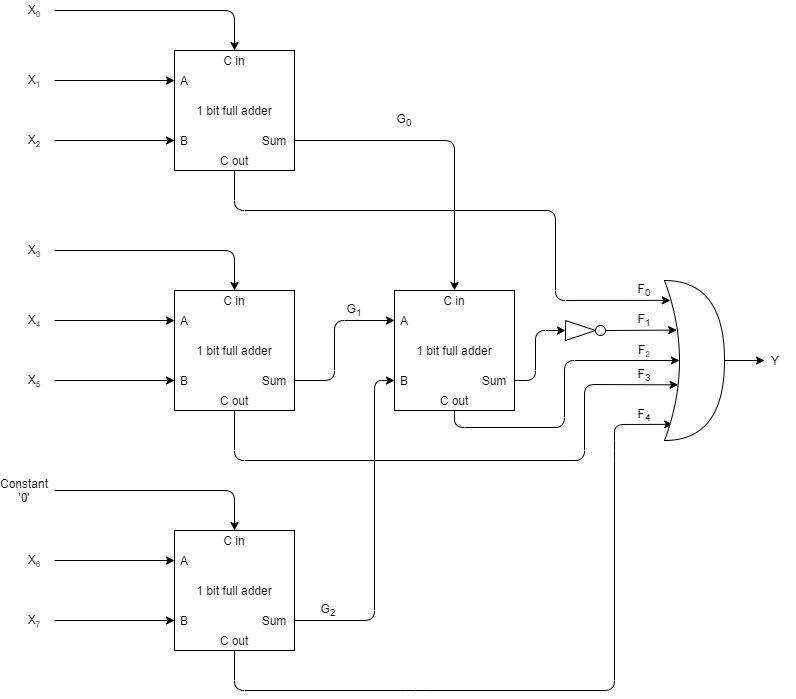
* DATA BUS Input – כניסה של המידע הנכתב לרשימת בוררויות המותאמת אישית מה-BUS החיצוני המחובר למעבד.
* DATA BUS Output – יציאה ל-BUS החיצוני המחובר למעבד לצורך הודעה למעבד בסיום הכתיבה האם הרשימה שהתקבלה תקינה או שקיימת שגיאה.
* Internal BUS Output – יציאה ל-BUS הפנימי של בקר הפסיקות לצורך החזרת המערכת למצב עבודה על פי רשימת ברירת מחדל במידה וקיימת שגיאה ברשימה שהתקבלה.
* Reset to Default – יציאה לצורך הודעה על העברת המערכת למצב עבודה על פי רשימת ברירת מחדל במידה וקיימת שגיאה ברשימה שהתקבלה.
* PA BUS Enable – כניסה המעידה האם מתבצעת כתיבה ל-PA דרך ה-BUS הפנימי של בקר הפסיקות.

הערה: הרשימה תכיל 8 מילים שונות, המילים החוקיות הן:

0x01, 0x02, 0x04, 0x08, 0x10, 0x20, 0x40, 0x80.

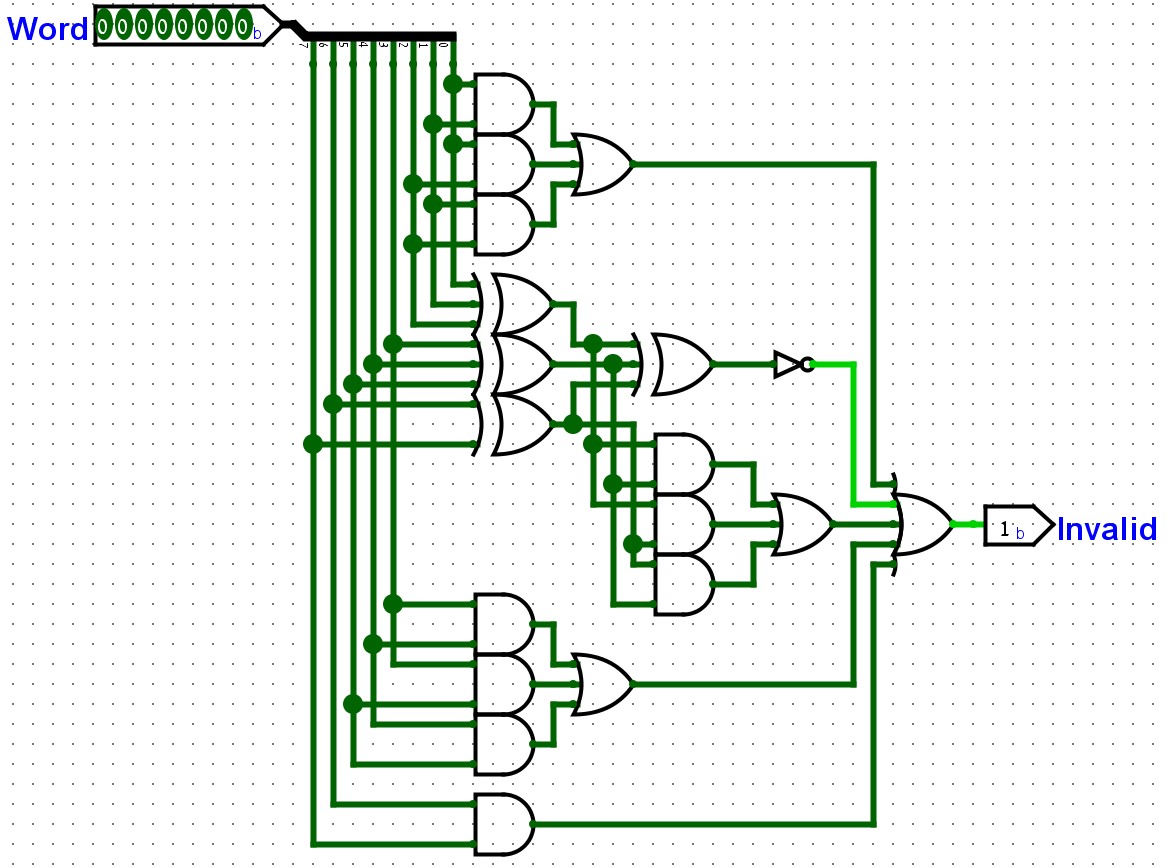
תכנון תתי המערכות:

Valid Word Detector

תהליך התכנון התחיל ממערכת של ארבעה 1bit Full adder עם שער OR לפי השרטוט הבא:

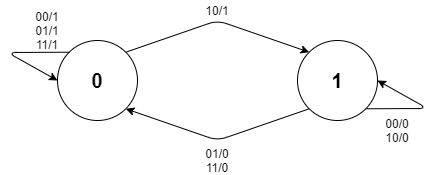
את כל אחד מרכיבי ה-Full adder מימשנו בשערים לוגים, לפי פונקציות המוצא הידועות שלהם:

מכאן פונקציית המוצא שהתקבלה:



Valid Word Detector Register

כניסות:

*יציאה:*

טבלת מצבים:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N.S. | | | | P.S. |
|  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

פונקציות המערכת:

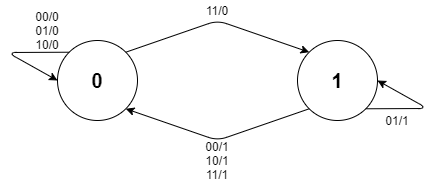
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 1 |

All the words are different Detector

כניסות:

*יציאה:*



*טבלת מצבים:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N.S. | | | | P.S. |
|  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

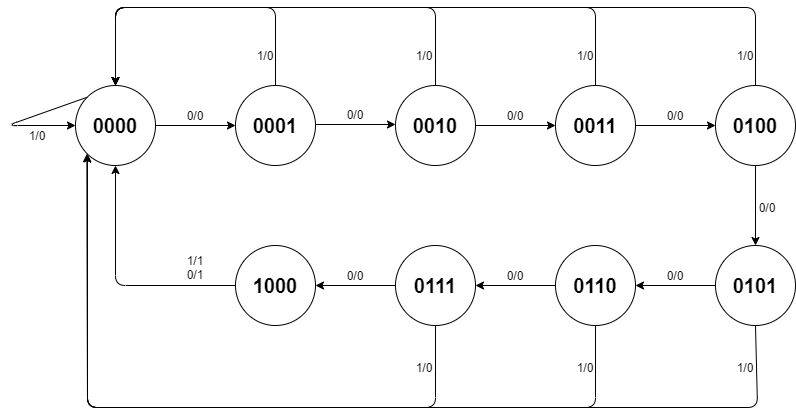
פונקציות המערכת:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

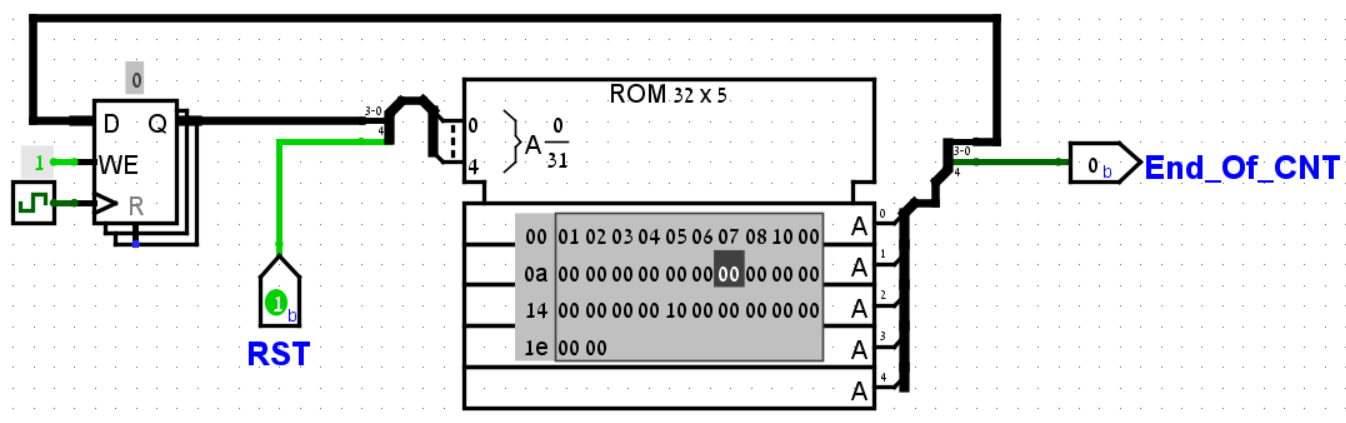
Counter 0 to 8

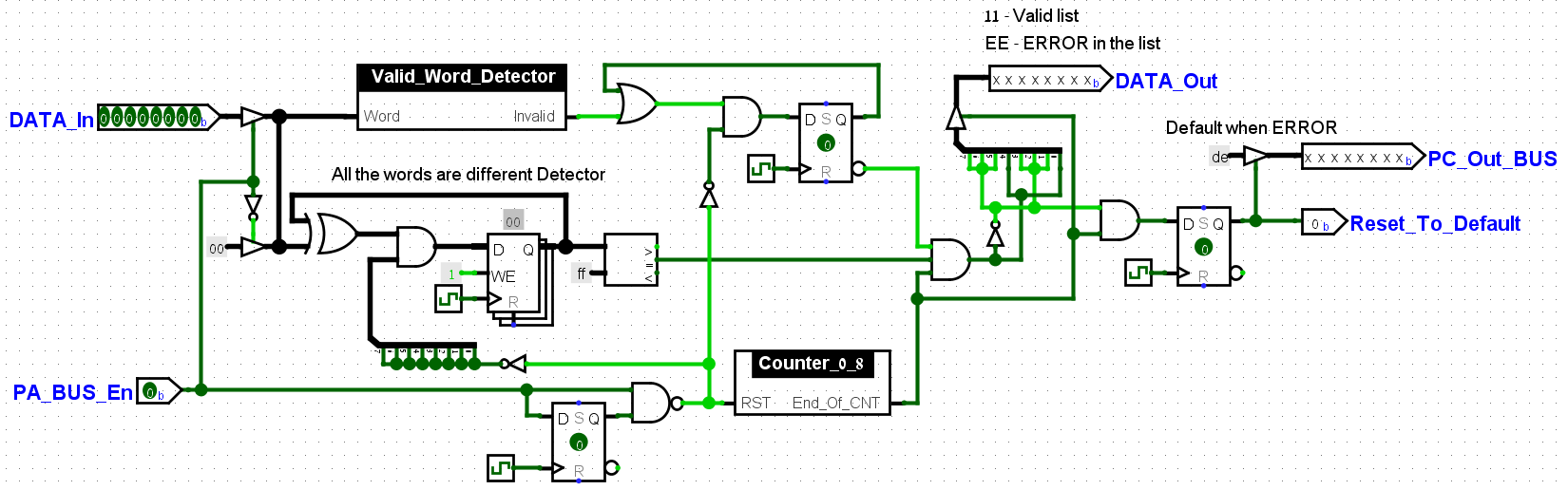
כניסה:

*יציאה:*

טבלת צריבה: מצורפת לקובץ זה.

**



**

Internal BUS Controller

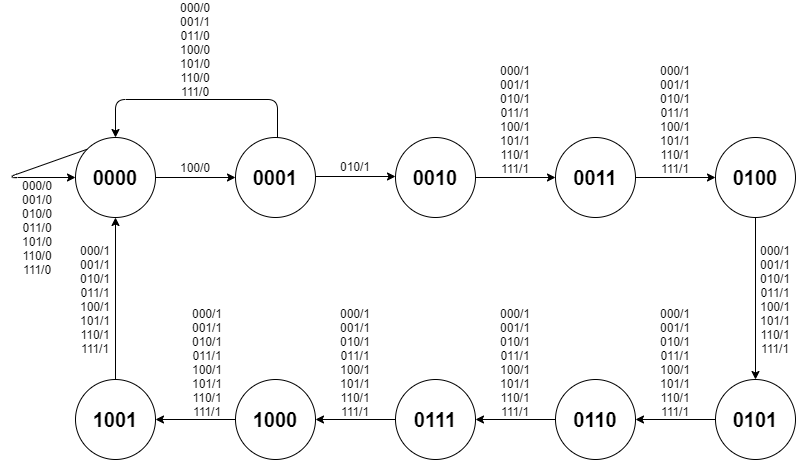
תיאור כניסות ויציאות:

* DATA BUS Input – כניסה של המידע מה-BUS החיצוני המחובר למעבד לצרכי זיהוי מילים והעברת המידע ליחידות השונות בבקר דרך ה-BUS הפנימי.
* Internal BUS Output – יציאה ל-BUS הפנימי של בקר הפסיקות לצורך העברת מידע מהמעבד ישירות ליחידות השונות בבקר במידה ונדרש לכך מהמעבד.
* Reset to Default – כניסה המעידה על כך שהמערכת צריכה לעבור למצב עבודה על פי רשימת ברירת מחדל, פעיל בגבוה.
* Write – כניסת בקרה מהמעבד למצב כתיבה לבקר הפסיקות, פעיל בגבוה.
* Read - כניסת בקרה מהמעבד למצב קריאה מבקר הפסיקות, פעיל בגבוה.
* 2nd Interrupt Acknowledge Pulse – כניסה של המידע האם התקבל פולס שני של אישור מהמעבד, פעיל בגבוה.
* IRR BUS Enable – יציאת אפשור קבלת מידע מ-IRR דרך ה-BUS הפנימי של בקר הפסיקות.
* CHRR BUS Enable – יציאת אפשור קבלת מידע מ-CHRR דרך ה-BUS הפנימי של בקר הפסיקות.
* IMR BUS Enable – יציאת אפשור שליחת מידע ל-IMR דרך ה-BUS הפנימי של בקר הפסיקות.
* PA BUS Enable – יציאת אפשור שליחת מידע ל-PA דרך ה-BUS הפנימי של בקר הפסיקות.
* Reset – יציאת איפוס של CHRR בסיום ההתקשרות בין המעבד לרכיב ה-I/O שקיבל שירות ממנו.
* Read Out – יציאה המעידה האם המעבד דורש קריאה מבקר הפסיקות.

*תכנון תתי המערכות:*

PA BUS Enable

כניסות:

*יציאה:*

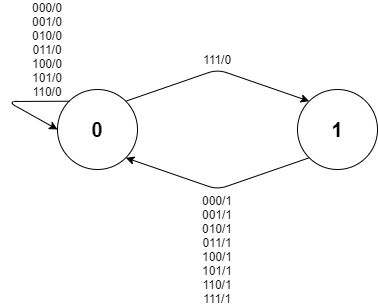
טבלת צריבה: מצורפת לקובץ זה.

**

קיימות 3 מערכות מצבים זהות באופן פעולתן:

CHRR BUS Enable, IRR BUS Enable, IMR BUS Enable

*כניסות:*

*יציאה:*

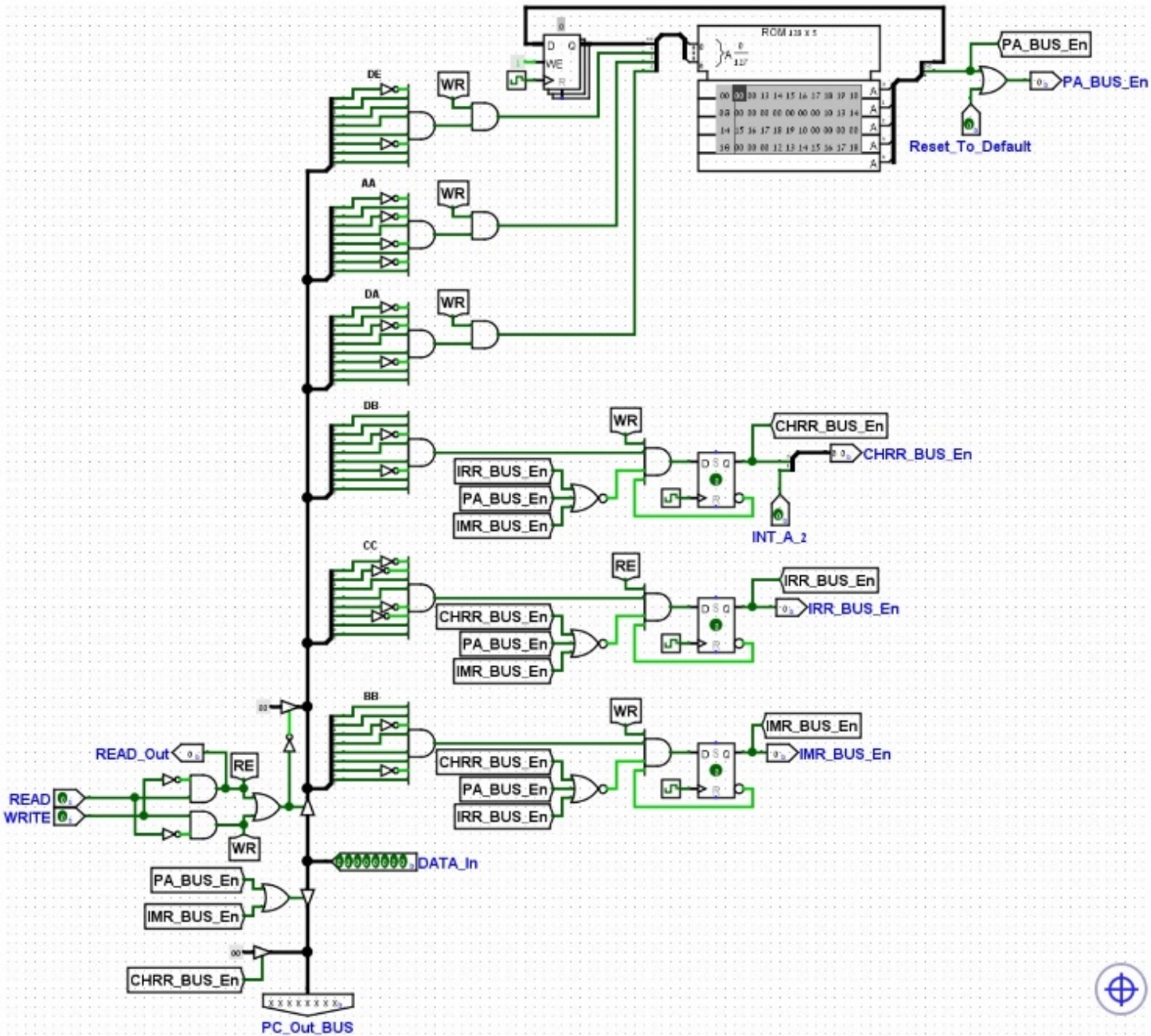
*טבלת מצבים:*

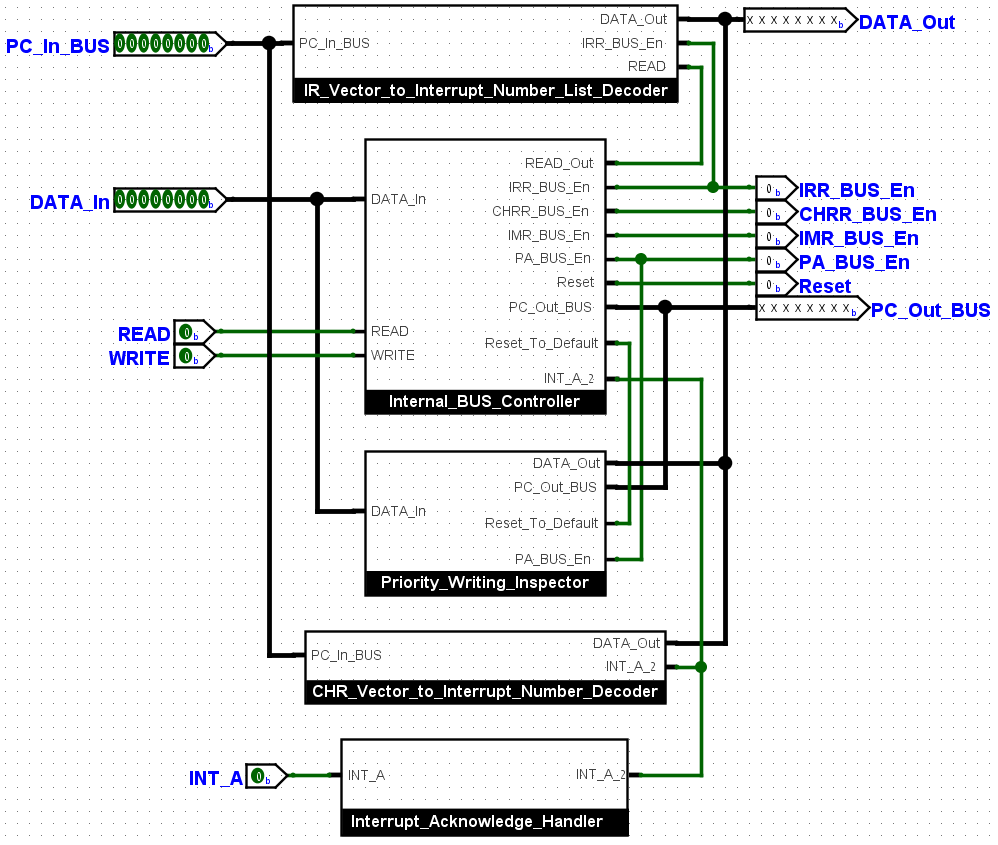
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N.S. | | | | | | | | P.S. |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

פונקציות המערכת:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 0 | 0 | 0 | 00 |
| 0 | 0 | 0 | 0 | 10 |
| 0 | 0 | 0 | 0 | 11 |
| 0 | 1 | 0 | 0 | 10 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 11 | 01 | 00 |  |
| 0 | 0 | 0 | 0 | 00 |
| 1 | 1 | 1 | 1 | 10 |
| 1 | 1 | 1 | 1 | 11 |
| 0 | 0 | 0 | 0 | 10 |



**

חישוב תזמוני המערכת

חישוב מדגמי של מסלולי המערכת:

*מתוך המשוואות הקריטיות:*

סימוכין

בקר הפסיקות תוכנן בהשראת בקר הפסיקות Intel 8259:

https://tinyurl.com/y9lfakr2

מפרטי הרכיבים בהם השתמשנו:

D-FF:

<https://tinyurl.com/yc8scp6d>

JK-FF:

https://tinyurl.com/yc2pmagh

NOT gate:

https://tinyurl.com/yb355tf4

2 input AND gate:

<https://tinyurl.com/y9j8sgj5>

2 input OR gate:

https://tinyurl.com/y8gskrpc

2 Input XOR gate:

<https://tinyurl.com/ycmhyytm>

2 Input NAND gate:

<https://tinyurl.com/y9432njc>

2 Input NOR gate:

<https://tinyurl.com/y7exucqf>

3 Input AND gate:

<https://tinyurl.com/y8dd7u6r>

3 input OR gate:

<https://tinyurl.com/yacz4l7x>

3-input XOR gate:

<https://tinyurl.com/s22s4pl>

3 input NOR gate:

<https://tinyurl.com/y7mm9l7g>

4 input AND gate:

<https://tinyurl.com/y9r5a9rh>

4 input OR gate:

<https://tinyurl.com/y8lujyg9>

8 Input NAND gate:

<https://tinyurl.com/ycnosntt>

8 Input NOR/OR gate:

<https://tinyurl.com/y9zch5xb>

Bus Buffer:

<https://tinyurl.com/y84j5r8y>

2 input multiplexer:

<https://tinyurl.com/ydb8nsfp>

8-bit magnitude comparator:

<https://tinyurl.com/webx678>

512x8 bit ROM:

<https://tinyurl.com/ybowbyur>

64Kx8 bit RAM:

<https://tinyurl.com/y8obyea4>