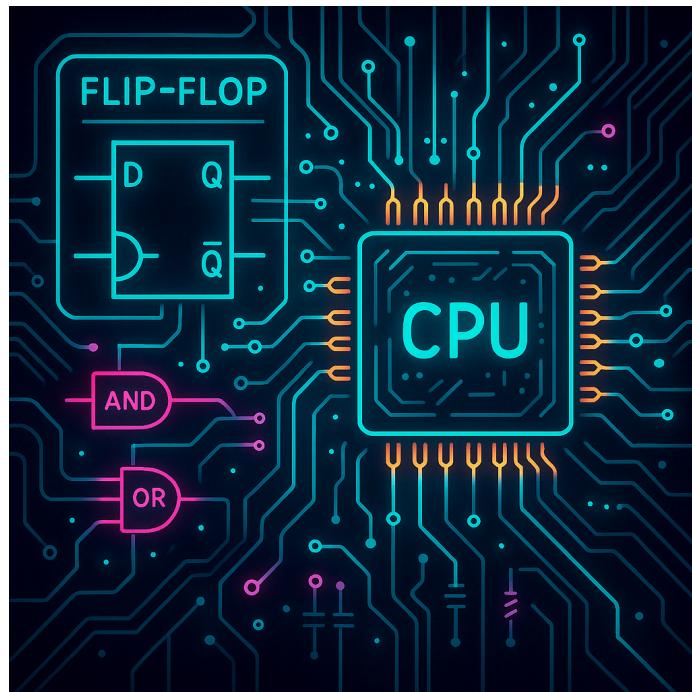


LAB 2

ADVANCED CPU ARCHITECTURE AND HARDWARE ACCELERATORS LAB

IDO RON : 322384330

BENYAMIN OUMANSKY: 322688946



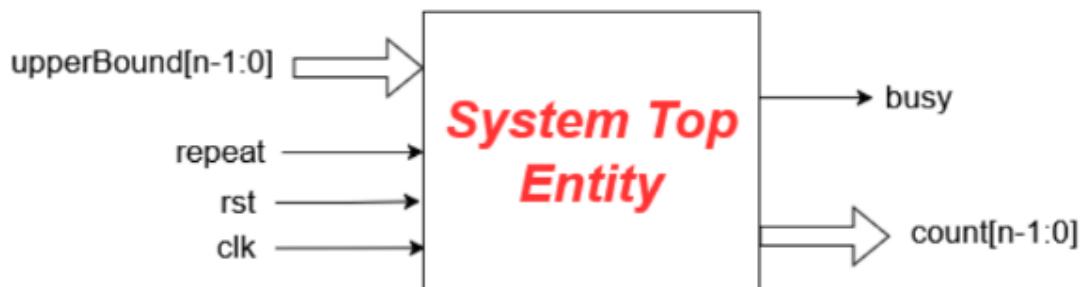
מבוא

במעבדה זו המטרה היא התמימות בקוד רציף (Sequential logic) לעומת מעבדה 1 בה בוצע שימוש בקוד מקבילי (Combinational logic).

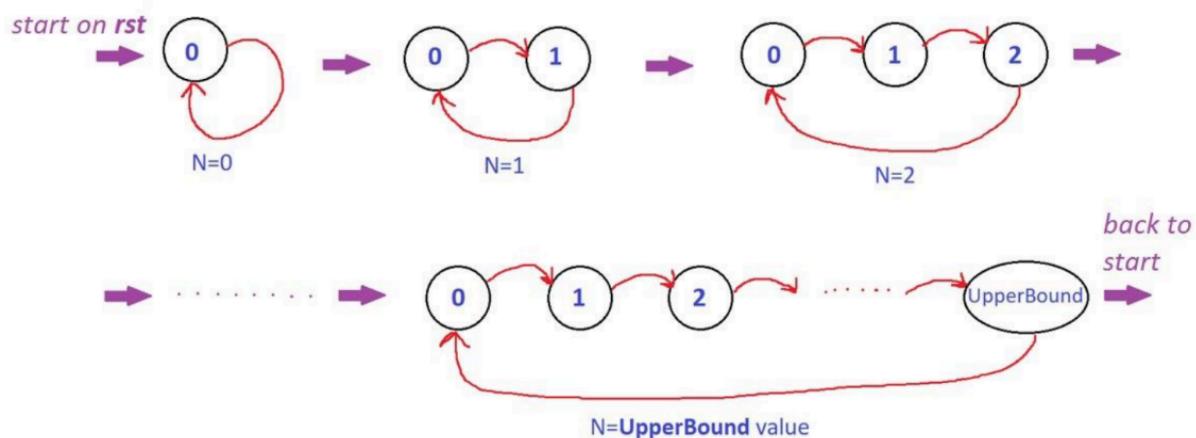
במשימה זו נ命asz מערכת דיגיטלית סינכרונית המקבלת 3 קלטים: וקטור כניסה של ערך מקסימלי: "[n-1:0]Upperbound", אותן שאחראן על איפוס "rst", אותן כניסה של שעון "clk" ואות שאחראן על חזרה "repeat".

המערכת תוציא ביט שמסמל אם היא עסוקה "busy" ומונה "Count[n-1,0]" אוון הפעולה באוון כללי: המ מערכת מישמת מונה שגדל באוון דינמי (ctalות במונה שני) בכל סיבב עד שהוא מגיע לערך UpperBound, כאשר הקלט repeat שווה ל-1 אז המונה מתאפס וחוזר שוב על הפעולה של הספירה.

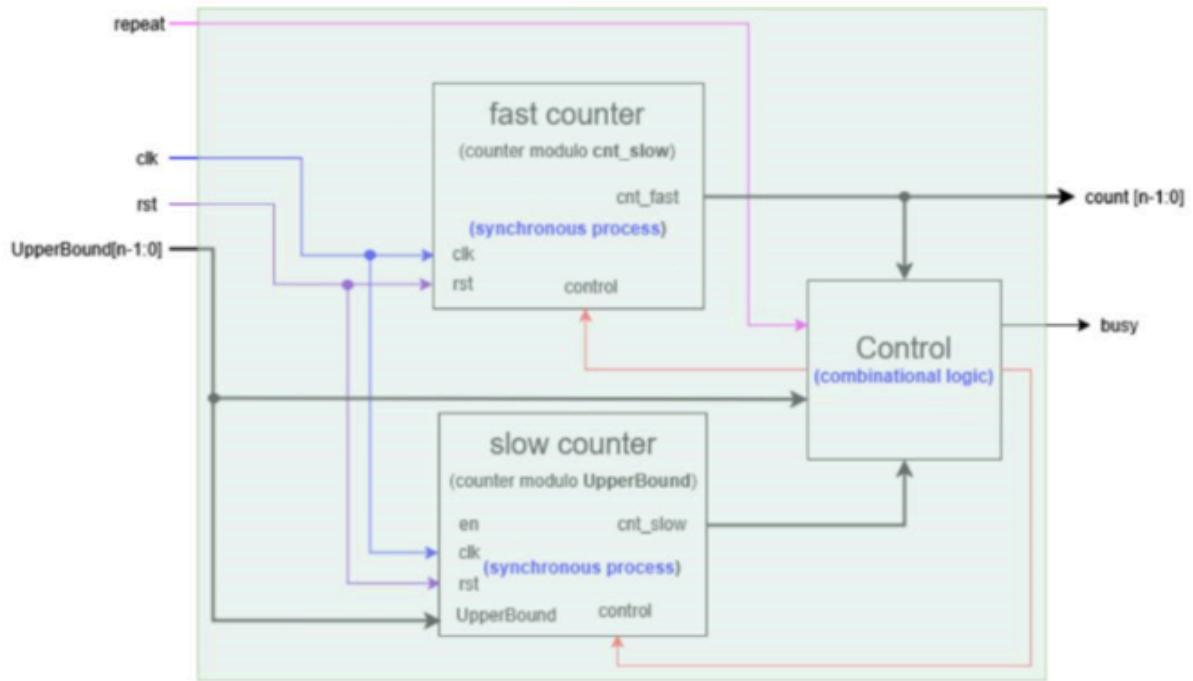
המערכת :



תיאור של המערכת על ידי דיאגרמת מצבים :



תרשים בלוקים של מודול מערכתי:



System top entity:

Inputs:

- Repeat
- Rst
- Clk
- Upperbound

Outputs:

- Count
- Busy

זהו הרכיב היחיד במערכת, כלומר לא מימשנו עוד קומponentות בניגוד למעבדה הראשונה בה ה-TOP היה מרכיבים.

כאן מימשנו מודולים: שני מונחים באמצעות Process כלומר Sequential logic ואת Control באמצעות Combinational logic.

Fast counter (module 1):

Input :

- Clk
- Rst
- Control

Output :

- Count

המונה המהיר: החסם שלו הוא מוצא המונה האיטי (שאותו הוא מקבל מה (control), המונה המהיר סופר כל פעם עד שmagiu לערך המונה האיטי, כך עד שהמונה האיטי מגיע לחסם העליון של המערכת כולה. המונה המהיר מקבל מידע מבЛОק ה - Control אשר קובע את פועלתו (הסביר בהמשך). המוצא של המונה המהיר הוא גם מוצא המערכת כולה.

Slow Counter (module 2):

Input :

- Cnt_slow
- Cnt_fast
- Upperbound
- Repeat

Output:

- Cnt_slow

המונה האיטי:

המונה האיטי סופר עד ה - Upperbound ומתקדם ב - 1 בכל פעם שהמונה המהיר מגיע אליו. במידה והגיא אל ה- Upperbound (או עבר אותו) הוא מתאפס כתלות בביט ה- repeat ובמידע מבЛОק ה - Control אשר קובע את פועלתו (הסביר בהמשך).

Control (module 3):

Inputs:

- Repeat
- Rst
- Clk
- Upperbound

Outputs:

- Busy

התפקיד של Control

הוא לא גשר בין שני המונחים: שני המונחים שולחים בכל עליית שעון את ערכם כסיגナル ל-control.

control שולח למונה המהיר את ערכו של המונה האיטי כך שהמונה המהיר ידע עד לאן עליו לספר. בנוסף הוא שולח למונה המהיר עוד בית שאחראי להגדיר לו האם ערכו ('1') עליו להמשיך לספר ברגע שגיעה למונה האיטי, ככלומר לאפס את עצמו ולהמשיך לספר שוב או במקרה שערכו ('0') לעצור ברגע שגיעה לערכו של המונה האיטי.

ה-control שולח סיגナル של בית בודד למונה האיטי שתפקידו להגדיר לו אם להמשיך ('1') בספירה ברגע שהמונה האיטי הגיע אליו או אם לעצור ('0'). ערכו של סיג널 זה תלוי באם המונה המהיר הגיע לערכו של המונה האיטי, ביחסם ההפוך וב-repeat.

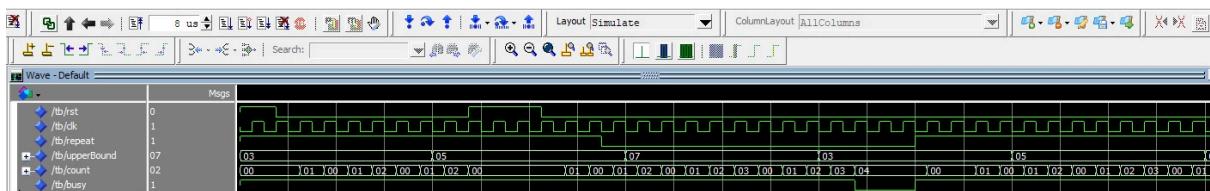
אם המונה המהיר הגיע למונה האיטי והמונה האיטי הגיע לחסם (או אפילו עבר אותו במידה והחסם קטן תוך כדי הריצה) ו-repeat כבוי אז עליו לשדר למונה האיטי 0 אבל אם repeat דלוק אז עליו לשדר 1 ומונה האיטי צריך להתחילה את הספירה מחדש.

בנוסף, ה-control אחראי על מוצאת המערכת `uses` כך שכל עוד המונה המהיר סופר אז בית זה הוא '1' וכאשר הוא עוצר אז בית זה הינו '0'.

פתרונות הסימולציה:

ביצענו בדיקות תרחישי קצה שונים על מנת לאמת את תקינות המערכת:

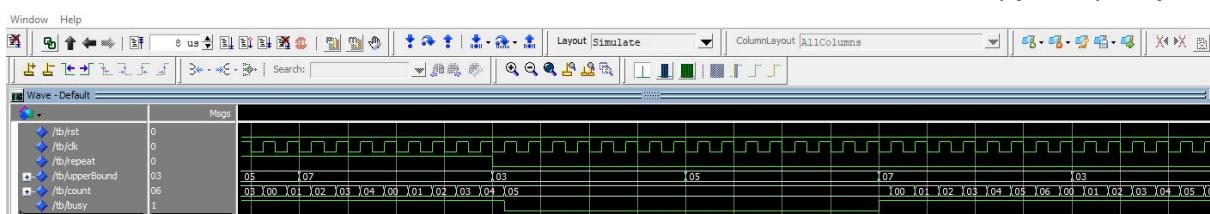
חלק א':



ניתן להבחן כי בתחילת הפעולה בית ה- `rst` דלוק, ולכן מוצא המערכת הינו 1. ה `Upperbound` מוגדר בהתחלה להיות 03 ולכן ברגע שבית ה- `rst` מתאפס המערכת מתחילה במניה. לאחר זמן ה- `phUpperbound` עולה ב-2 ל- 05 אך זה לא משפיע על המניה כי היא עוד לא הגיעו ל- 03. נשים לב כי בית ה`RST` עולה ולכן מוצא המערכת מתאפס שוב. לאחר מכן ברגע שה- `rst` מתאפס המערכת מתחילה לספור שוב וה- `Upperbound` עולה ב-2 ל- 07.

מקרה קיצוני: ברגע מסוים החסם העליון יורד ל03 אבל המונה האיטי נמצא כבר על 04 (ניתן להבחין זאת מכיוון שהמזהה של המערכת שהוא המזהה המקורי כבר ספער עד ל- 03 וכעת אמרור לספור עד 04) בגלל שהחסם ירד אז המונה המהיר ממשיר לספור עד שיגיע למונה האיטי שכרגע עומד על 04 ולאחר מכן תעצר עד שידלוק בית ה `repeat` ולכן המערכת תתחילה את הספירה מחדש מכיוון שהוא עברה את החסם העליון הנוכחי שלה.

חלק ב' (המשך):



כעת המערכת סופרת והחסם העליון שלה הוא 05 ולאחר מכן הוא 07. וברגע מסוים הוא יורד ל- 03 אך כמו מקודם בגלל שהמונה האיטי עומד על 05 אז המונה המהיר (המוחזק) ימשיר לספור עד שיגיע ל- 05 ואז המערכת תפסיק לעבוד ובית ה- `BUSY` יתאפשר. עד שהחסם העליון יעלה ויגיע ל- 07 והמערכת תמשיך לספור.