**­**

***ארכיטקטורת יחידת עיבוד מרכזית***

***מגישים: אייר גלעד 300309937***

***038109484 דודו אבו***

**Content:**

[1. הערות לעבודה 3](#_Toc245477475)

[2.ALU Unit 3](#_Toc245477476)

[3. Shift Unit 7](#_Toc245477477)

[4. Output selector Unit 9](#_Toc245477475)

[5. Arithmetic Unit 11](#_Toc245477476)

[6. Add Sub Unit 14](#_Toc245477477)

[7. Mul Unit 17](#_Toc245477475)

[8.Absolute Unit 19](#_Toc245477476)

[9. Mac Unit 21](#_Toc245477477)

[10.Max/Min Unit 26](#_Toc245477475)

[11. ALU+Communication Unit (Top level) 26](#_Toc245477475)

[12. Transmitter Unit 27](#_Toc245477476)

[13. Reciver Unit 29](#_Toc245477477)

[14. Seven segment decoder Unit 30](#_Toc245477475)

[15.Signal tap analysis 13](#_Toc245477476)

[16. Conclusions 33](#_Toc245477477)

Part 1 : ***Performance Test Case***

הערות:

בדיקות הביצועים נעשתה על פי הדברים הבאים:

1. כל מודול חובר לרגיסטר כניסה ורגיסטר יציאה למען סנכרון המודול עם clock כאשר שעון הרפרנס שלנו יהיה בתדירות של 50 mhz.

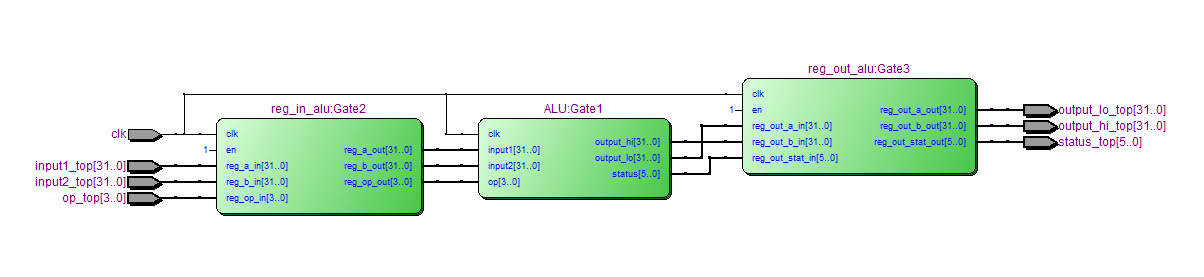
2. כלל הבדיקות נעשו עבור כניסות למודולים של קלט באורך 32 ביט

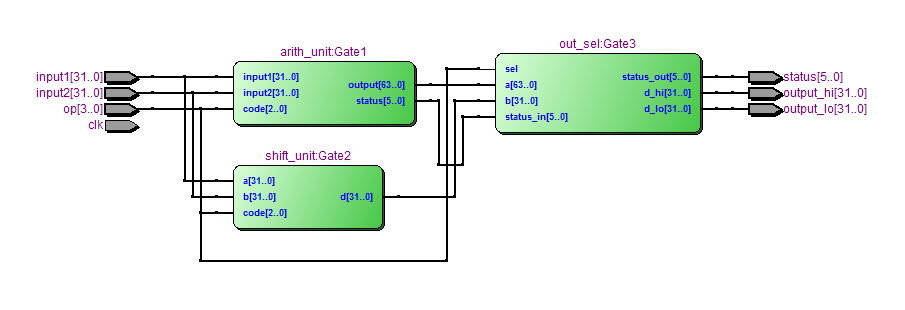
3.עבור כל מודול נמצא את המסלול הקריטי,שעבורו נקבע התדר המקסימלי שהמודול יכול לעבוד.

4. נמצא את התדר המקסימלי עבורו המודול יכול לעבוד, בהתחשב בהגבלה מהמסלול הקריטי.

5.עבור מודולים אשר ניתן לראות את rtl view בצורה ברורה, נציג גם אותו.

**1.ALU**

במעבדה זו נדרשנו לתכנן ולצרוב רכיב ALU- Arithmetic Logic Unitהמבצעת פעולות מסוימות(ראה טבלה 1.2) על הקלט אותה היא מקבלת, ובהתאם לקוד פעולה אותה היא מקבלת גם כן. בסיום ביצוע הפעולה היא מוציא את תוצאת הפעולה כפלט, וסטטוס במקרה מסוים(יפורט בהמשך).

איור 1.1 חיבור רגיסטרים ל- ALU

איור 1.2 ALU המורכב מ-3 מודולים שונים.

1.1 ALU port table

|  |  |  |  |
| --- | --- | --- | --- |
| Port description | Port length | Port direction  - in/out | *Port name* |
| Input logical vector | N | in | *A* |
| Input logical vector | N | in | *B* |
| Code for operation to be executed(see table 1) | 4 | in | *Op Code* |
| Present the N msb bits of the result of the operation executed on A and B. | N | out | *Hi* |
| Present the N Lsb bits of the result of the operation executed on A and B.  \* Some operation use **only** LO for output in this case HI is irrelevant | N | out | *LI* |
| register presents the status for the sub operation(#1) only, (see table 4.1)  \*on other operation status output is irrelevant | 6 | out | *status* |

**Table 1.1: ALU port map**

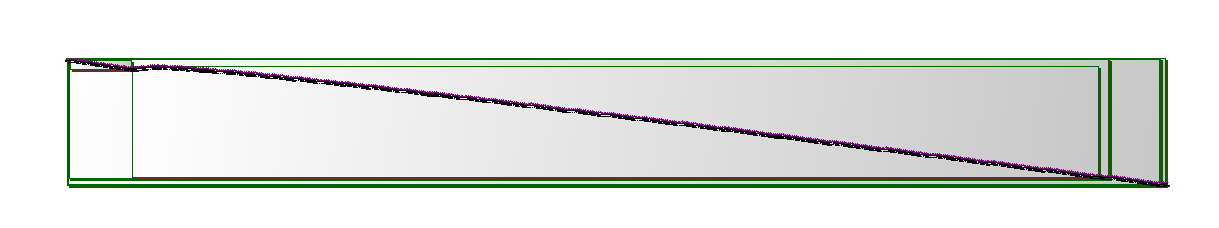
1.2 ALU available operations and codes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Op code | operation | RTN | Description |
| 0 | 0000 | ADD | LO = A + B | Arithmetic Add |
| 1 | 0001 | SUB | LO = A - B | Arithmetic Sub |
| 2 | 0010 | MIN | LO=Min(A,B) | Return minimum between A and B |
| 3 | 0011 | MAX | LO=Max(A,B) | Return maximum between A and B |
| 4 | 0100 | ABS | LO=Abs(B) | Return absolute value of B |
| 5 | 0101 | RST | MAC=0 | Reset MAC |
| 6 | 0110 | MAC | MAC= MAC+A\*B  HI,LO)=MAC) | Multiply Accumulate (MAC is internal N\*2 bits  register) signed numbers |
| 7 | 0111 | MUL | (HI,LO) = (A\*B) | Multiply two signed numbers (Result is N\*2 bits register) |
| 8 | 1000 | SHL | LO = A << B | Shift left register A, B times |
| 9 | 1001 | SHR | LO = A>>B | Shift right register A, B times |

**Table 1.2: ALU Op Codes and operation**

Note: Other op codes available for input (4 bits) will consider as nop- no operation.

1.3 ALU synthesis

1.4.1 critical path

איור 1.3 מסלול קריטי של ALU, מבט על

היחידה האריתמיטית ובתוכה יחידת ה mac היא המסלול הקריטי ומגבילה את תדר העבודה של ה- ALU.

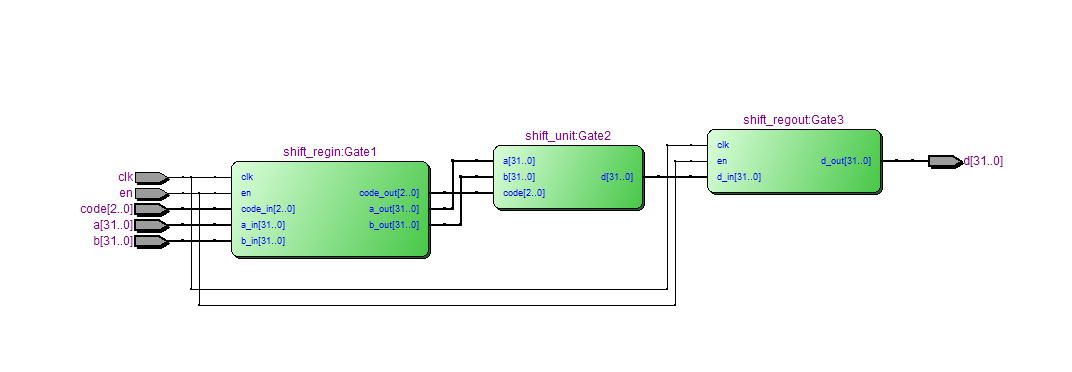
אם נניח כי אנו לא משתמשים הרבה ברכיב המחלק, נוכל לקבוע את תדר השעון להיות מוגבל לפי הרכיב הבא שמגביל אותו, ובמקרה שנקבל פעולת חילוק, נשמש בכמה מחזורי שעון עם bubbling, לביצוע הפעולה.

בהנחה כי כן נשתמש הרבה בפעולת mac, נוכל ליצור רכיב נפרד אשר הוא יבצע את פעולת ה- mac ללא כל קשר ל-ALU המקורי, ובזמן זה יוכל ה- ALU לבצע פעולות נוספות.

1.4.2 max freq

Maximum frequency possible (slow model): 87.11 MHz

גם כאן ניתן לראות כי התדר המקסימלי שיכול לעבוד ה- ALU הוא בקרוב זה של יחידת ה- mac ושל היחידה האריתמטית המוגבלים שניהם ע"י ה- mac, בהאטה מסויימת שכן המסלול שיש לעבור ב-

ALU דורש מעבר דרך מודולים נוספים כגון output selector.

|  |
| --- |
| **2.shift unit**  This unit makes shifting arithmetic to input A number of times as in input b(the significance of be is an integer with unsigned binary presentation), the shifting can be done on right or left direction according to code signal. This unit has a relevant output only for shift op code on if it gets code numbers #(10-15) it does none.    איור 2.1 shift unit מחובר בין רגיסטרים.    איור 2.2 shift unit – סינטזה לוגית |

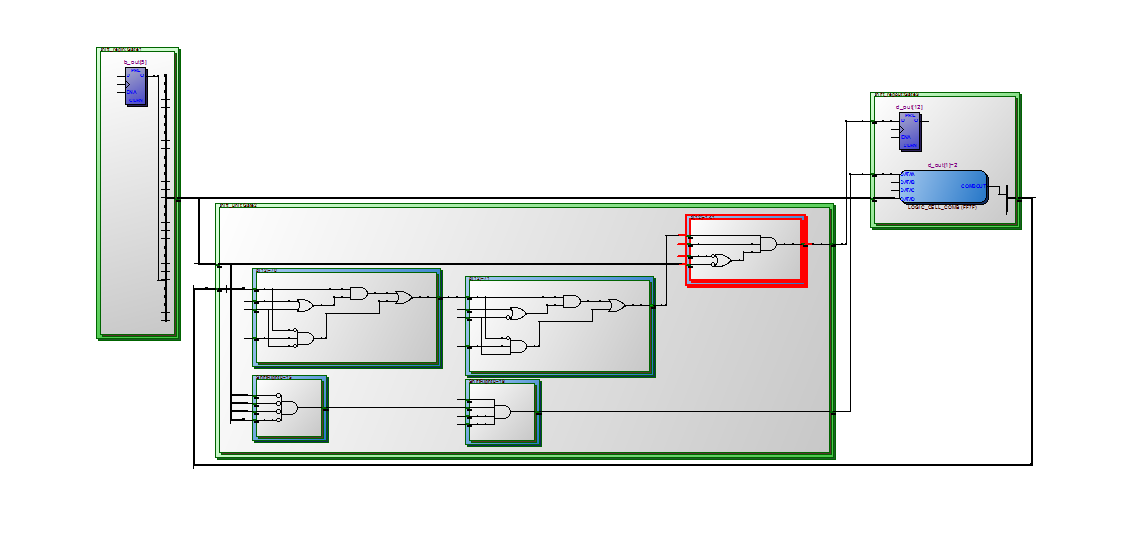
|  |  |  |  |
| --- | --- | --- | --- |
| Port description | Port length | Port direction  - in/out | Port name |
| Input logical vector | N | in | A |
| Input logical vector | N | in | B |
| Selects between operation(ssl/ssr/nop) | 3 | in | Code |
| The result of the executed operation | N | out | Output |

2.1 shift unit port table

**Table 2.1: shift unit port table**

2.2 shift unit synthesis

2.2.1 Critical path

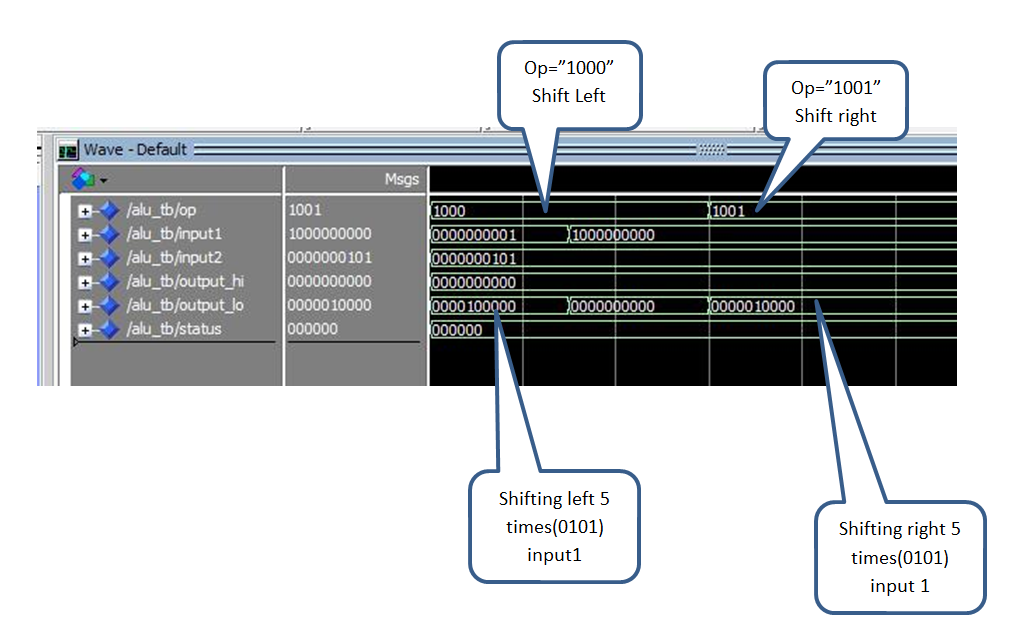


איור 2.3 מסלול קריטי shift unit

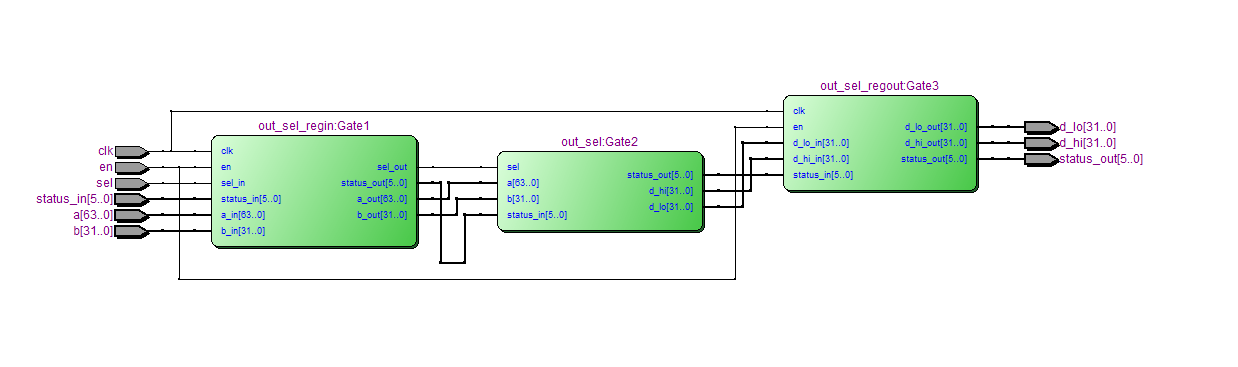
2.2.2 max freq:

Maximum frequency possible (slow model): 105.46 mhz

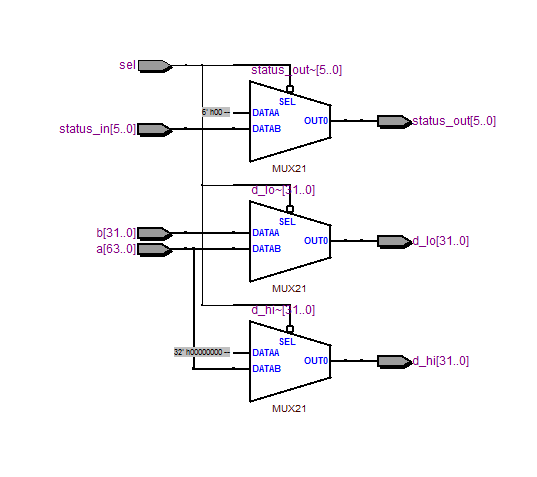
יחידה זו, יכולה לעבור בתדר גבוה יחסית לשעון הרפרנס, ולכן היא אינה מגבילה את פעולת ה- ALU.

2.3 shift unit model-sim simulation

**3.Output selctor**

This unit selects with unit in the ALU has a relevant output for the entire system. If it’s the arithmetic unit, it takes is status output also, if it’s the shit unit the status output will be “0”.if selection signal is “0” the output will be from the arithmetic unit otherwise it will be from the arithmetic unit. This output will go directly to the ALU output ports. This unit is work similar to multiplexer.

איור 1.3 output selector מחובר בין רגיסטרים.



איור 2.3 output selector - סינטזה לוגית

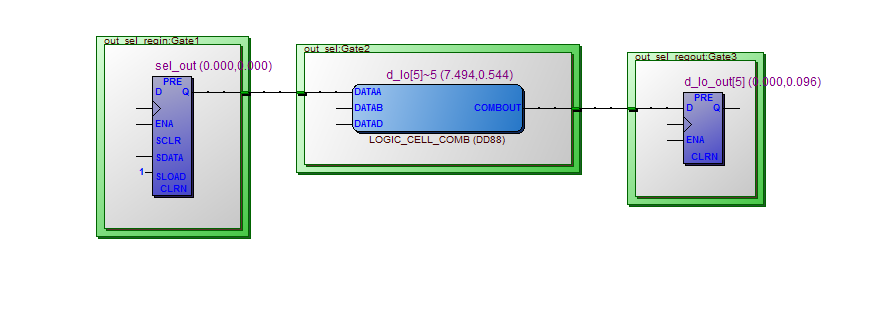
3.1 output selector port table

|  |  |  |  |
| --- | --- | --- | --- |
| Port description | Port length | Port direction  - in/out | Port name |
| Input logical vector from arithmetic unit | N2 | In | A |
| Input logical vector | N | In | B |
| Status input dorm arithmetic unit | 6 | In | Status in |
| Selecting between units | 1 | In | Sel |
| Relevant status to be out of entire system | 6 | out | Status out |
| Output for HI register from the entire system | N | out | Out hi |
| Output for LO register from the entire system | N | out | Out lo |

**Table 3.1: output selector port map**

3.2 output selector synthesis

3.3.1 critical path



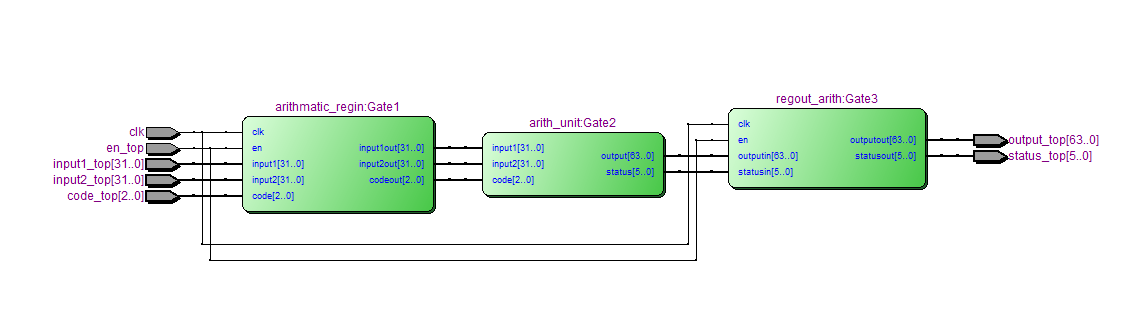
איור 3.3 output selector - מסלול קריטי

3.3.2 max freq

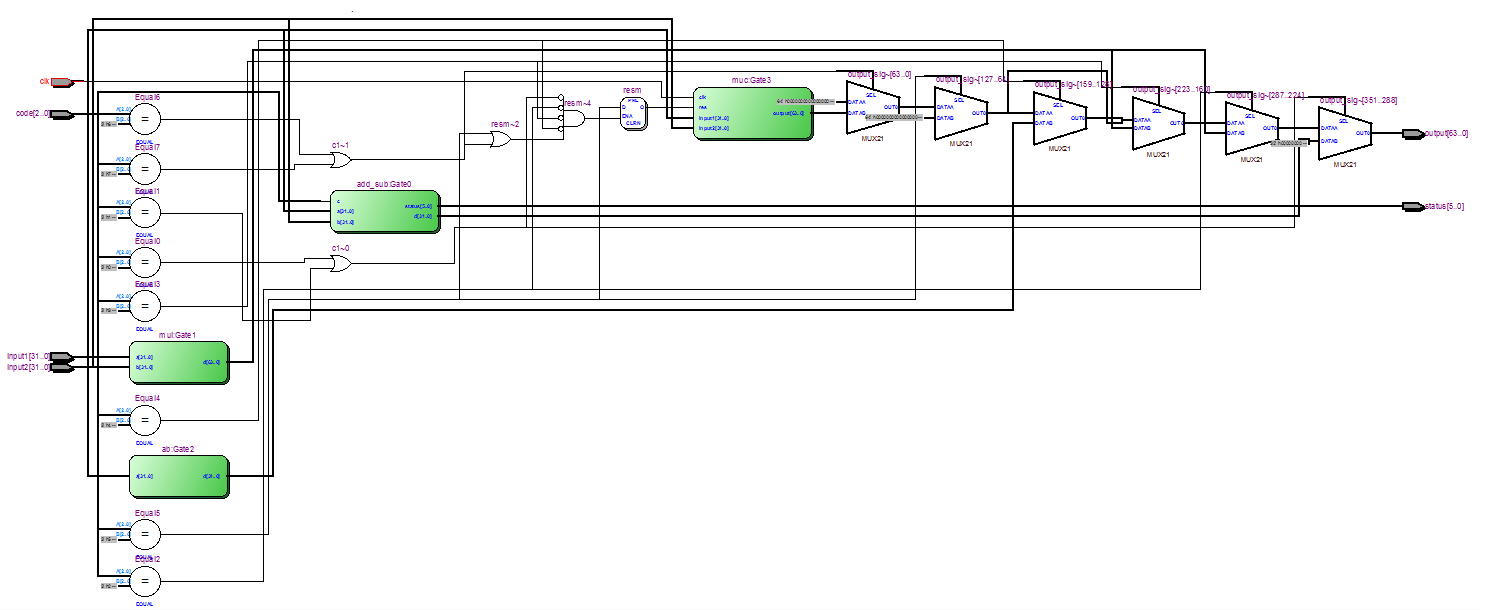
maximum frequency possible (slow model): 119.39 mhz

**4.Artihmethic unit**

The arithmetic unit performs all the arithmetic operation available in the ALU, as see in table (1.2) the operation codes are numbers # (0-7).



איור 4.1 arithmethic unit - יחידה אריתמטית בין רגיסטרים



איור 2.4 arithmethic unit - סינטזה לוגית

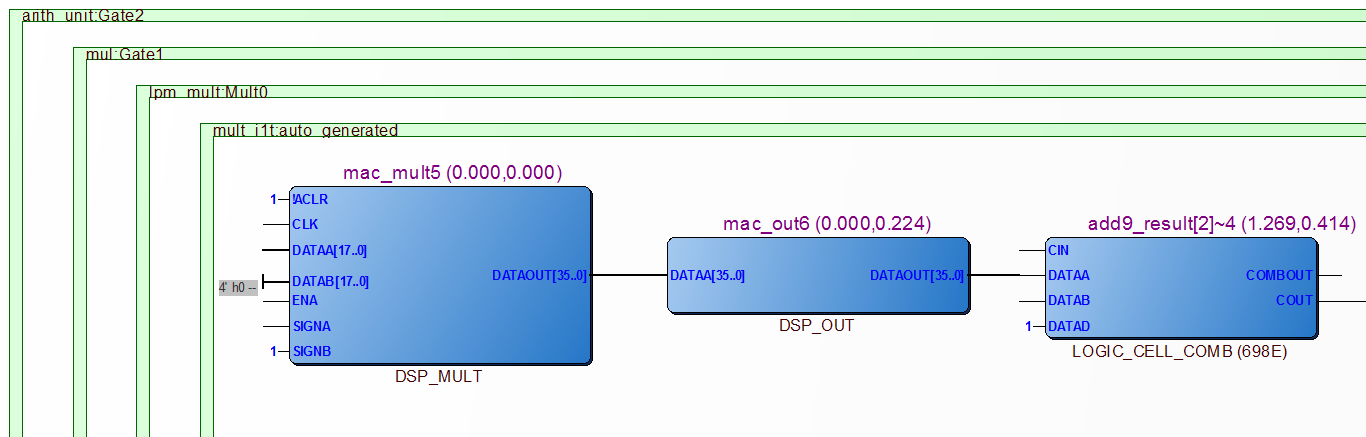
4.1 Arithmetic unit port table

|  |  |  |  |
| --- | --- | --- | --- |
| Port description | Port length | Port direction  - in/out | Port name |
| Input logical vector | N | in | Input1 |
| Input logical vector | N | in | Input2 |
| Code for operation to be executed | 3 | in | Op Code |
| Present the result of the operation executed on A and B. | N2 | out | output |
| register presents the status for the sub operation(#1) only, (see table 4.1) \*on other operation status is irrelevant | 6 | out | status |

**Table 4.1: Arithmetic unit port map**

4.2 arithmethic unit synthesis

4.2.1 critical path



איור 3.4 arithmethic unit - מסלול קריטי

גם כאן ניתן לראות כי ה mac הוא זה המגביל את תדר השעון של היחידה האריטמתית שכן המסלול הקריטי עובר דרכו, ולכן תדר המקסימום שהיחידה האריטמתית יכולה לפעול הנו תדר המחלק מוקטן מעט שכן המסלול הוא ארוך יותר.

4.2.2 max freq

maximum frequency possible (slow model): 88.42 MHz

גם כאן ניתן לראות כי מה שמגביל את היחידה האריתמטית הוא יחידת ה mac, אשר מגבילה גם את ה- ALU עצמו כפי שראינו לעיל.

**5.Adder/subber unit**

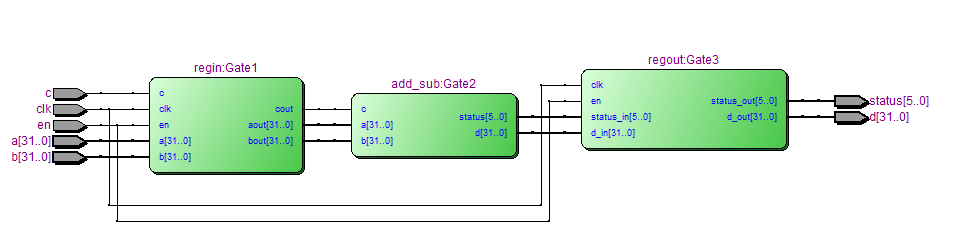
The add/sub makes adding( a+b ) or subtraction(a-b) between to input logical vector its gets (a, b). The selecting between the operations needed is by input signal called sel.

This unit also responsible for the status output for the all system which is relevant only on subtraction operation(#1).

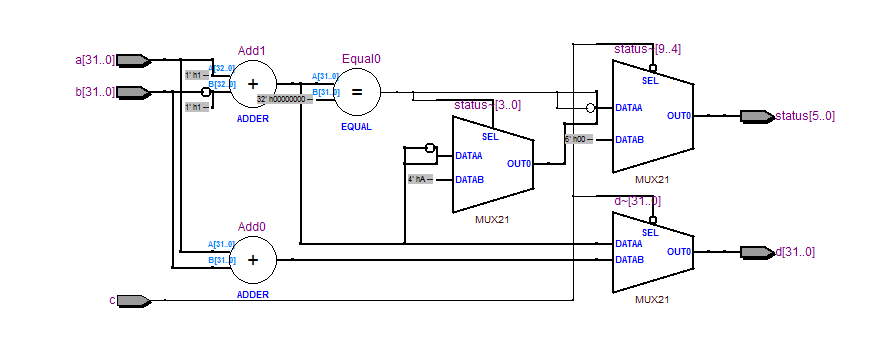
The status output contains 6 bits(0-5), when ‘0’ is the lsb, and ‘5’ is the msb. Each bits has it significance, when a specific bit has value of”1” that means the condition fits to this bit is true. The conditions and the output bits relation showed is table 5.1

|  |  |  |
| --- | --- | --- |
| Bit number | Condition name | description |
| 0 | Eq | A=B |
| 1 | Ne | A!=B |
| 2 | Ge | A>=B |
| 3 | Gt | A>B |
| 4 | Le | A<=B |
| 5 | Lt | A<B |

**Table 5.1 status output bits relative to conditions**



איור 5.1 adder/subber between registers



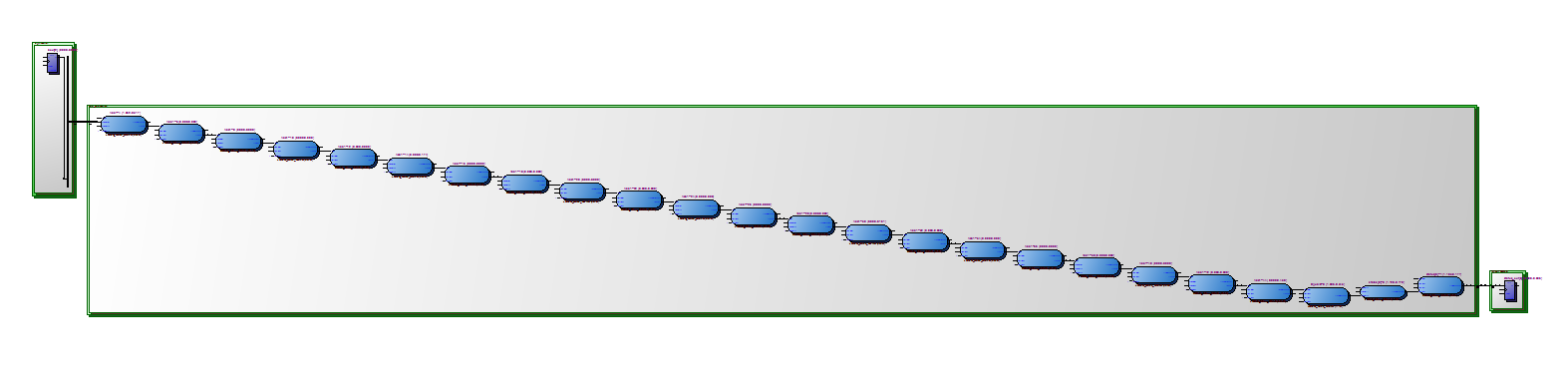
איור 5.2 adder/subber - סינטיזה לוגית

5.1 add/sub port table.

|  |  |  |  |
| --- | --- | --- | --- |
| Port description | Port length | Port direction  - in/out | Port name |
| Input logical vector | N | in | A |
| Input logical vector | N | in | B |
| Selects between add or sub operation | 1 | in | Sel |
| The result of the executed operation | N | out | Output |
| register presents the status for the sub operation(#1) only, (see table 4.1) \*on other operation status is irrelevant | 6 | out | Status |

**Table 5.2 add/sub unit port map**

5.2 add/sub synthesis.

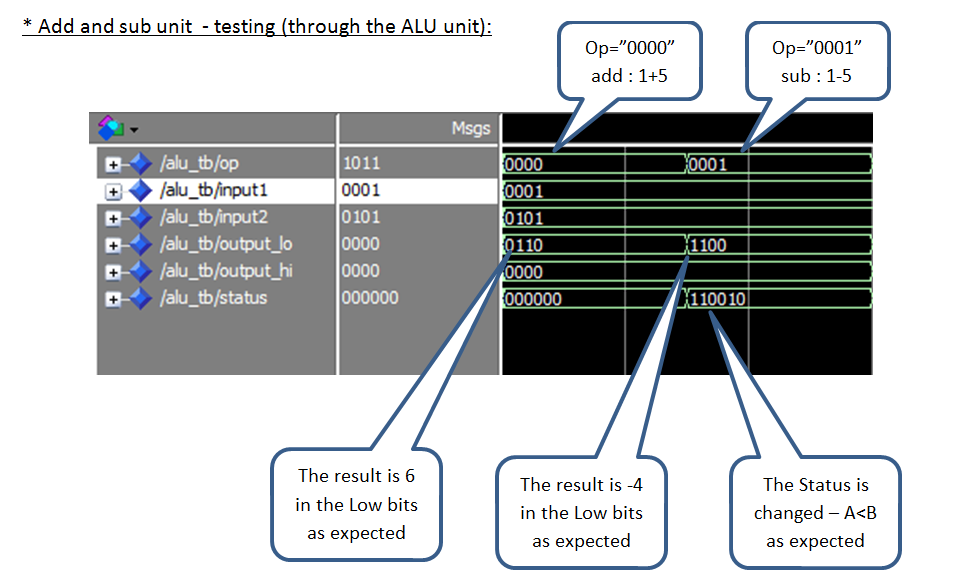
 5.2.1 critcal path

איור 5.3 adder/subber - מסלול קריטי

5.2.2 max freq

maximum frequency possible (slow model): 109.12 mhz

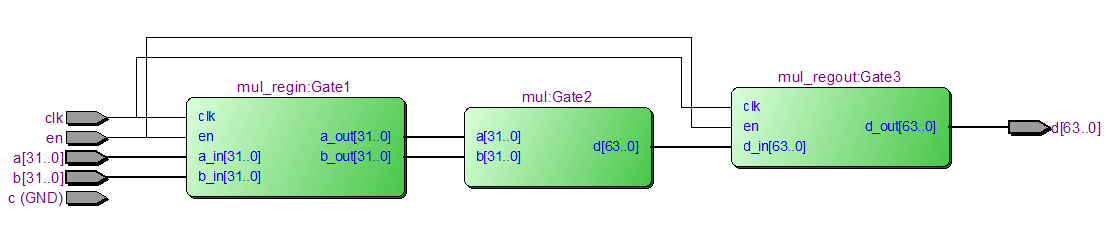
5.3 model-sim simulation



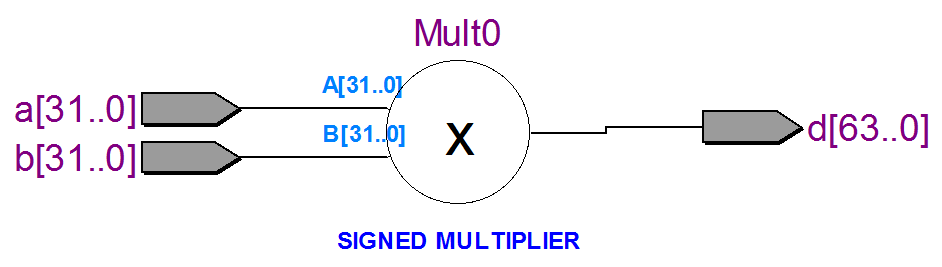
**6.Mul**

The mul makes multiply between to input logical vector its gets (a, b). The multiply operation is signed.

Overflow would be ignored in signed operation.



איור 6.1 mul/umul - בין רגיסטרים



איור 6.2 mul - סינטזה לוגית

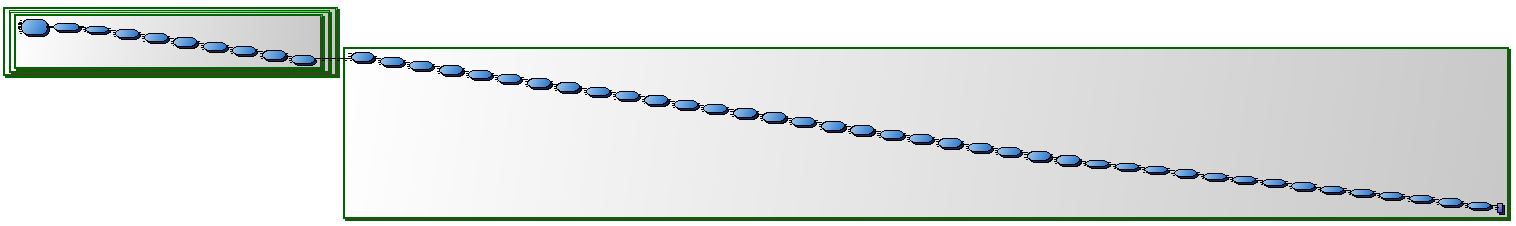
6.1 mul port table

|  |  |  |  |
| --- | --- | --- | --- |
| Port description | Port length | Port direction  - in/out | Port name |
| Input logical vector | N | in | A |
| Input logical vector | N | in | B |
| The result of the executed operation | 2N | out | output |

**Table 6.1 mul unit port map**

6.2 mul synthesis.

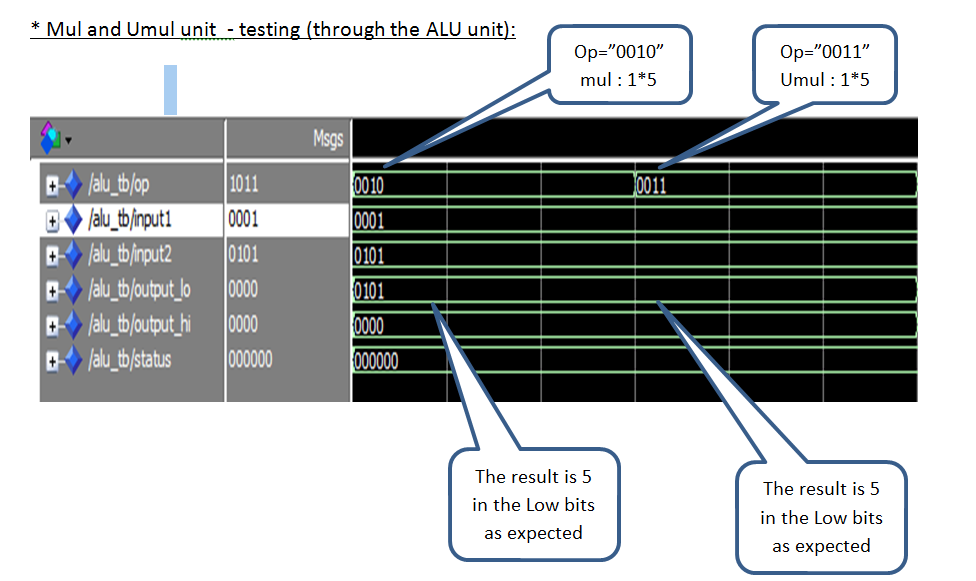
6.2.1 critical path



איור 6.3 mul - מסלול קריטי

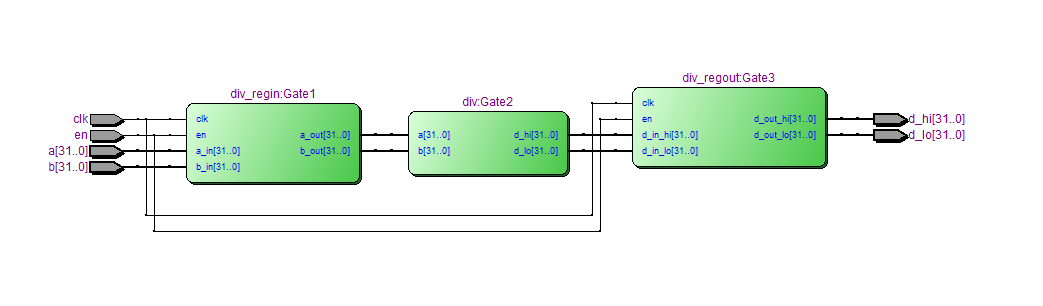
6.2.2 max freq

maximum frequency possible (slow model): 98.13 MHz

6.3 model-sim simulation

**7.Absolute unit**

Gives the Absolute value of the signed input vector(N) . The output is 1 output vector.



איור 7.1 absolute unit - בין רגיסטרים

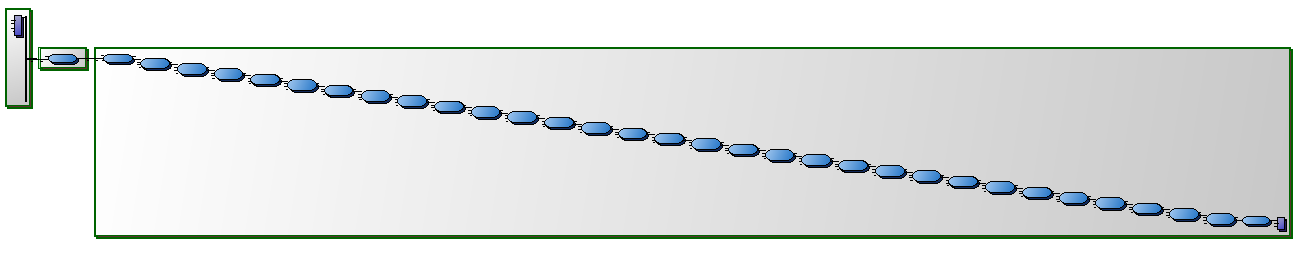
7.1 absolute port table

|  |  |  |  |
| --- | --- | --- | --- |
| Port description | Port length | Port direction  - in/out | Port name |
| Input logical vector | N | in | B |
| Register represents the reminder of the division Integer part of the division | N | out | D lo |

**Table 7.1 absolute unit port map**

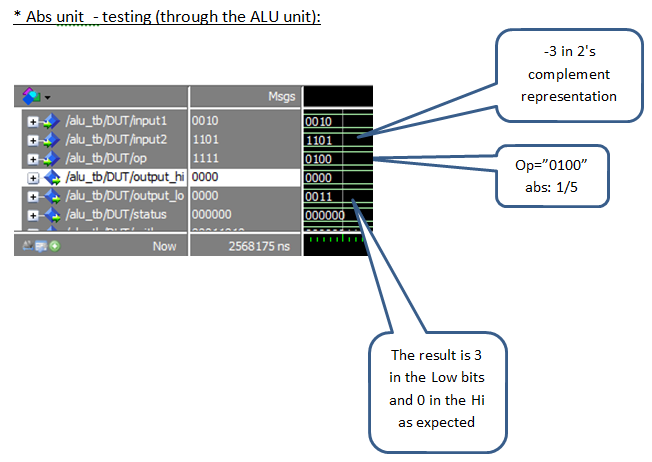
7.2 absolute unit synthesis.

7.2.1 critical path:



7.2.2 max freq:

maximum frequency possible (slow model): 212.59 MHz

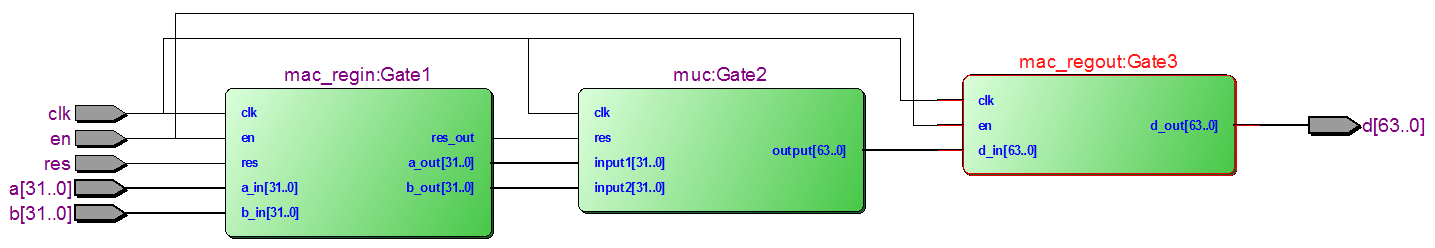
7.3 model-sim simulation

**8.Mac**

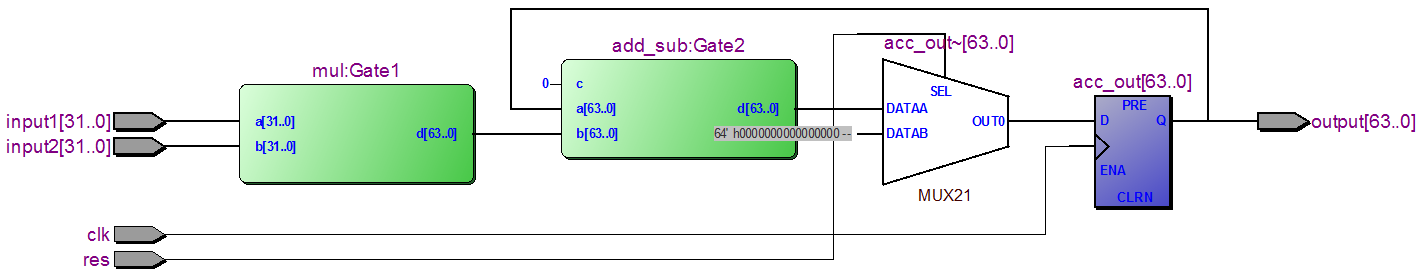
This unit has its internal register called mac(length N), doing the operation between the inputs (a,b) which is mac+a\*b ( signed multiplication). Mac is initialized as logical vector equal to “0”.

This unit is synchronic by 1/10 ns clock(rising edge). Allowing multiply and accumulation for same input. As different from other opcodes the mac/umac opcode needs the possibility to sum the same inputs with mac register,(loops for example), and that’s why a clock is needed for this unit design.

This unit contains 3 other sub-units (add/sub, mul which detailed above),and **mac clock unit** especially for this unit only(8). The unit also knows to reset the mac to “0” no matter which value in contains.



איור 8.1 mac - בין רגיסטרים



איור 8.2 mac - סינטזה לוגית

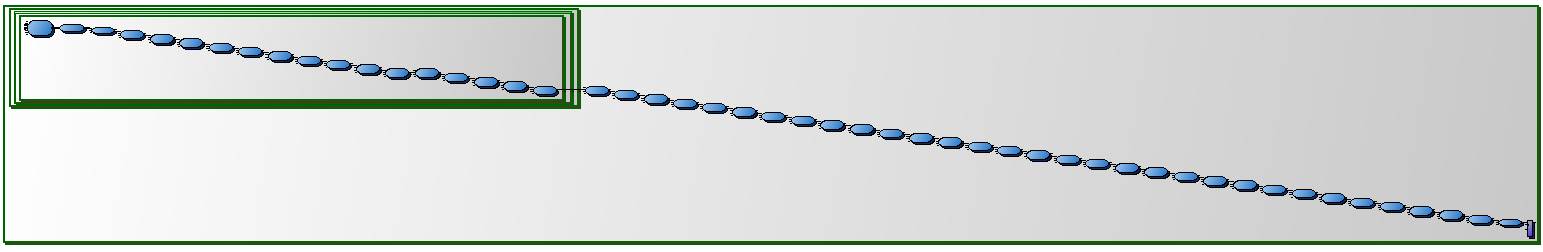
8.1 mac port table

|  |  |  |  |
| --- | --- | --- | --- |
| Port description | Port length | Port direction  - in/out | Port name |
| Input logical vector | N | in | A |
| Input logical vector | N | in | B |
| Reset signal for mac register | 1 | in | Res |
| Clk enters for mac/umac clock unit, the output value is according to current input upon clock rising . | 1 | in | Clk |
| The result of the executed operation | 2N | out | Output |

**Table 8.1: mac port map**

8.2 mac unit synthesis.

8.2.1 critical path:



איור 8.3 mac - מסלול קריטי

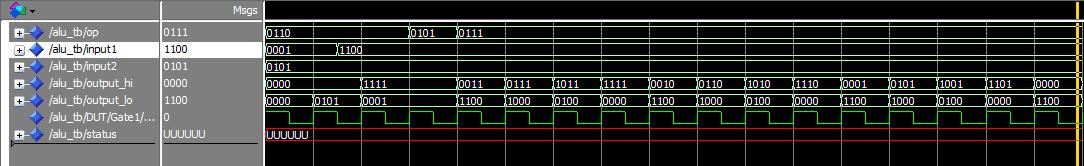
8.2.2 max freq

maximum frequency possible (slow model): 87.11 MHz

כפי שניתן לראות, המסלול הקריטי ביחידה זו האורך ביותר. יחידה זו הנה נקודת התורפה מבחינת הקצב בו יכול לעבור ה-ALU שלנו, ולכן עלינו לבדוק את תדירות שימוש ביחידה זו כמפורט לעיל.

8.3 model-sim simulation

Op=”0110” mac



Clock signal, output in respect to rising edge of clock

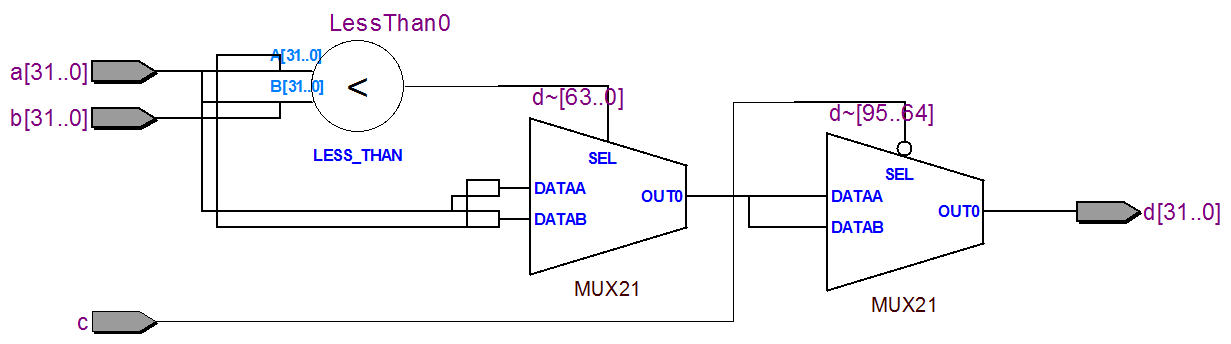
Reset for the mac register, mac=0

The result of mac, 5+(-4)\*5= -15 as excepted

Output of mac Operation when clock rise Result is 5

**9.Max/Min Unit**

Similar to the Add/Sub unit the Max/Min unit takes two signed input vectors and gives the minimum or maximum of them according to the Sel input**.**



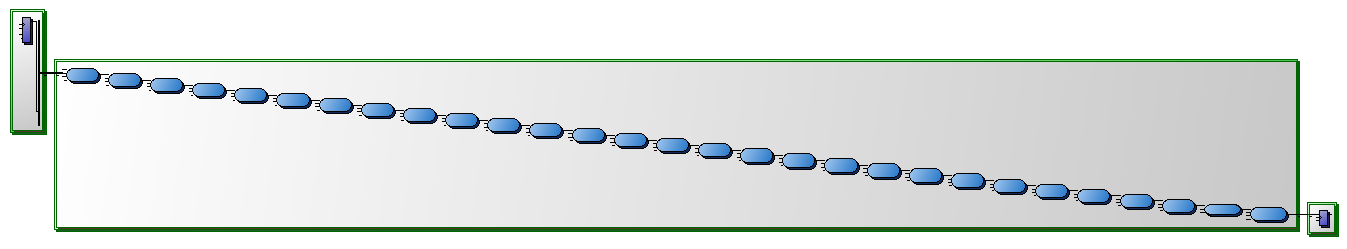
איור 9.1 – Max/Min סינטזה לוגית

9.1 Max/Min port table

|  |  |  |  |
| --- | --- | --- | --- |
| Port description | Port length | Port direction  - in/out | Port name |
| Input logical vector | N | in | A |
| Input logical vector | N | in | B |
| Selects between Max or Min operation | 1 | in | Sel |
| The result of the executed operation | N | out | Output |

**Table 9.1 max/min unit port map**

9.2 max/min unit synthesis.



איור 9.1 – Max/Min מסלול קריטי

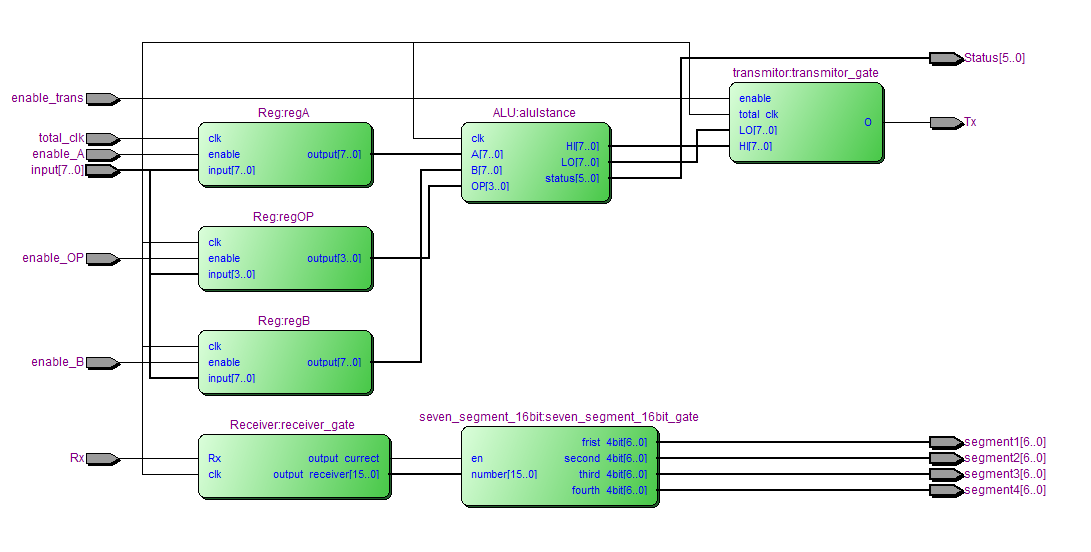
9.2 max/min max frequency possible (slow model): 159.85 MHz

Part 2 : ***Hardware Test Case***

בחלק זה אנו נבדוק ALU של 8 bit בלוח ה FPGA כאשר עבור התקשורת נשתמש בפרוטוקול RS232 .

נעבור על החלקים העיקריים של המודול שנוספו ובדיקה של הtop level entity.

**Lab2– Top Level Entity**

פועל על פי הסכמה הבא:

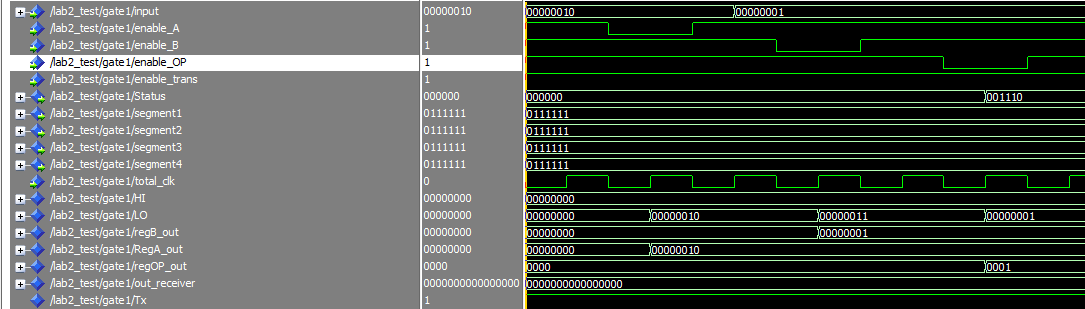
איור 5 Lab2- Entity

ניתן לראות כי המידע הנכנס דרך המצגים עובר נטען לכל רגיסטר כאשר ניתן לו enable, לאחר מכאן מתבצע החישוב בALU. כאשר ניתן enable ליחידת ה transmitter הוא משדר את המידע באמצעות פרוטוקול RS232. בנוסף קיים רכיב אשר קולט אות באמצעות הפרוטוקול ומציג את התוצאה בצג הדיגיטלי של הseven segment.

|  |  |  |  |
| --- | --- | --- | --- |
| **שם** | **גודל** | **כיוון** | **שימוש** |
| enable\_trans | 1 | IN | Enable for the transmiter |
| enable\_OP | 1 | Enable for the Opcode |
| enable\_A | 1 | Enable for first input to ALU |
| enable\_B | 1 | Enable for second input to ALU |
| Total\_clk | 1 | IN | clock |
| input | 8 | Input from the switches |
| Rx | 1 | Receiver input |
| STATUS | 6 | OUT | מוצא של ה status לLED |
| Tx | 1 | Transmitter output |
| Segment1 | 7 | Digital 7-segment |
| Segment2 | 7 |
| Segment3 | 7 |
| Segment4 | 7 |

**טבלה מס' 12- Port Table lab2**

תוצאות הבדיקה בModelSim וניתוחם:עבור lab2:



טעינת הערך לרגיסטר A

טעינת הערך לרגיסטר B

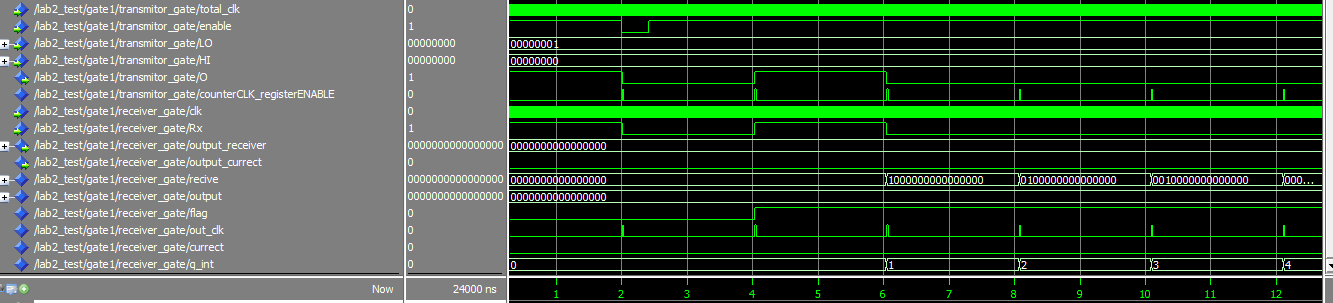
טעינת הערך של ה opcode

התייצבות הערך של HI,LO לאחר שכל הinput –ים התייצבו

**איור 5.1 קובץ בדיקה lab2**

מתקבלים enable בtransmitter לאחר מכאן נשלח start bit והreceiver מקבל את האות ומתחיל לקרוא 16 bit

לאחר הstart bit הreceiver מתחיל לקרוא אות כל clock

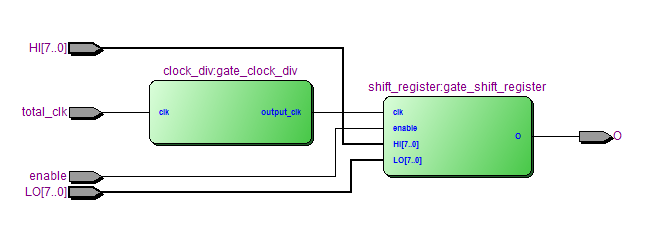


**איור 5.1 קובץ בדיקה lab2**

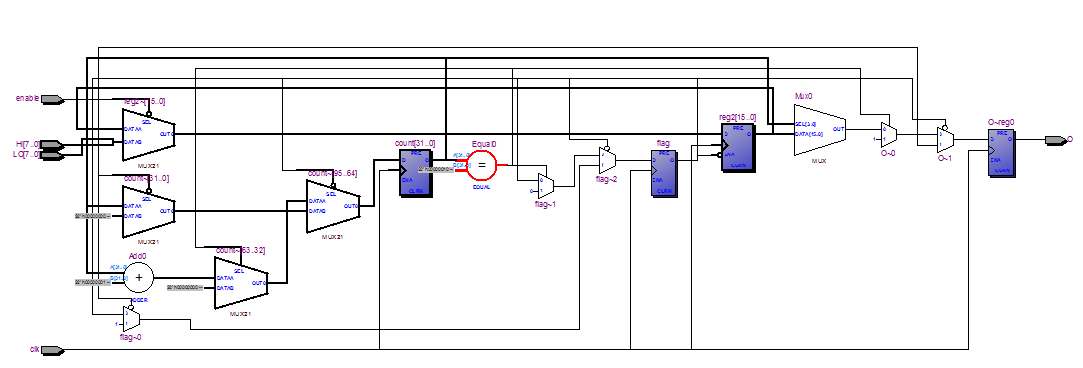
**Transmitter**

היחידה האחראית על שליחת המוצא בתדירות 0.5MHz לReceiver.

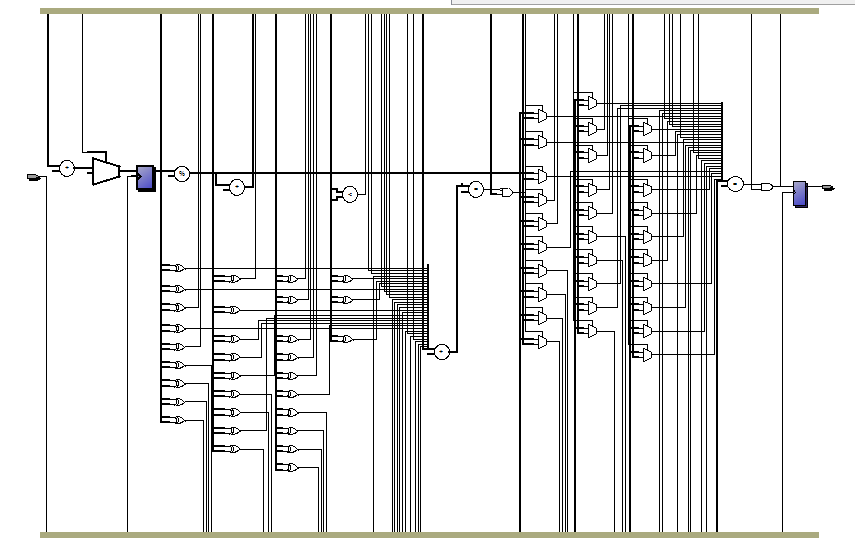
* היחידה מקבלת שתי אותות, כל אחד 8 ביט, אשר מהווים את תוצאת הפעולה המתמטית ומשרשת אותם ליצירת אות 16 ביט.
* בלחיצה על כפתור Key4 היחידה שולחת מPin\_H17 ביט תחילת שידור '0' לאחריו אות 16 ביט לבסוף שולחת ביט סיום '1', בתדירות 0.5MHz.



**איור 6 Transmitter**

****

**איור 6.1 Shift Register המשומש בTransmitter**

****

**איור 6.2 מחלק השעון- 50MHz-0.5MHz**

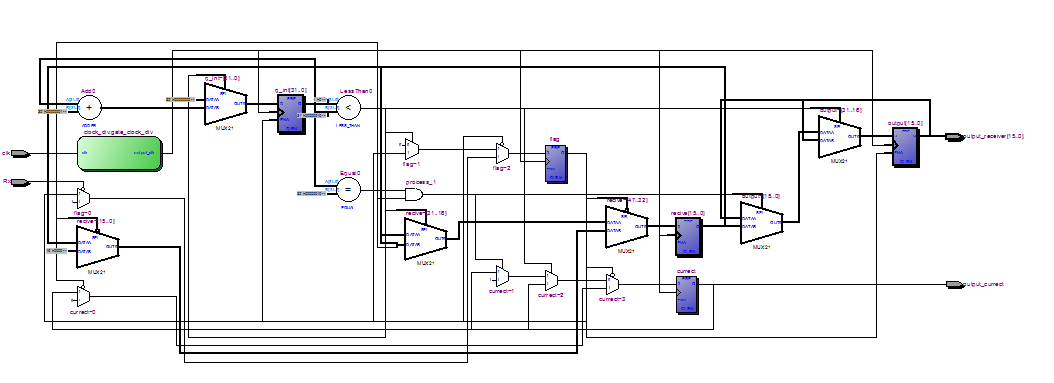
|  |  |  |  |
| --- | --- | --- | --- |
| **שם** | **גודל** | **כיוון** | **שימוש** |
| LO | N | IN | LSB של התוצאה |
| HI | N | IN | MSB של התוצאה |
| CLK | 1 | IN | שעון 50 MHz |
| Enable | 1 | IN | Key 4 |
| O | 1 | OUT | ביט הנשלח מהמערכת |

**טבלה מס' 13- Port Table Transmitter**

**Reciver– Top Level Entity**

היחידה האחראית של לקבלת מידע על פי פרוטוקול RS232 ב 0.5MHz.

* מחכה לקבלת start bit- 0 לוגי.
* קוראת מילה של 16 bit.
* מוציאה אות אשר מחווה על תקינות ה frame.

עובדת על פי הסכימה הבא:

**איור 7 Receiver RTL**

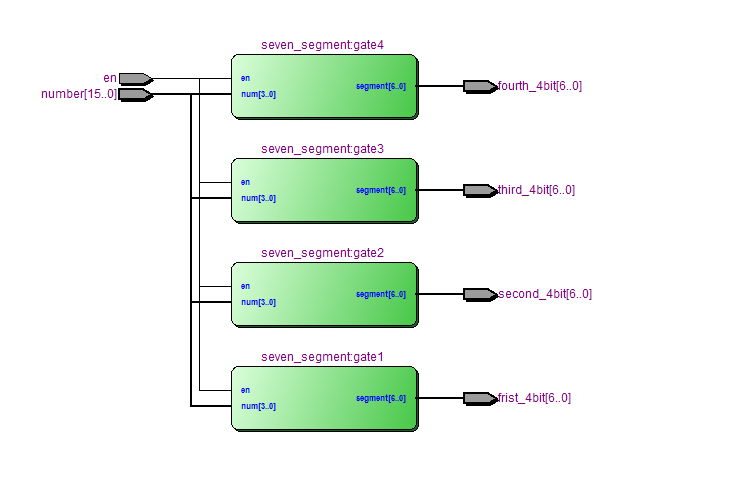
|  |  |  |  |
| --- | --- | --- | --- |
| **שם** | **גודל** | **כיוון** | **שימוש** |
| clk | 1 | IN | clk |
| Rx | 1 | IN | Input |
| Outpt\_receiver | 2\*N | IN | מוצא אשר הולך ל7-segment |
| Output\_currect | 1 | IN | מחווה על תקינות הframe |

**טבלה מס' 14- Port Table receiver**

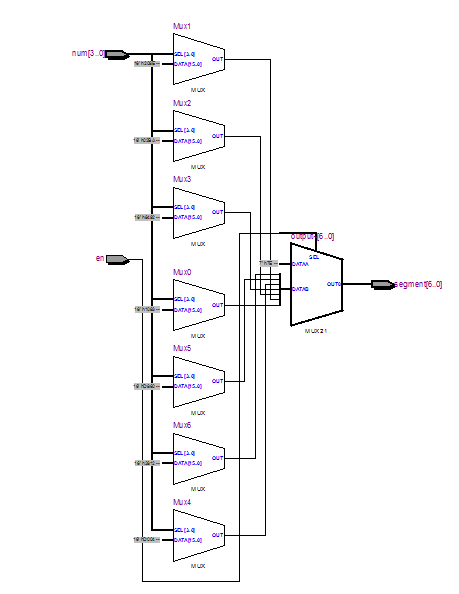
**7 Segment**

היחידה האחראית על הצגת האות שנכנס לReceiver על הצג הדיגיטלי.

* היחידה מקבלת אות באורך 16 ביט ומציגה אותו על הצג הדיגיטלי בHex.



**איור 8- 7 Segment**

****

**איור 8.1 7 Segment - RTL**

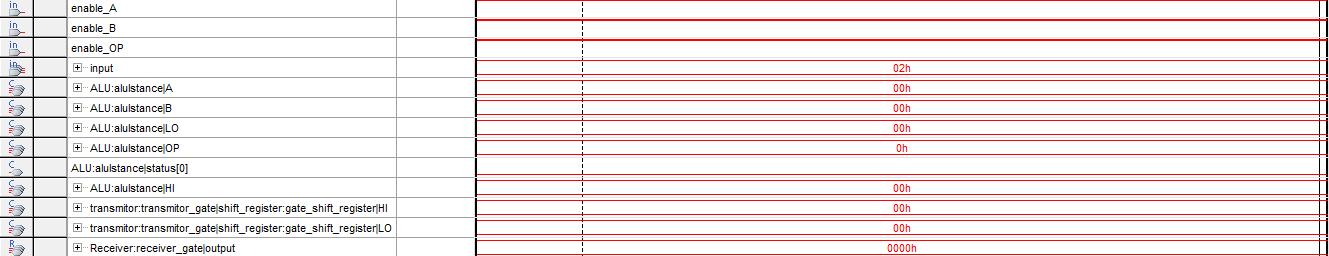
|  |  |  |  |
| --- | --- | --- | --- |
| **שם** | **גודל** | **כיוון** | **שימוש** |
| Number | N2 | IN | התוצאה |
| En | N | IN | אות הפעלה מהRceiver- |
| First 4 Bit | 4 | OUT | סיפרה ראשונה בצג |
| Second 4 Bit | 4 | OUT | סיפרה שניה בצג |
| Third 4 Bit | 4 | OUT | סיפרה שלישית בצג |
| Forth 4 Bit | 4 | OUT | סיפרה רביעית בצג |

**טבלה מס' 15- Port Table 7 Segment**

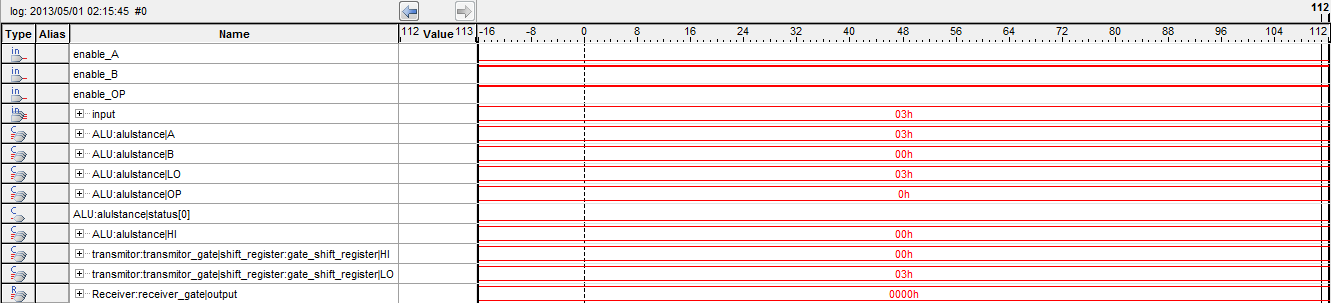
**Signal Tap analysis**

נראה את השלבים הנעשים בתוך הFPGA באמצעות הsignal tap:

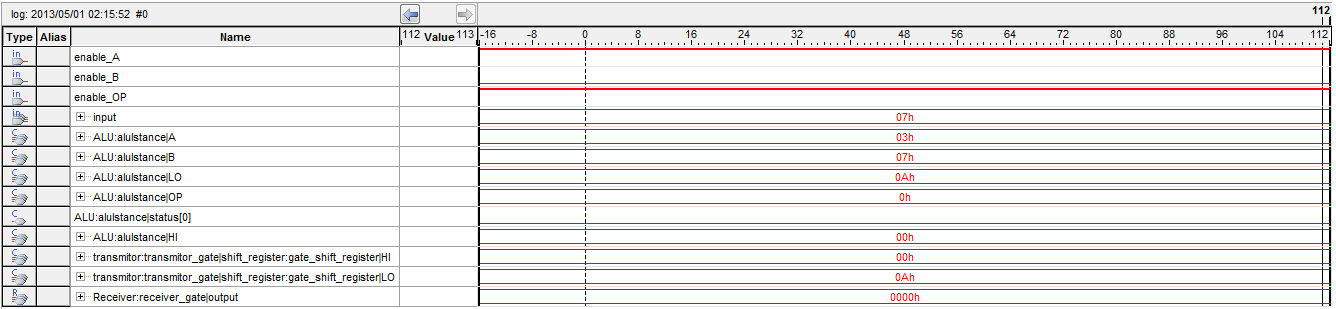
1. קבלת input ראשון :



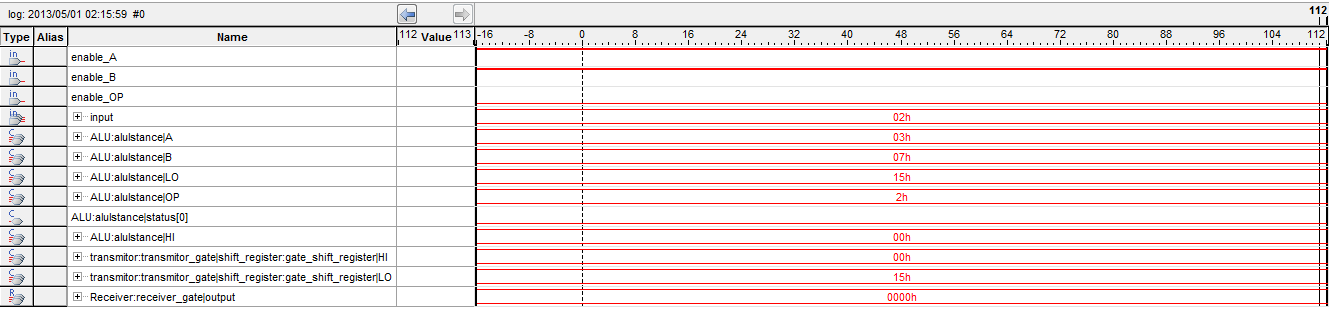
1. השמת הערך ברגיסטר A וקבלת input שני:



1. השמת ערך בתוך רגיסטר B וקבלת input שלישית ניתן לראות פעולת חיבור היות והopcode באפס:



1. השמת ערך ברגיסטר opcode וקבלת הפעולה הרצויה ביציאה מהALU:



1. ניתן לראות את האות משודר ומתקבל בreceiver :

