**Generic list – Internal Logic Analyzer Core**



| Number | Generic Parameter | Description | Type | Default Value |
| --- | --- | --- | --- | --- |
|  | record\_depth\_g | Determine the number of bits that will be recorded for each signal | positive | 256 |
|  | num\_of\_signals\_g | Determine the number of signals that will be recorded | positive | 8 |
|  | reset\_polarity\_g | Reset polarity:  '1': Active high  '0': Active low | std\_logic | '0' |
|  | data\_width\_g | The width of the basic 'word' of wishbone interface | positive | 8 |
|  | add\_width\_g | The address width of wishbone interface | positive | 8 |
|  | single\_ram\_depth\_g | Number of lines in the basic RAM used in the core | positive | 10 |
|  | single\_ram\_width\_g | The width of the basic 'word' of the basic RAM used in the core | positive | 8 |



Pin table of core parts:

|  |  |  |  |
| --- | --- | --- | --- |
| Description | Width | Direction | Pin Name |
| Clock | 1 | In | Clk |
| Synchronous reset signal | 1 | In | Reset |
| Signal used for capturing data | 1 | In | trigger |
| The input signals which will be recorded | Num\_of\_signals\_g | in | input\_data |
|  |  |  | WBS |
|  |  |  | WBM |

|  |  |
| --- | --- |
| ~~Size(bit)~~ | ~~outputs~~ |
| ~~?~~ | ~~Wbm~~ |
| ~~[0..7]~~ | ~~Signals (recorded)~~ |

Write controller:

Get the recording configuration from the Gui, and record the data according it.

|  |  |  |  |
| --- | --- | --- | --- |
| Description | Width | Direction | Pin Name |
| Clock | 1 | In | Clk |
| Synchronous reset signal | 1 | In | Reset |
| Identify the trigger type in order to find it in the incoming signal | 5 | In | Trigger\_type |
| The percent from the recorded data that will appear before the trigger | 7 | in | Trigger\_position |
| 0-system off, we don't search for trigger rise (set at the beginning to 0 until all other registers are ready, change in second time after the trigger was found- don't continue to search for trigger rise)  1- system on, start search for trigger rise (changed by write controller after all other registers are in the right mode). | 1 | In\out | System\_status |
| Counts the number of clk cycles that passed since the system started working until trigger raised | 8 | out | Clk\_to\_start |
| Check in the RAM for trigger rise and determine according the trigger position the start and the end addresses of the recording data |  | in | RAM\_to\_wc |
| the start and the end addresses of the recorded data |  | out | Wc\_to\_rc |

|  |  |
| --- | --- |
| outputs | Size(bit) |
| Register\_3(system status) | [0..4] |
| Register\_4(start) | 5 |
| Read controller | [6.. |
| RAM | ? |

**Read controller:**

This is the unit which sends the data out through the WBM according the resent configuration.

|  |  |  |
| --- | --- | --- |
| Inputs | Size[bite] | Description |
| clk | 0 |  |
| reset | 1 |  |
| Register\_3(system status) | [2..6] |  |
| Register\_4(start) | 7 |  |
| Write controller | [8.. | Received the Address of the trigger and the recording depth, and with the trigger position register, find in the RAM the data that needs to send out |

|  |  |  |
| --- | --- | --- |
| Output | Size[bite] | Description |
| Register\_4(start) | 0 |  |
| Register\_3(system status) | [1..5] |  |
| Data\_out | [6..13] | Send the data out to the WBM |
| WBM | ? |  |

**RAM:**

The memory unit. The RAM is recording the incoming signals all the time, and when we find the correct configuration, the read controller read the data from the RAM.

|  |  |  |
| --- | --- | --- |
| Inputs | Size(bit) | Description |
| Clk | 0 |  |
| Reset | 1 | Din\_valid |
| Signals\_in | [2..9] | Incoming data to record (data in) |
| Write controller | [10..19] | Address in. put the incoming data in that address |
| Read controller | [20..28] | Address out. Take the output data from that address |

|  |  |  |
| --- | --- | --- |
| Outputs | Size(bit) | Description |
| Data\_out | [0..15]\*(# RAM) | output data to send out |
| Dout\_valid | 16 | '1' if the output data is valid |

שאלות

* האם לממש RAM קטן יותר או להשתמש ביחידה הבסיסית אפילו אם ההקלטה קטנה יותר (ביחידה בסיסית 4096 ביטים להקלטה)

יש GENERIC-ים לטובת הגודל והרוחב של ה- RAM הבסיסי.

* האם לקבוע גודל מקסימלי של הקלטה או שהמשתמש יכול להקליט כל מספר ביטים שהוא רוצה?

יש GENERIC, והכול צריך להיות תלוי ב- GENERICS-ים.

איך מייצגים את הבעיה כאשר צריך לשלוח את עומק ההקלטה בין היחידות במערכת (לדוגמא רוחב הפס בין READ-WRITE -> מצריך שליחת כתובת, תלוי ברוחב ההקלטה)

הכול צריך להיות תלוי ב- GENERIC-ים, ואם חסר GENERIC-ים שלא חשבנו עליהם אז יש להוסיף. ה- USER צריך לקבל גמישות מלאה.

* האם לRAM יש כניסת שעון וRESET

כן.

בהקצאת המקום עבור רגיסטר- SYSTEM STATUS לדוגמא, האם למרות שזה רגיסטר שרוחבו 8 ביט, בגלל שאנחנו משתמשים רק בביט אחד שלו אפשר להגיד שרוחבו 1 או שנידרש להקצות את כל רוחב הרגיסטר?(כרגע מוקצה לו רק ביט אחד)