|  |  |
| --- | --- |
| Error Correction Encoder & Decoder | **Digital Design and Logical Synthesis for Electrical Computer Engineering**  **(36113611)**  **Course Project** |
| **Digital High Level Design**  **Version 0.1** |

**Revision Log**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Rev** | **Change** | **Description** | **Reason for change** | **Done By** | **Date** |
| 0.1 | Initial document | First Design of Encoder Decoder |  | Refael Ben Maor  Tal Kapelnik | 2.12.2021 |
| 0.2 |  |  |  |  |  |
| 0.3 |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Table of Content**

[LISt of figures 4](#_Toc175011761)

[List of tables 5](#_Toc175011762)

[2. Blocks Functional Descriptions 6](#_Toc175011767)

[2.1.1 << component/feature #1 name>> 6](#_Toc175011768)

[2.1.2 << component/feature #2 name>> 7](#_Toc175011769)

[2.1.3 << component/feature #3 name>> 7](#_Toc175011770)

[2.1.4 << component/feature #4 name>> 7](#_Toc175011771)

[3. Appendix 8](#_Toc175011772)

[3.1 Terminology 8](#_Toc175011773)

[3.2 References 8](#_Toc175011774)

# LISt of figures

[Figure 1: view of the block. 6](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355462)

[Figure 2: view of the block. 8](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355463)

[Figure 7: view of the block. 9](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355464)

[Figure 6: view of the block. 10](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355465)

[Figure 3: view of the block. 11](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355466)

[Figure 8: view of the block. 12](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355467)

[Figure 9: view of the block. 13](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355468)

[Figure 10: view of the block. 13](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355469)

[Figure 4: view of the block. 14](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355470)

[Figure 11: view of the block. 15](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355471)

[Figure 12: view of the block. 16](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355472)

[Figure 5: view of the block. 17](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355473)

[Figure 13: view of the block. 18](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355474)

[Figure 14: view of the block. 19](file:///C:\Users\TalKa\Desktop\לימודים\תכן%20ספרתי\pdf-%20EncDec\Project1x.docx#_Toc89355475)

# List of tables

[Table 1: Block interface. 9](#_Toc175011869)

# Blocks Functional Descriptions

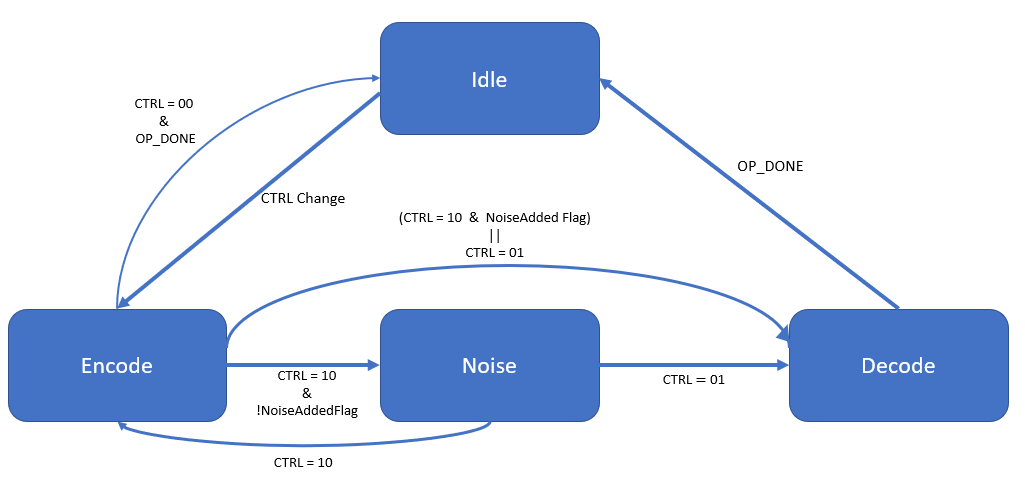
## Top Level – Encoder Decoder & Error Fixer

**Functional Description**:

This module is a slave to the CPU , which control it with the APB bus and the register files.  
The CPU send encoder or decoder data to the Error correction Encoder & Decoder diagram block and it give back the data result, operation\_done out put ‘1’ when he finish, and num\_of\_errors that tell if was between 0-2 errors.

The Top-Level Error correction Encoder & Decoder consist 4 modules:  
1) Registor\_selctor – save the information about the Error correction encoder& decoder and initial parameters and send the date to require modules and also the CPU if he want to read from the registers.  
2) Encoder – Get the data from the Registers\_selector and the Top to encode the wanted data, also this module is use for the decoder part to save a space in the design the decoder part using the same components as the decoder.  
3) Num\_of\_errors – Get data from the Top and the encoder to tell us how much errors we have in the wanted data , this data we sending out with num\_of\_errors and sending it to the next module Error\_fix .  
4) Error\_fix – Get data from the Top and the Num\_of\_errors to fix the error in the data if he can, only when we have 1 error we fixing the data at the bit spot that need the correction.

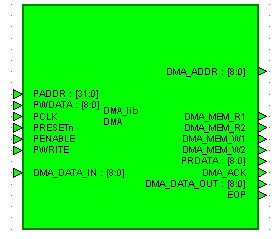
**Top State Machine:**



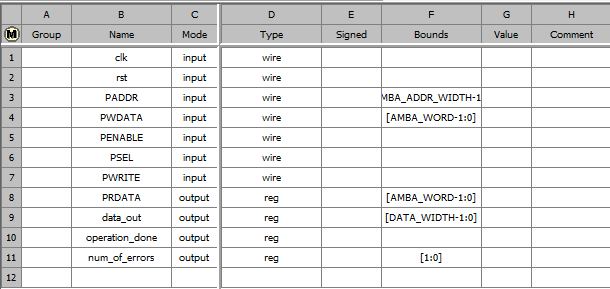
### Interface

<<Edit the titles of these subsections with your specific names>>

>> <<REMOVE



1. view of the block.
2. Block interface.view of the block.



1. Block interface.
2. view of the block.Block interface.

### Block Diagram

<< Simply copy/paste the Heading 2 line and rename it if you need more subsections that what is here. >>

### << component/feature #3 name>>

<< Delete the Heading 2 lines you don’t need. >>

### << component/feature #4 name>>

<< For any figures, number them sequentially, and copy/paste them into the document. Use Hdl Designer OLE drag and drop for drawing diagrams instead of Word’s drawing tools.. >>

## Register Selector

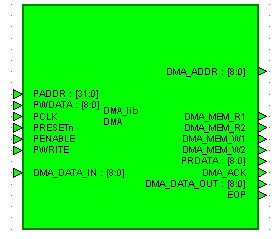
**Functional Description**:

This module has the registers files that contains all the data for the Top-Level to work, all the registers get the data from the CPU via the APB bus . The Error correction Encoder & Decoder module can only read from this module with the wires DATA\_IN\_REG, CTRL\_REG, CODEWORD\_WIDTH\_REG, NOISE\_REG, PRDATA\_REG when the PRDATA\_REG is used to send data from the registers to the CPU when he ask for it.

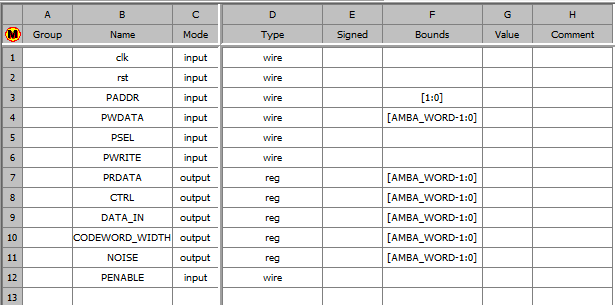
### Interface

<<Edit the titles of these subsections with your specific names>>

>> <<REMOVE

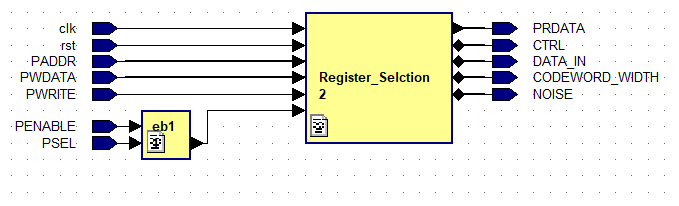


1. view of the block.
2. view of the block.



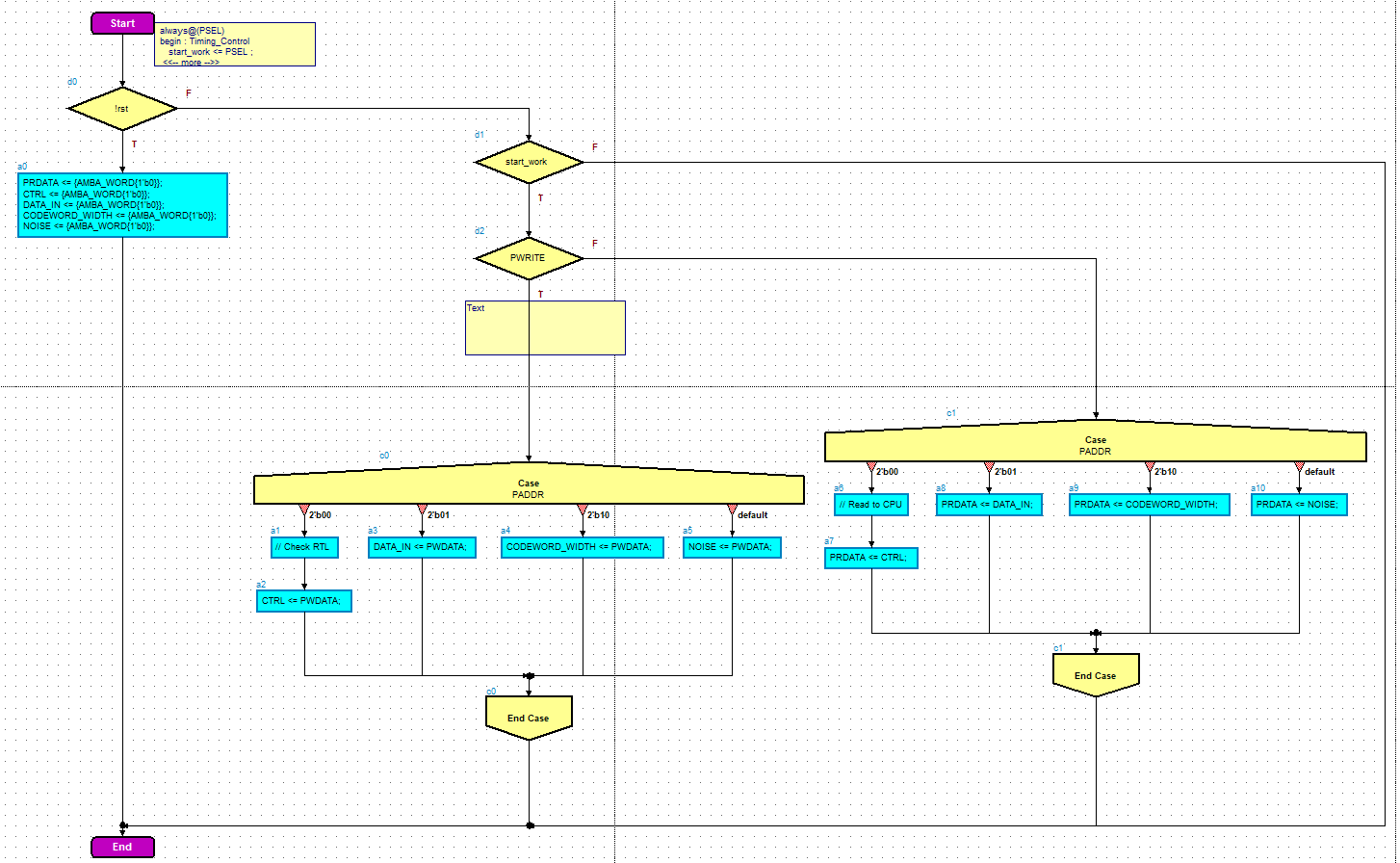
1. Block interface.
2. view of the block.Block interface.

### Block Diagram



1. view of the block.
2. view of the block.view of the block.

### Flow Diagram



1. view of the block.
2. view of the block.view of the block.

## Encoder

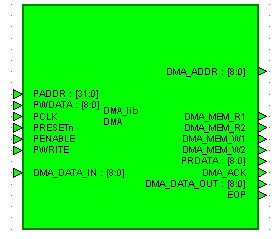
**Functional Description**:

This module gets the data from the top entity and from the Register\_Selctor module. This module is being used in each of the functions available, Encode, Decode, and full channel. For that reason, we put special attention to the details, and tried to minimize the calculation needed, in order to find the parity bits. To support all width sizes, the encoder works for each of the sizes possible. To do that, we set the data to the MSB in the input, padder with zeroes from the left LSBs.

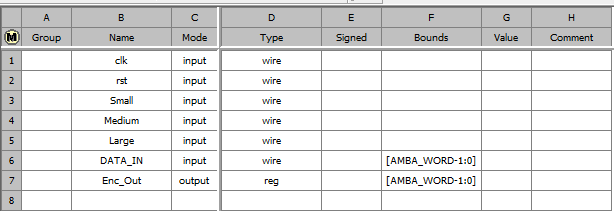
### Interface

<<Edit the titles of these subsections with your specific names>>

>> <<REMOVE

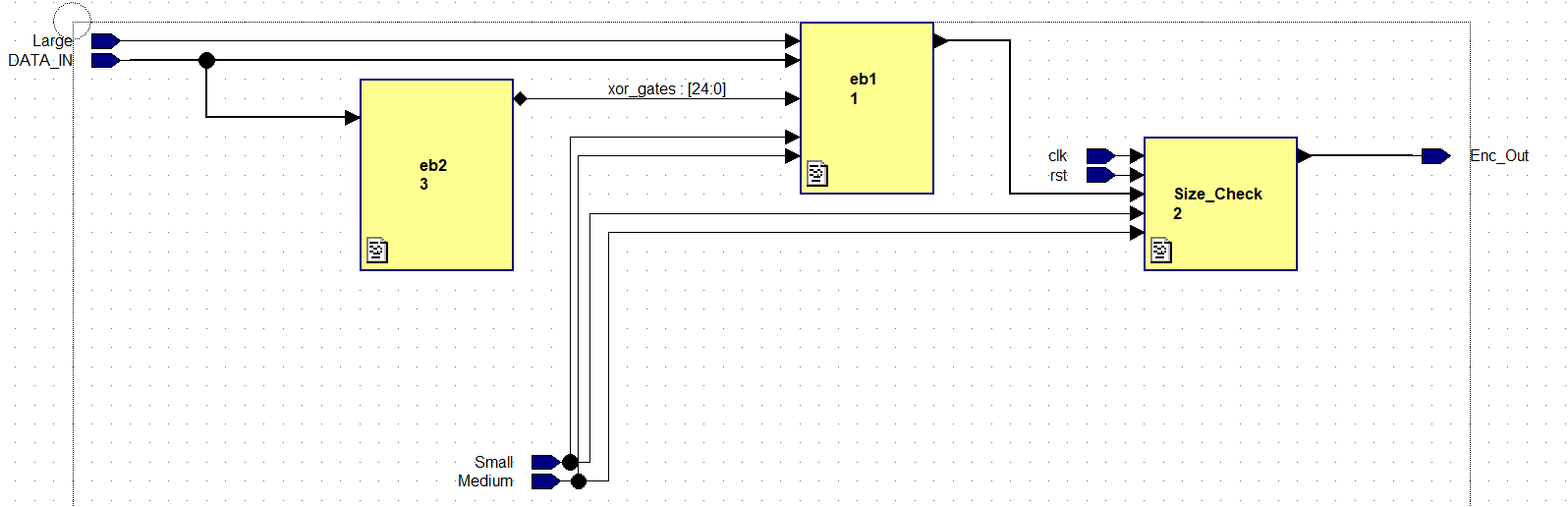


1. view of the block.
2. view of the block.



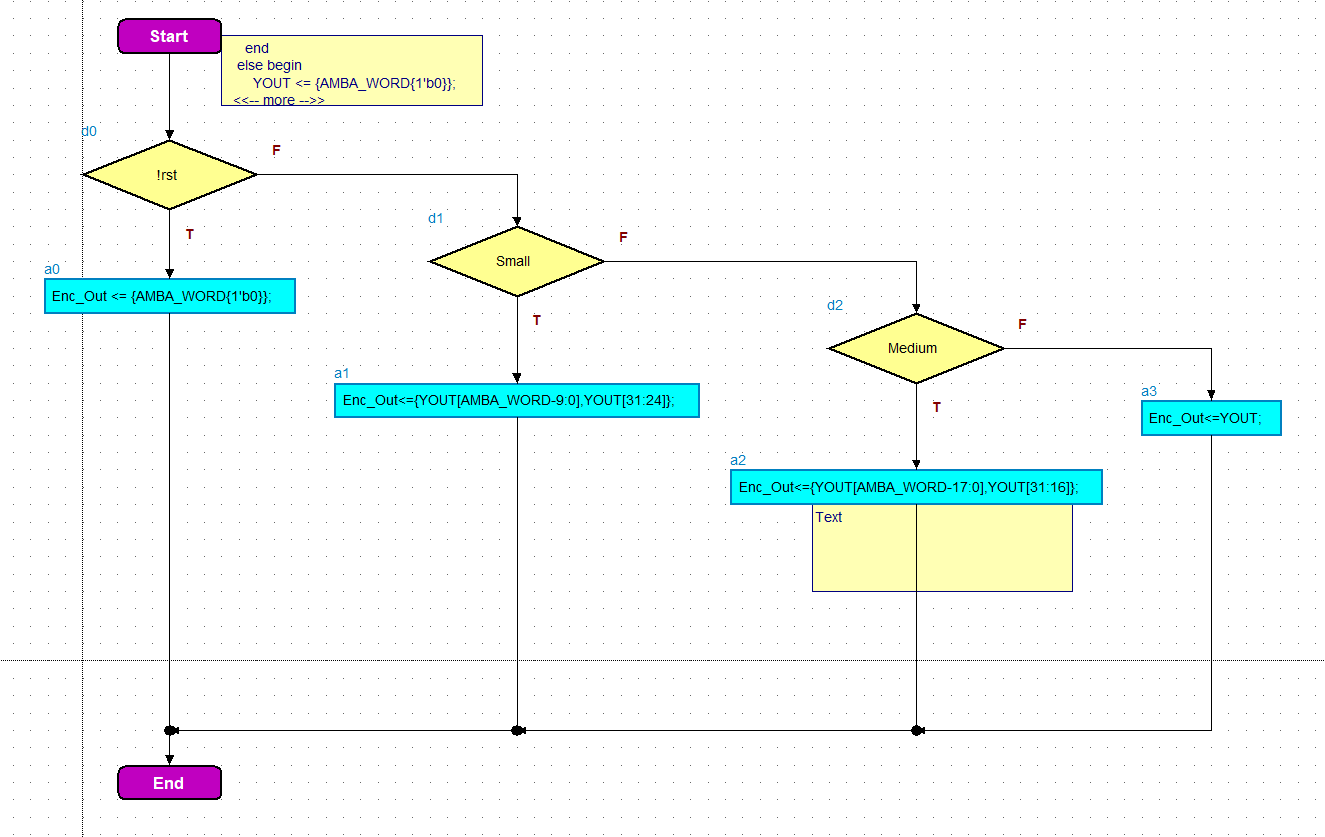
1. Block interface.
2. view of the block.Block interface.

### Block Diagram



1. view of the block.
2. view of the block.view of the block.

### Flow Diagram



1. view of the block.
2. view of the block.view of the block.

### Main XOR gates for the module

With this component we calculate the parity, in order to reduce calculation and space (XOR gates) in the design, we minimized the XOR gates by using repeating calculations for C1, … , Cn.

1. view of the block.
2. view of the block.view of the block.

## Num\_Of\_Errors

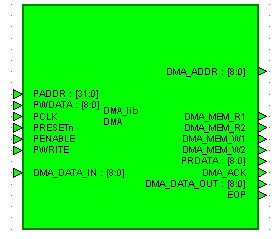
**Functional Description**:

This module gets the data from the Encoder and from the top entity. For the Input of a vector with its parity, the output will be the number of corrupted bits. The number of errors that calculated are sent to the top entity, and from there to Error\_Fix module for further calculation.

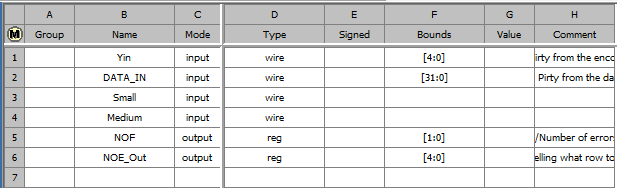
### Interface

<<Edit the titles of these subsections with your specific names>>

>> <<REMOVE



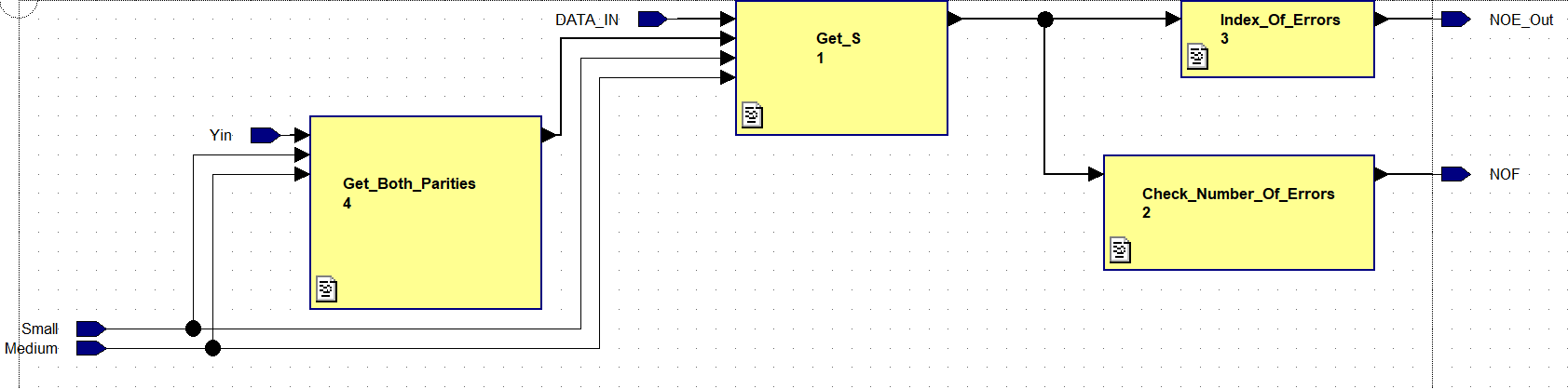
1. view of the block.
2. view of the block.



1. Block interface.
2. Block interface.

### Block Diagram

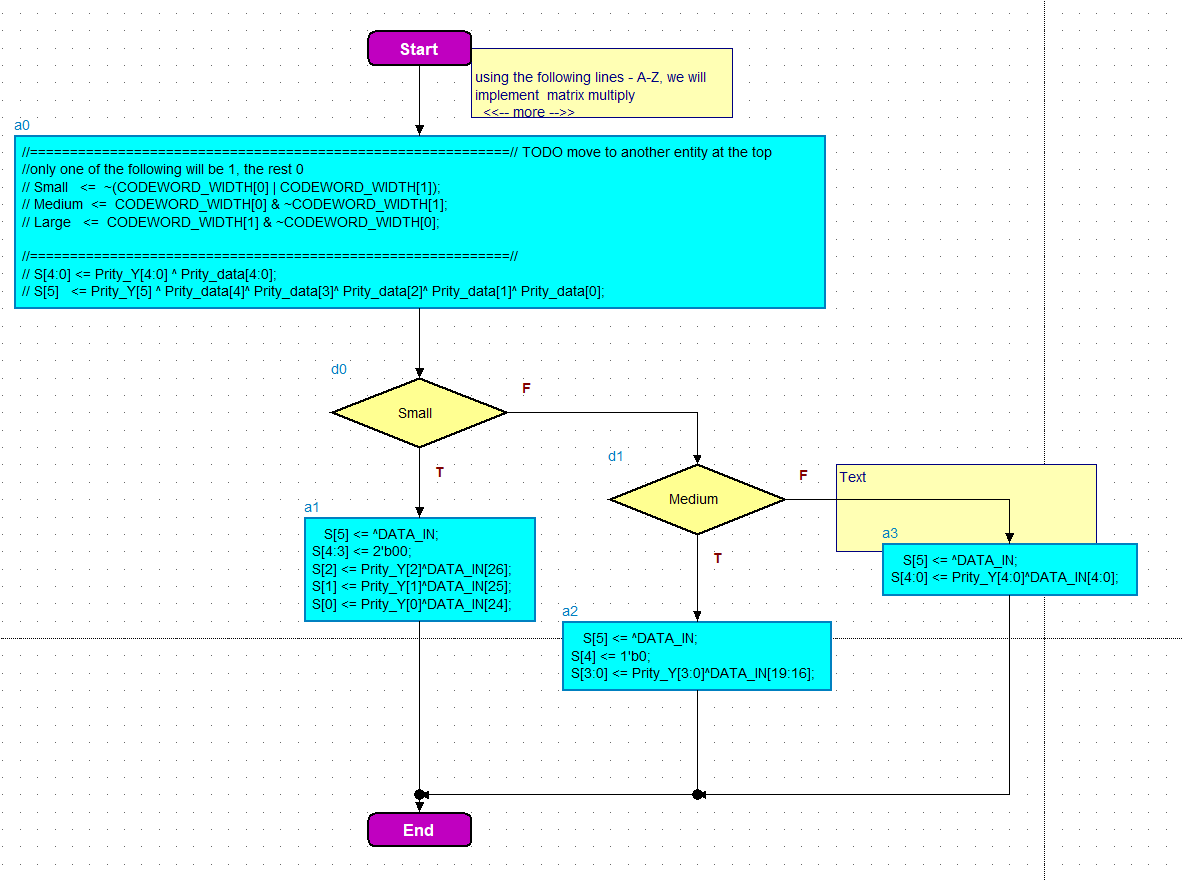
<< Simply copy/paste the Heading 2 line and rename it if you need more subsections that what is here. >>



1. view of the block.
2. view of the block.view of the block.

### Flow Diagram

<< Delete the Heading 2 lines you don’t need. >>



1. view of the block.
2. view of the block.view of the block.

### << component/feature #4 name>>

<< For any figures, number them sequentially, and copy/paste them into the document. Use Hdl Designer OLE drag and drop for drawing diagrams instead of Word’s drawing tools.. >>

## Error\_Fix

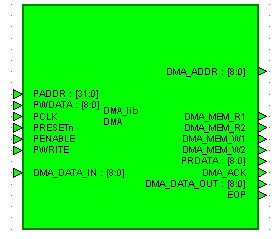
**Functional Description**:

This module gets the data from the Num\_Of\_Errors and Top modules . With the data from the Num\_Of\_Errors he know if he need to fix (only when we have one error) and also know what bit is corrupt and with the data from the Top he send the data\_out and also changing operation\_done to ‘1’ when he done.

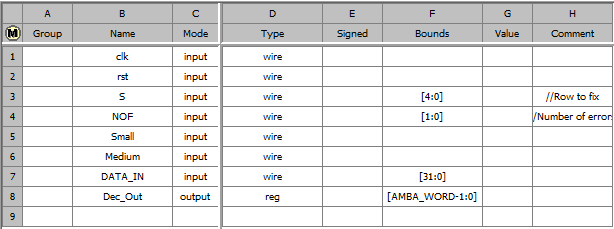
### Interface

<<Edit the titles of these subsections with your specific names>>

>> <<REMOVE



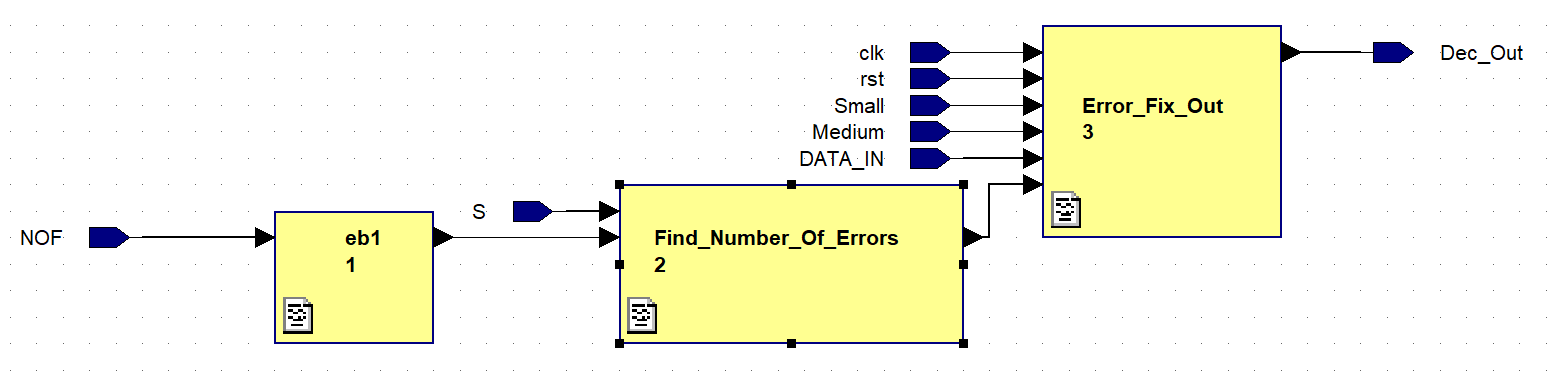
1. view of the block.
2. view of the block.view of the block.



1. Block interface.
2. view of the block.Block interface.

### Block Diagram

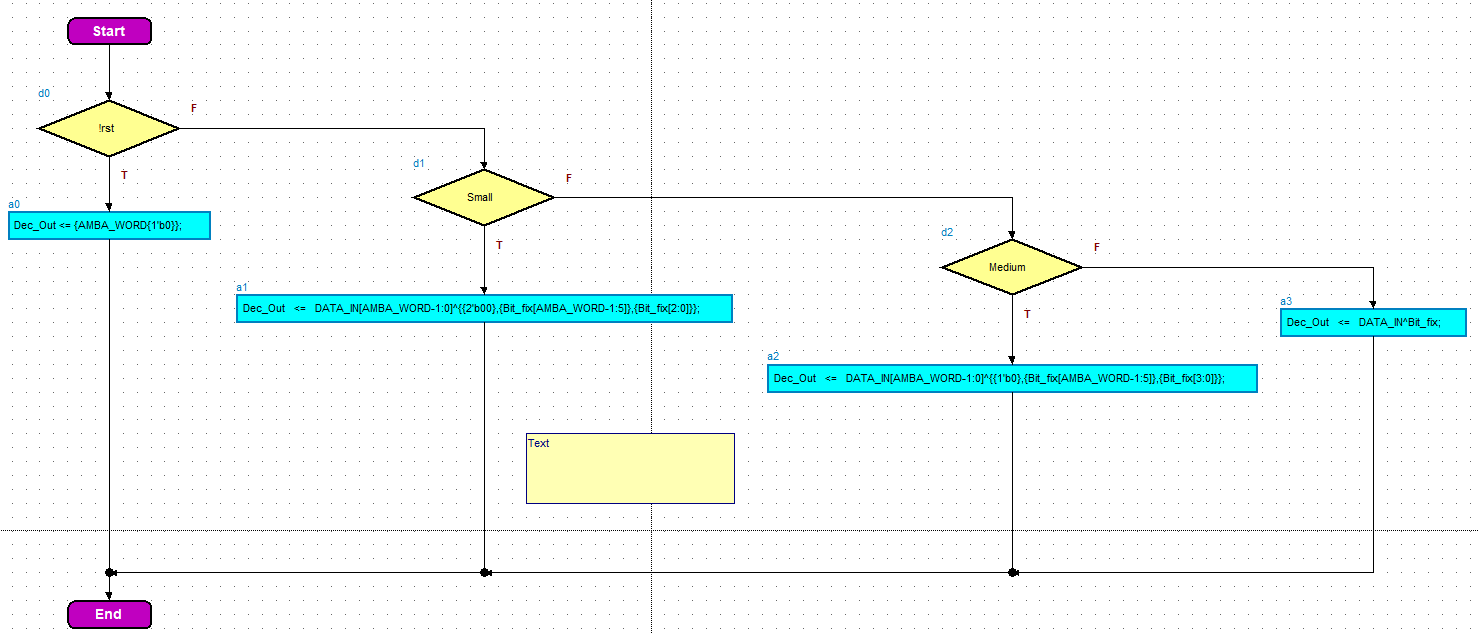
<< Simply copy/paste the Heading 2 line and rename it if you need more subsections that what is here. >>



1. view of the block.
2. view of the block.view of the block.

### Flow Diagram

<< Delete the Heading 2 lines you don’t need. >>



1. view of the block.

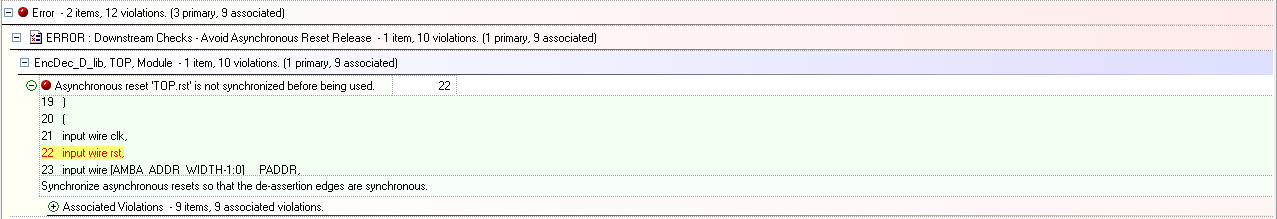
### << component/feature #4 name>>

<< For any figures, number them sequentially, and copy/paste them into the document. Use Hdl Designer OLE drag and drop for drawing diagrams instead of Word’s drawing tools.. >>

# Rules in design checker

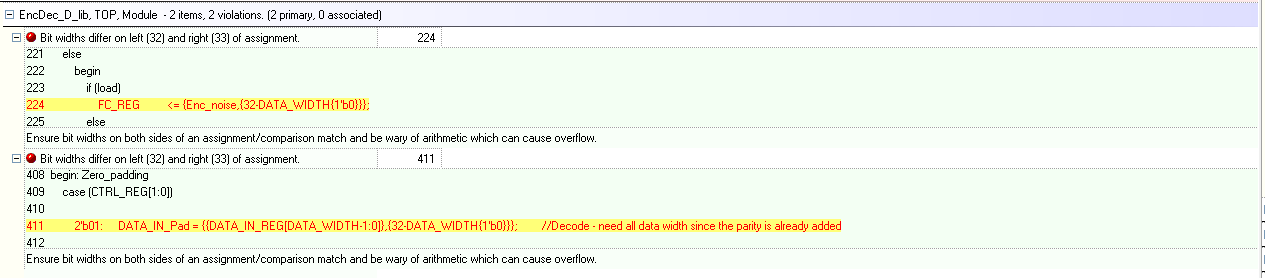
## Errors

### Rst error



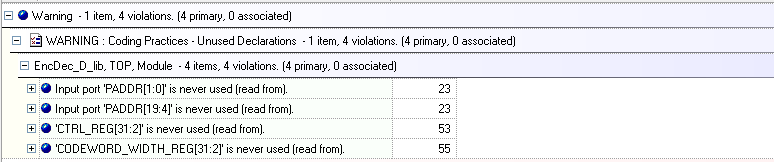
Error that was allowed to waver because that not need to show

### Size error



As we can see even though that say we have wrong bit amount we can se ethe even on both sides  
[DATA\_WIDTH-1:0] = DATA\_WIDTH amount of bits  
Data in pad is 32 bit  
so 32 = DATA\_WIDTH – (32 – DATA\_WIDTH) = 32 bits

## Warning



We keep those values for when the CPU reads from the registers, but we don’t use them inside the top structure .

# Appendix

## Terminology

**LSB** - Least Significant Bit

**TBR** - To Be Reviewed

**TBD** - To Be Defined

## References

<<I expect your team’s work to make appropriate citing of relevant literature—whether it is product manuals, the text, or other works you have read or consulted in order to do the project. If you don’t have any references, then leave the section blank. >>