

Cat Recognizer Using Neural Network

Digital Design and Logical Synthesis for Electric Computer Engineering (36113611) Course Project Test Plan Version 1.0

Maor Assayag 318550746

Refahel Shetrit 204654891

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	1 of 21

Revision Log

Rev	Change	Description	Reason for change	Done By	Date
0.1	Initial document	Basic Modules	More efficient design	Maor Assayag Refahel Shetrit	17/11/2018
0.2	Progress	All modules implemented	Complete design requirements	Maor Assayag Refahel Shetrit	27/11/2018
0.3	Test Bench	Stimulus, Checker, Coverage	Check component operations: check if we reached all component operations and got all possible outcomes, as expected.	Maor Assayag Refahel Shetrit	28/11/2018
0.4	Design check (TBD)	Verify modules with design checker	Design implementation checker	Maor Assayag Refahel Shetrit	17/12/2018

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	2 of 21

Table of Content

LIST OF FIGURES	4
LIST OF TABLES	5
5. VERIFICATION PLAN	6
5.1 Verification Test Objectives	7
5.2 Test Bench Architecture and Functionality	7
5.3 Test Bench Input & Output	שגיאה! הסימניה אינה מוגדרת.
6. VERIFICATION RESULTS	שגיאה! הסימניה אינה מוגדרת.
6.1.1 Test All_images.v	שגיאה! הסימניה אינה מוגדרת.
7. APPENDIX	21
7.1 Terminology	21
7.2 References	21

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	3 of 21

LIST OF FIGURES

Figure 1: Test Bench Block Diagram _____ 8

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	4 of 21

LIST OF TABLES

Table 1: Test Plan Functionality _____שגיאה! הסימניה אינה מוגדרת.

Table 2: Test Plan FunctionalCheckers _____ 10

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	5 of 21

1. VERIFICATION PLAN

We would verificate our design CatRecognizer, a system which recognizes a cat in an image. Our strategy is to test the reliability of our modules. Operation scenarios we tested in our test bench :

1. Standard scenarios – feeding 40 valid images to the system, while simulating a CPU that control the design (clocks, control bus etc.)
2. Extreme scenarios – feeding extreme images such as white blank image, black blank image followed by extreme weight files value (all max or min values) – those will test the extreme values that should be computed in the system.
3. Forbidden scenarios – feeding invalid images & weights values (overflow values etc.), simulate a CPU which functions poorly or slowly that control the design.

Test Number	Functionality being tested	Test data set	Expected result	Scenario
1	Writing & Reading to/from APB (pixel bank) & Weights banks	Valid image and 3 weights file (3 levels of precision)	For each address in the address space we want to see the correct value been read	Standard
2	Design result – standard input	Set of 40 valid images & 3 levels of precision	CatRecOut	Standard

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	6 of 21

1.1 Verification Test Objectives

Our verification process is consisting of 3 stages:

1. Stimulus - describe the way we can activate our design - stimulate the component inputs.
2. Function Coverage - checks if all outcomes have been reached.
3. Functional checker - checks if the design worked as expected.

1.2 Test Bench Architecture and Functionality

CatRecognizer is consist of 4 main parts: APB Register File, Weights Register File, 3 Computing Neurons and Internal logic that connects them together. CatRecognizer has 2 outputs – CatRecOut (which is the result of the model if the Image has a cat in it) and PRDATA which is APB Read Data Bus.

CatRecOut starts from no signal (high-z) and when *start_work* bit is applied to Start_work_register in the APB Register File (Address 0x00), the model start fetching relevant data from the Registers files and compute the mathematical equation to decide if the loaded image (with the loaded weights precision) have a cat in it. The process takes at least 4096 cycles for fetching (an computing "on-the-way") the data.

As we mentioned earlier, our Test Bench is consisted of 4 stages where the Stimulus is the component which is responsible for ‘mimicking’ the signals sent through the APB bus. Hence, all inputs to CatRecognizer are connected to Stimulus component outputs.

On the other end, the CatRecognizer output (PWM wave) and the Stimulus outputs are connected to both Checker component and Coverage component inputs. In this architecture design, we are able to push all permutations of inputs signal and check if the CatRecognizer is outputs correct/expected values (through Checker) and also check whether all outputs permutations has occurred (through Coverage component).

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	7 of 21

Cat Recognizer Using Neural Network Test Plan Document

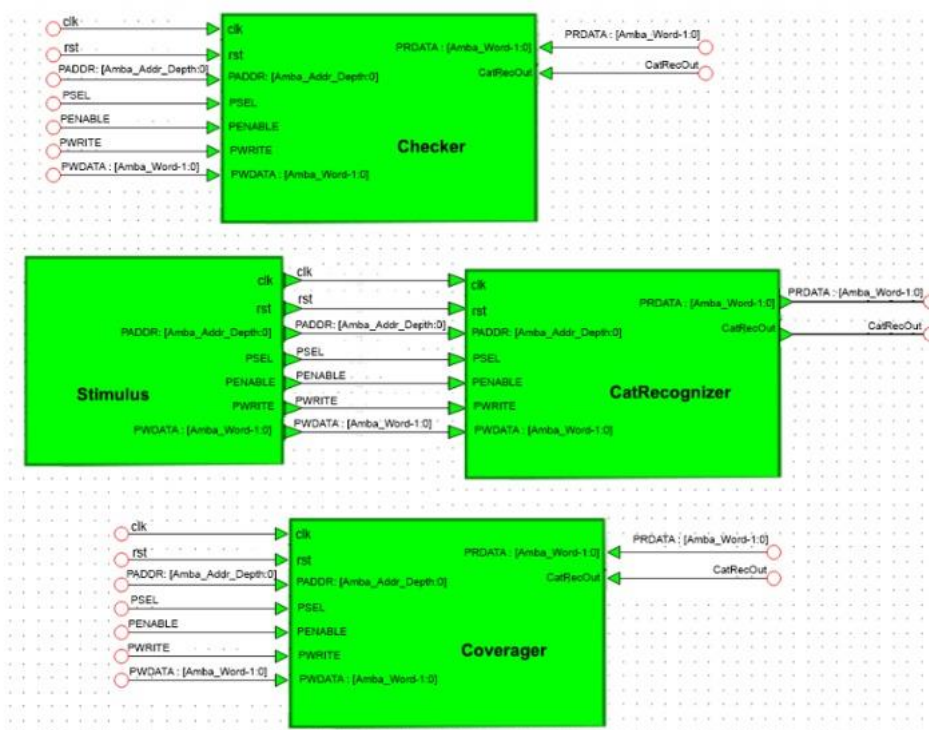


Figure 1: Test Bench Block Diagram

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	8 of 21

1.3 Functional Coverage

The table below describe the coverage of our design; we describe the cover group of CatRecognizerClk and all its cover points.

Function – name of the covered function.

Event – event the functional coverage is checked.

Coverage point – the point we check

Bins – values that should be covered.

FUNCTION	EVENT	COVERAGE POINT	BINS	scenario
Amba_Addr	Posedge clock	PADDR	[0:(2**`Amba_Addr_Depth)/4 - 1], [(2**`Amba_Addr_Depth)/4:2*(2**`Amba_Addr_Depth)/4 - 1], [2*(2**`Amba_Addr_Depth)/4:3*(2**`Amba_Addr_Depth)/4 - 1], [3*(2**`Amba_Addr_Depth)/4:4*(2**`Amba_Addr_Depth)/4 - 1]	standard
Amba_Addr_illegal	Posedge clock	PADDR	Amba_Addr_Depth<12 Amba_Addr_Depth>14	illegal
APB Bus Enable	Posedge clock	PENABLE	0:1	standard
APB Bus Select	Posedge clock	PSEL	0:1	standard
APB Write Data Bus	Posedge clock	PWDATA	[0:(2**`Amba_Word)/4 - 1], [(2**`Amba_Word)/4:2*(2**`Amba_Word)/4 - 1], [2*(2**`Amba_Word)/4:3*(2**`Amba_Word)/4 - 1], [3*(2**`Amba_Word)/4:4*(2**`Amba_Word)/4 - 1]	standard
APB Bus Write	Posedge clock	PWRITE	0:1	standard
System clock	Posedge clock	clk	0:1	standard
Reset active high	@reset	rst	0:1	standard
APB Read Data Bus	Posedge clock	PRDATA	[0:(2**`Amba_Word)/4 - 1], [(2**`Amba_Word)/4:2*(2**`Amba_Word)/4 - 1], [2*(2**`Amba_Word)/4:3*(2**`Amba_Word)/4 - 1], [3*(2**`Amba_Word)/4:4*(2**`Amba_Word)/4 - 1]	standard
CatRegonizer results	Posedge clock	CatRecOut	0:1	standard

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	9 of 21

Table 1: Test Plan Functional Coverage

1.4 Test Bench Functional Checkers

Herein, we describe the functional Checkers for each of the test conditions we have covered and what are the expected results of the specific test case.

Condition	Event	Expected Result	Scenario
Reset Active	@(posedge APB.clk)	(APB.rst==1) => APB.CatRecOut===1'bz && APB.PRDATA === {(`Amba_Word){1'bz}}	Standard
APB Stable	@(posedge APB.clk)	APB.PSEL && APB.PWRITE && APB.PENABLE==1'b0 -> \$stable(APB.PWRITE) && \$stable(APB.PWDATA) && \$stable(APB.PADDR);	Standard
CatRecOut positive check	Posedge clock	APB.CatRecOut == 1'b1 -> (currentResult > 0)	Standard
CatRecOut negative check	Posedge clock	APB.CatRecOut == 1'b0 -> (currentResult < 0)	Standard
Accumulate currentResult checker	Posedge clock	start_outputChecker == 1'b1 -> (CheckerCurrentResult == currentResult)	Standard

Table 2: Test Plan External FunctionalChecker

In The last checker we compare the current Accumulate result in CatRecognizer to local golden model.

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	10 of 21

2. VERIFICATION IMPLEMENTATION

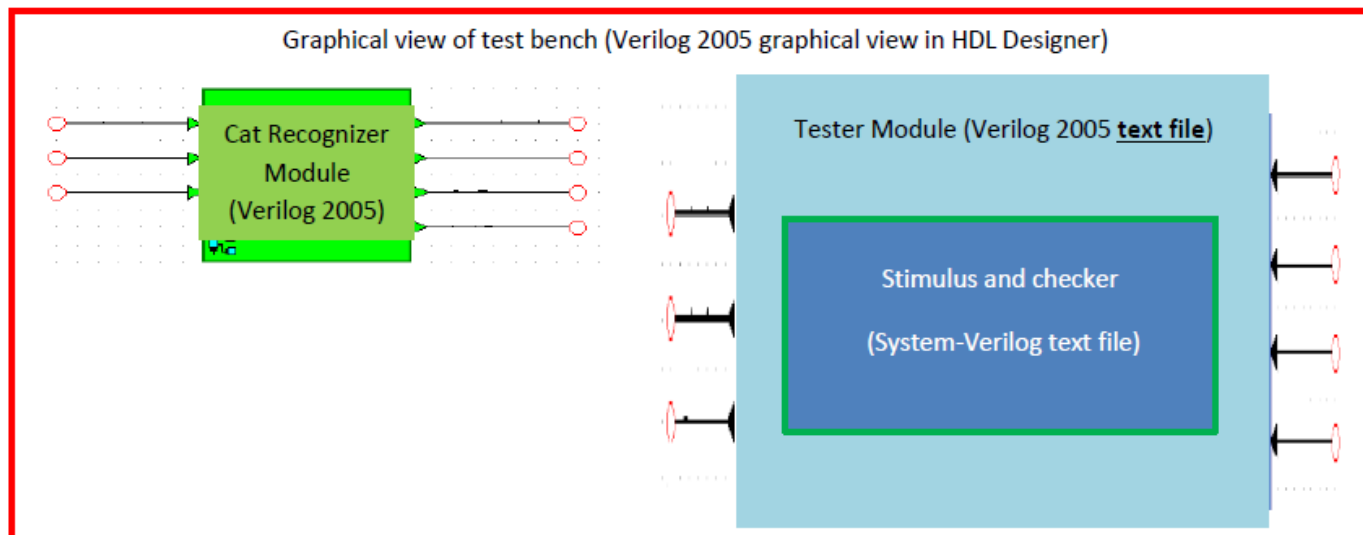


Figure 2: Combining system Verilog with graphical views of Verilog 2005

2.1. Functional Coverage

See attached Coverage.v file

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	11 of 21

2.2. Stimulus

We implemented the stimulus as described in the tables.

```
package classes;

//*****Begin Class*****
// This is the class that we will randomize.
class Randomizer #(int Amba_Word = 24);

    rand reg [Amba_Word-1:0] RandomData; //regular random variable

    // Randomization constraints.
    constraint UpperLimit {
        RandomData<(1<<(Amba_Word)-1);
    }

    // Print out the items.
    function void print();
        $display("RandomData = %0b", RandomData);
    endfunction

endclass
//*****End Class*****

endpackage
```

Figure 3: Randomizer class to randomize the data for the Random input test

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	12 of 21

2.3. Interface

```

interface APB_interface();

    logic                clk;           // system clock
    logic                rst;           // Reset active low
    logic                PENABLE;       // APB Bus Enable/clk
    logic                PSEL;          // APB Bus Select
    logic                PWRITE;        // APB Bus Write
    logic [`Amba_Addr_Depth:0] PADDR;   // APB Address Bus
    logic [`Amba_Word-1:0] PWDATA;      // APB Write Data Bus
    logic [`Amba_Word-1:0] PRDATA;      // APB Read Data Bus
    logic                CatRecOut;     // CatRecognizer result

    modport Coverage_net(input  clk,
                        rst,
                        PENABLE,
                        PSEL,
                        PWRITE,
                        PADDR,
                        PWDATA,
                        PRDATA,
                        CatRecOut);

    modport Stimulus_net (output clk,
                        rst,
                        PENABLE,
                        PSEL,
                        PWRITE,
                        PADDR,
                        PWDATA,
                        PRDATA,
                        CatRecOut);

    modport Checker_net(input clk,
                        rst,
                        PENABLE,
                        PSEL,
                        PWRITE,
                        PADDR,
                        PWDATA,
                        PRDATA,
                        CatRecOut);

    modport Cat_net(input  clk,
                    rst,
                    PENABLE,
                    PSEL,
                    PWRITE,
                    PADDR,
                    PWDATA,
                    output PRDATA, CatRecOut);
endinterface

```

Figure 4: APB Interface

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	13 of 21

3. VERFICTION RESULTS

3.1. Functional Coverage Report & Functional Checkers coverage Report

Show the report of your functional coverage for each scenario. Explain why you got this functional coverage. Could it be improved? How?

Cover Directives													
Name	Language	Enabled	Log	Count	AtLeast	Limit	Weight	Cmplt %	Cmplt graph	Included	Memory	Peak Memory	P
/Stimulus_tb/check/cover_currentResult...	SVA	✓	Off	67895	1 Unli...		1	100%		✓	0	0	
/Stimulus_tb/check/cover_CatRecOut_ne...	SVA	✓	Off	43043	1 Unli...		1	100%		✓	0	0	
/Stimulus_tb/check/cover_CatRecOut_po...	SVA	✓	Off	02410	1 Unli...		1	100%		✓	0	0	
/Stimulus_tb/check/cover_stable_apb_ch...	SVA	✓	Off	31932	1 Unli...		1	100%		✓	0	0	
/Stimulus_tb/check/cover_rst_check	SVA	✓	Off	406	1 Unli...		1	100%		✓	0	0	

COVERGROUP COVERAGE:

Covergroup	Metric	Goal	Status
TYPE /Stimulus_tb/Coverager/CatRecognizer_clk		87.5%	100 Uncovered
covered/total bins:	18 22		
missing/total bins:	4 22		
% Hit:	81.8%	100	
Coverpoint CatRecognizer_clk::rst_check		100.0%	100 Covered
covered/total bins:	2 2		
missing/total bins:	0 2		
% Hit:	100.0%	100	
Coverpoint CatRecognizer_clk::PADDDR_check		100.0%	100 Covered
covered/total bins:	4 4		
missing/total bins:	0 4		
% Hit:	100.0%	100	
Coverpoint CatRecognizer_clk::PENABLE_check		100.0%	100 Covered
covered/total bins:	2 2		
missing/total bins:	0 2		
% Hit:	100.0%	100	
Coverpoint CatRecognizer_clk::PSEL_check		100.0%	100 Covered
covered/total bins:	2 2		
missing/total bins:	0 2		
% Hit:	100.0%	100	
Coverpoint CatRecognizer_clk::PWDATA_check		100.0%	100 Covered
covered/total bins:	4 4		
missing/total bins:	0 4		
% Hit:	100.0%	100	
Coverpoint CatRecognizer_clk::PWRITE_check		100.0%	100 Covered
covered/total bins:	2 2		
missing/total bins:	0 2		
% Hit:	100.0%	100	
Coverpoint CatRecognizer_clk::PRDATA_check		0.0%	100 ZERO
covered/total bins:	0 4		
missing/total bins:	4 4		
% Hit:	0.0%	100	
Coverpoint CatRecognizer_clk::CatRecOut_check		100.0%	100 Covered
covered/total bins:	2 2		
missing/total bins:	0 2		
% Hit:	100.0%	100	
Covergroup instance vStimulus_tb/Coverager/CarRecCoverageClk		87.5%	100 Uncovered
covered/total bins:	18 22		

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	14 of 21

Cat Recognizer Using Neural Network Test Plan Document

```

missing/total bins:          4    22
% Hit:                      81.8%   100
Coverpoint rst_check          100.0%   100  Covered
  covered/total bins:        2      2
  missing/total bins:        0      2
  % Hit:                    100.0%   100
    bin rst_bins[0]          1249596    1  Covered
    bin rst_bins[1]           404      1  Covered
Coverpoint PADDR_check        100.0%   100  Covered
  covered/total bins:        4      4
  missing/total bins:        0      4
  % Hit:                    100.0%   100
    bin PADDR_bins[0]        998005    1  Covered
    bin PADDR_bins[1]        83968    1  Covered
    bin PADDR_bins[2]        83968    1  Covered
    bin PADDR_bins[3]        83968    1  Covered
Coverpoint PENABLE_check      100.0%   100  Covered
  covered/total bins:        2      2
  missing/total bins:        0      2
  % Hit:                    100.0%   100
    bin PENABLE_bins[0]      1081983    1  Covered
    bin PENABLE_bins[1]      168017    1  Covered
Coverpoint PSEL_check         100.0%   100  Covered
  covered/total bins:        2      2
  missing/total bins:        0      2
  % Hit:                    100.0%   100
    bin PSEL_bins[0]         1      1  Covered
    bin PSEL_bins[1]       1249989    1  Covered
Coverpoint PWDATA_check       100.0%   100  Covered
  covered/total bins:        4      4
  missing/total bins:        0      4
  % Hit:                    100.0%   100
    bin PWDATA_bins[0]       1004661    1  Covered
    bin PWDATA_bins[1]       89898    1  Covered
    bin PWDATA_bins[2]       84938    1  Covered
    bin PWDATA_bins[3]       70492    1  Covered
Coverpoint PWRITE_check       100.0%   100  Covered
  covered/total bins:        2      2
  missing/total bins:        0      2
  % Hit:                    100.0%   100
    bin PWRITE_bins[0]       350041    1  Covered
    bin PWRITE_bins[1]       899949    1  Covered
Coverpoint PRDATA_check       0.0%    100  ZERO
  covered/total bins:        0      4
  missing/total bins:        4      4
  % Hit:                    0.0%    100
    bin PRDATA_bins[0]       0      1  ZERO
    bin PRDATA_bins[1]       0      1  ZERO
    bin PRDATA_bins[2]       0      1  ZERO
    bin PRDATA_bins[3]       0      1  ZERO
Coverpoint CatRecOutt_check   100.0%   100  Covered
  covered/total bins:        2      2
  missing/total bins:        0      2
  % Hit:                    100.0%   100
    bin CatRecOutt_bins[0]    643043    1  Covered
    bin CatRecOutt_bins[1]    102410    1  Covered

```

TOTAL COVERGROUP COVERAGE: 87.5% COVERGROUP TYPES: 1

DIRECTIVE COVERAGE:

```

-----
Name          Design Design  Lang File(Line)  Count Status
Unit  UnitType
-----

```

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	15 of 21

Cat Recognizer Using Neural Network Test Plan Document

```
/Stimulus_tb/check/cover__currentResult_check
Checker Verilog SVA
C: /Users/MaorA/Desktop/CatRecognizer/CatRecognizer/CatRecognizer_lib/hdl/Checker.v(107)
167895 Covered
/Stimulus_tb/check/cover__CatRecOut_negative_check
Checker Verilog SVA
C: /Users/MaorA/Desktop/CatRecognizer/CatRecognizer/CatRecognizer_lib/hdl/Checker.v(96)
643043 Covered
/Stimulus_tb/check/cover__CatRecOut_positive_check
Checker Verilog SVA
C: /Users/MaorA/Desktop/CatRecognizer/CatRecognizer/CatRecognizer_lib/hdl/Checker.v(89)
102410 Covered
/Stimulus_tb/check/cover__stable_apb_check
Checker Verilog SVA
C: /Users/MaorA/Desktop/CatRecognizer/CatRecognizer/CatRecognizer_lib/hdl/Checker.v(78)
731932 Covered
/Stimulus_tb/check/cover__rst_check Checker Verilog SVA
C: /Users/MaorA/Desktop/CatRecognizer/CatRecognizer/CatRecognizer_lib/hdl/Checker.v(66)
406 Covered

TOTAL DIRECTIVE COVERAGE: 100.0% COVERS: 5
```

In our case, we got 100% Function Coverage. Our stimulus loading 40 different images to the Design and run a full simulation of computing the results. The bins should be fill, which tell us that we are in the right direction.

The simulation runs without any warning or errors, the checker is done successfully.

```
# vsim -voptargs="+acc" work.Stimulus_tb
# Start time: 18:41:05 on Dec 20,2018
# ** Note: (vsim-8009) Loading existing optim
#
# Loading sv_std.std
# Loading work.Stimulus_tb(fast)
# Loading work.APB_interface(fast_1)
# Loading work.CatRecognizerWrapper(fast)
# Loading work.CatRecognizer(fast)
# Loading work.APB(fast)
# Loading work.WeighstBank(fast)
# Loading work.Neuron(fast)
# Loading work.Coverage(fast)
# Loading work.classes(fast)|
# Loading work.Stimulus_v_unit(fast)
# Loading work.Stimulus(fast)
# Loading work.Checker(fast)
VSIM 280> run
# RandomData = 10100000111100110011010
```

Choose 5000 us for the simulation time.

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	16 of 21

3.2. Code Coverage Report

Show the report of your code coverage for all the scenarios. **Explain why you got this code coverage.** Could it be improved? How?.

We couldn't figure out how show those images, but the results are known :

The only code lines that we didn't get into was those who inside a case. For e.g. the Weights in Weight Register File is preloaded on reset according to the current WeightPrecision using a case statement.

4. COMPARING TO GOLDEN MODEL

We can implement the fetching from the the golden model results again in our new verification environment, but we decided to use our previous test bench which is written in VERILOG and compare our design with the golden model running all 40 images.

4.1 AllImages_tb.v

For testing the module CatRecognizer we built a test bench that will simulate a CPU.

First, the test bench will fetch 40 correct answer for 40 images. We can decide which weights precision we want to test (5/8/16) and change only ONE integer in the simulation – WeightPrecision.

Secondly, we made sure to reset the design correctly.

Then, for each image we will read 3 lines from the correspond text file (for e.g. image4.txt) and write to the APB interface the data and address of the concatenation data that we extracted. The writing process according to AMBA protocol involved writing SEL and only in the next cycle writing ENABLE and enable writing to the APB bank registers.

After each image is writing we compare the output answer of CatReconizer to the right answer and write to the MODELSIM terminal the results (success / fail).

After all 40 images has been process and calculate we display the final result of how much the model is accurate.

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	17 of 21

Cat Recognizer Using Neural Network Test Plan Document

The simulation required 4800us time.

```
# TestBench for image 1 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 2 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 3 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 4 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 5 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 6 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 7 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 8 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 9 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 10 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 11 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 12 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 13 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 14 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 15 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 16 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 17 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 18 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 19 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 20 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 21 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 22 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 23 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 24 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 25 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 26 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 27 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 28 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 29 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 30 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 31 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 32 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 33 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 34 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 35 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 36 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 37 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 38 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 39 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench done successfully, CatRecgonizer is accurate
```

Figure 11: Simulation results of WeightPrecsion = 16

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	18 of 21

```
# TestBench for image 2 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 3 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 4 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 5 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 6 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 7 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 8 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 9 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 10 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 11 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 12 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 13 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 14 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 15 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 16 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 17 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 18 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 19 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 20 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 21 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 22 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 23 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 24 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 25 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 26 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 27 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 28 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 29 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 30 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 31 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 32 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 33 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 34 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 35 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 36 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 37 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 38 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 39 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench done successfully, CatRecgonizer is accurate
```

Figure 12: Simulation results of WeightPrecsion = 8

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	19 of 21

```
# TestBench for image 1 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 2 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 3 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 4 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 5 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 6 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 7 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 8 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 9 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 10 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 11 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 12 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 13 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 14 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 15 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 16 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 17 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 18 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 19 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 20 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 21 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 22 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 23 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 24 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 25 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 26 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 27 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 28 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 29 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 30 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 31 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 32 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 33 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 34 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 35 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 36 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 37 : true, Correct Answer = 1, CatRecgonizer = 1
# TestBench for image 38 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench for image 39 : true, Correct Answer = 0, CatRecgonizer = 0
# TestBench done successfully, CatRecgonizer is accurate
```

Figure 13: Simulation results of WeightPrecsion = 5

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	20 of 21

5. APPENDIX

5.1 Terminology

LSB - Least Significant Bit

TBR - To Be Reviewed

TBD - To Be Defined

5.2 References

Classification:	Template Title:	Owner	Creation Date	Page
Logic Design Course	General Test Plan	Maor Assayag Refahel Shetrit	13, Dec, 2018	21 of 21