A Design Methodology for the Implementation of a Fuzzy Logic Traffic Controller Using FPGA Technology

Mandar Ambre, Bing W.Kwan and Leonard J.Tung

Department of Electrical & Computer Engineering

FAMU-FSU College of Engineering, Florida State University

THE HUNTSVILLE SIMULATION CONFERENCE HSC 2003

Outline

- In this paper we suggest the use of state diagrams for capturing a rule base of a fuzzy traffic control system.
- The fuzzy rule base for the controller is described using the state diagrams and the fuzzy inference based on the fuzzy rules is implemented using MATLAB code.
- The output of the MATLAB program is stored in a ROM for use in the VHDL code.
- A complete description of the system is assembled in VHDL and is synthesized using VHDL-based logic synthesis and Synopsys design compiler is used to perform logic synthesis

What is Fuzzy Logic?

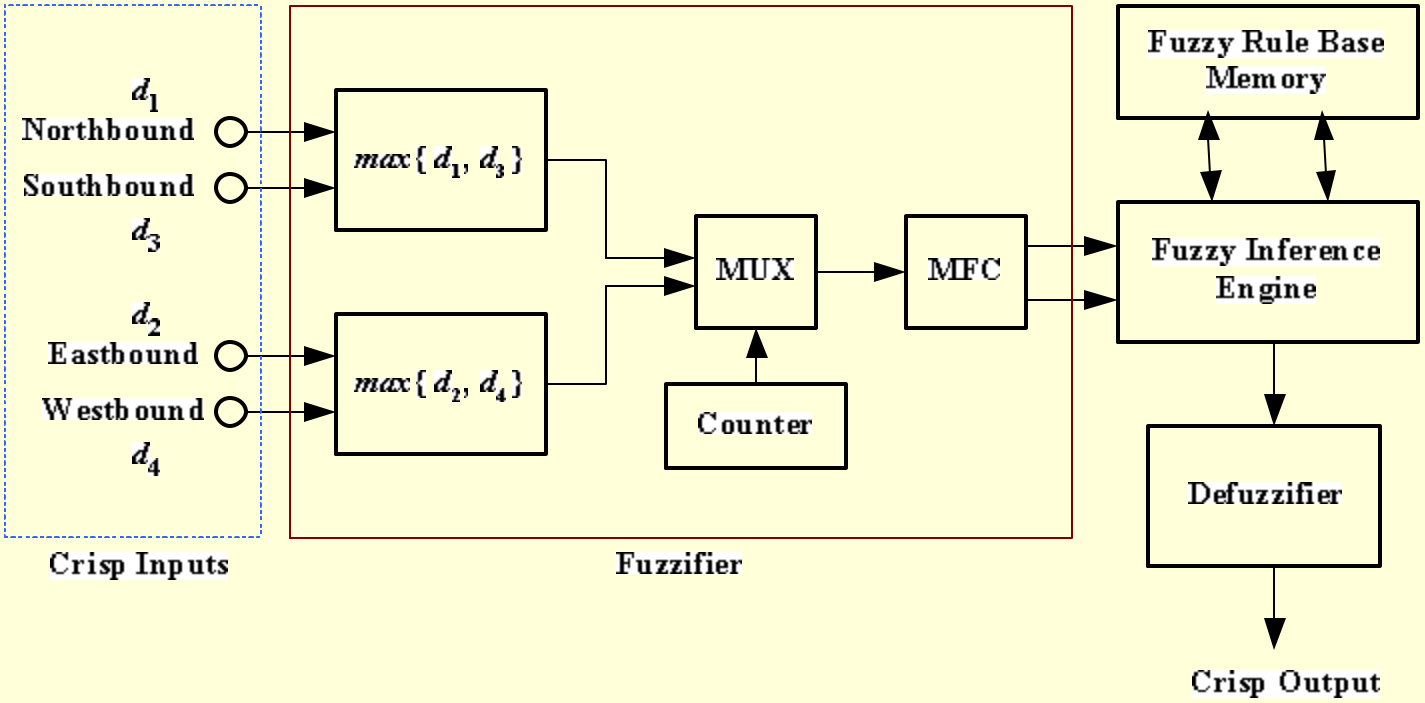
- Fuzzy logic refers to a logical system that generalizes the crisp true-or-false concept to a matter of degree.

- Fuzzy logic provides the mechanism by which numerical and linguistic information can be incorporated in a systematic manner.

Need of Fuzzy Logic in Traffic Controllers

- Traffic signaling at a given intersection is not truly adaptive because the settings can only be altered manually or by traffic control center.
- One of the desirable features of traffic controllers is to dynamically effect the change of signal phase durations without looking up pre-compiled information.
- This problem can be solved by use of fuzzy traffic controllers which are capable of signaling adaptively at an intersection.

Basic Structure of a Fuzzy Traffic Controller



Components of Fuzzy Traffic Controller

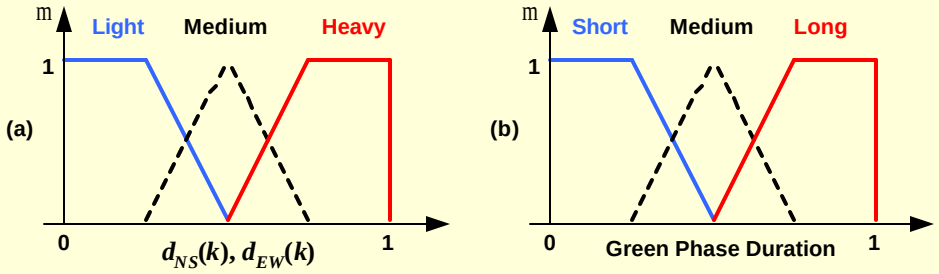
- **FUZZIFIER** – Fuzzifier takes the crisp inputs to a fuzzy controller and converts them into fuzzy inputs.
- **FUZZY RULE BASE** – It consists of fuzzy IF-THEN rules that form the heart of a fuzzy inference system.
- **FUZZY INFERENCE ENGINE** - Fuzzy Inference Engine makes use of fuzzy logic principles to combine the fuzzy IF-THEN rules.
- **DEFUZZIFIER** – It extracts a crisp value from a fuzzy set.

Fuzzy Rule Base

	Traffic Flow Condition								
NS-bound traffic	Light	Light	Light	**Med**	Med	Med	Heavy	Heavy	Heavy
EW-bound traffic	Light	Med	Heavy	**Light**	Med	Heavy	Light	Med	Heavy
NS green phase	Short	Short	Short	**Med**	Med	Med	Long	Long	Long
EW green phase	Short	Med	Long	**Short**	Med	Long	Short	Med	Long
	Duration								

IF-THEN rule base is used for fuzzy reasoning by the fuzzy traffic controller

Membership Functions

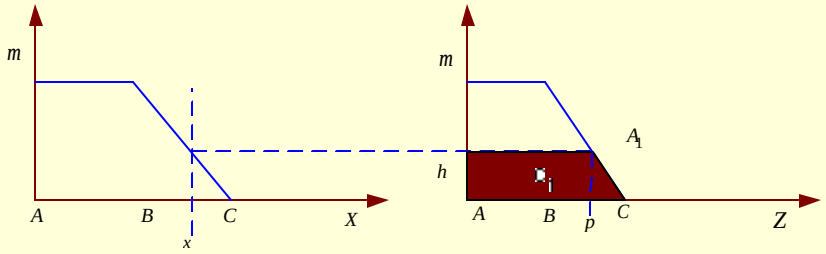


MEMBERSHIP FUNCTIONS FOR VARIOUS FUZZY SETS

- In this paper for simplicity we have considered fuzzy controllers that involve a small number of fuzzy sets with simple membership functions.

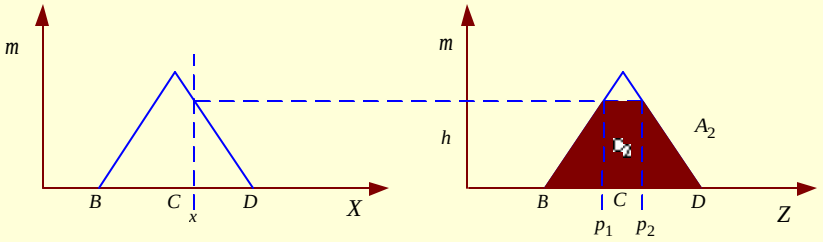
- Fuzzy sets of traffic flow are labeled as “Light”, “ Medium” and “Heavy” and fuzzy sets “Short”, “Medium” and “ Long” are associated with green phase duration.

Fuzzy Implementation



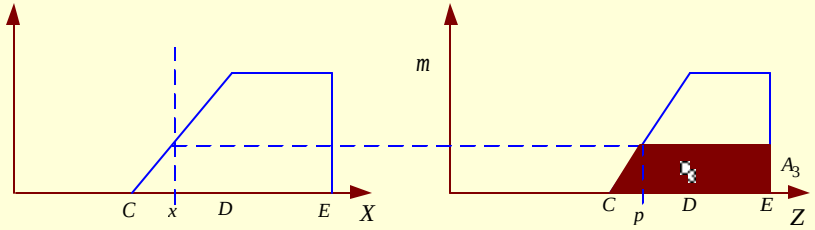
$$A_1 = h(C - A) + \frac{h^2 (B - C)}{2}$$

$$c_1 = \frac{5p^2 + 7pC - 11Ap - AC}{6(p + C - 2A)} \quad where\ p = h(b - c) + c$$



$$A_2 = h(D - B) + h^2 (C - B)$$

$$c_2 = \frac{b + d}{2}$$



$$A_3 = h(E - C) + \frac{h^2 (C - D)}{2}$$

$$c_3 = \frac{5p^2 + 7pC - 11Ep - EC}{6(p + C - 2E)} \quad where\ p = h(d - c) + c$$

A moments method is proposed as another form of defuzzification. This scheme weights the individual fuzzy sets with the respective areas. The ordinate values of the center of the fuzzy sets are used as the moments

Design Methodology

- The high level description of the fuzzy rule base needs to be converted to VHDL code.
- To simplify the process we design the fuzzy controller using MATLAB.
- The output of the MATLAB program is stored in the ROM for use in the VHDL code.
- A complete description of the system is assembled in VHDL and is synthesized using VHDL-based logic synthesis.
- Synopsys design compiler is used to perform logic synthesis.

MATLAB Implementation

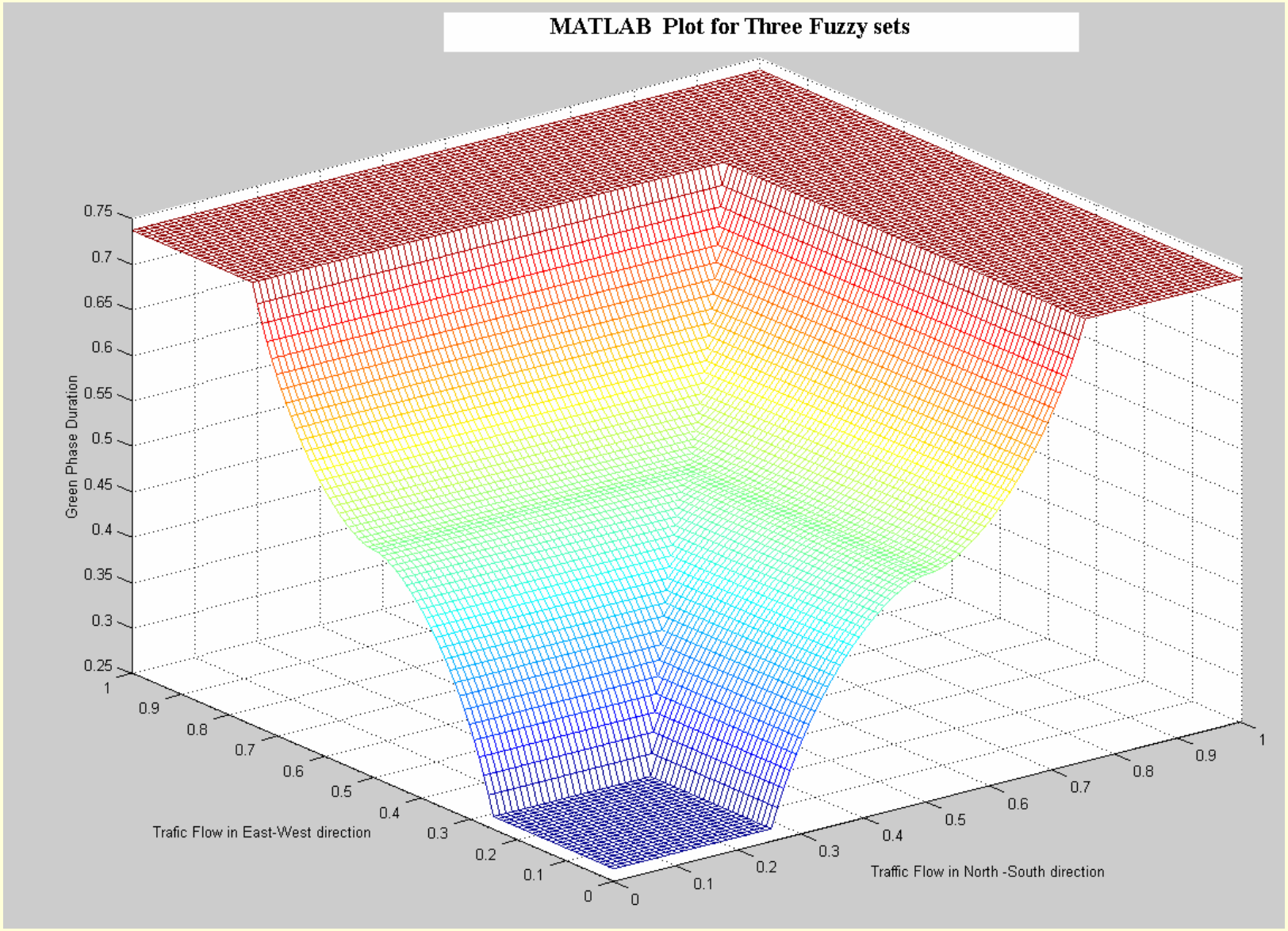
- Moments method is used to find the general form of the X co-ordinate value of the centroid or the fuzzy lengths given by

$$Z_{MA} = \frac{\sum_{i=1}^{n} A_i Z_i}{\sum_{i=1}^{n} Z_i}$$

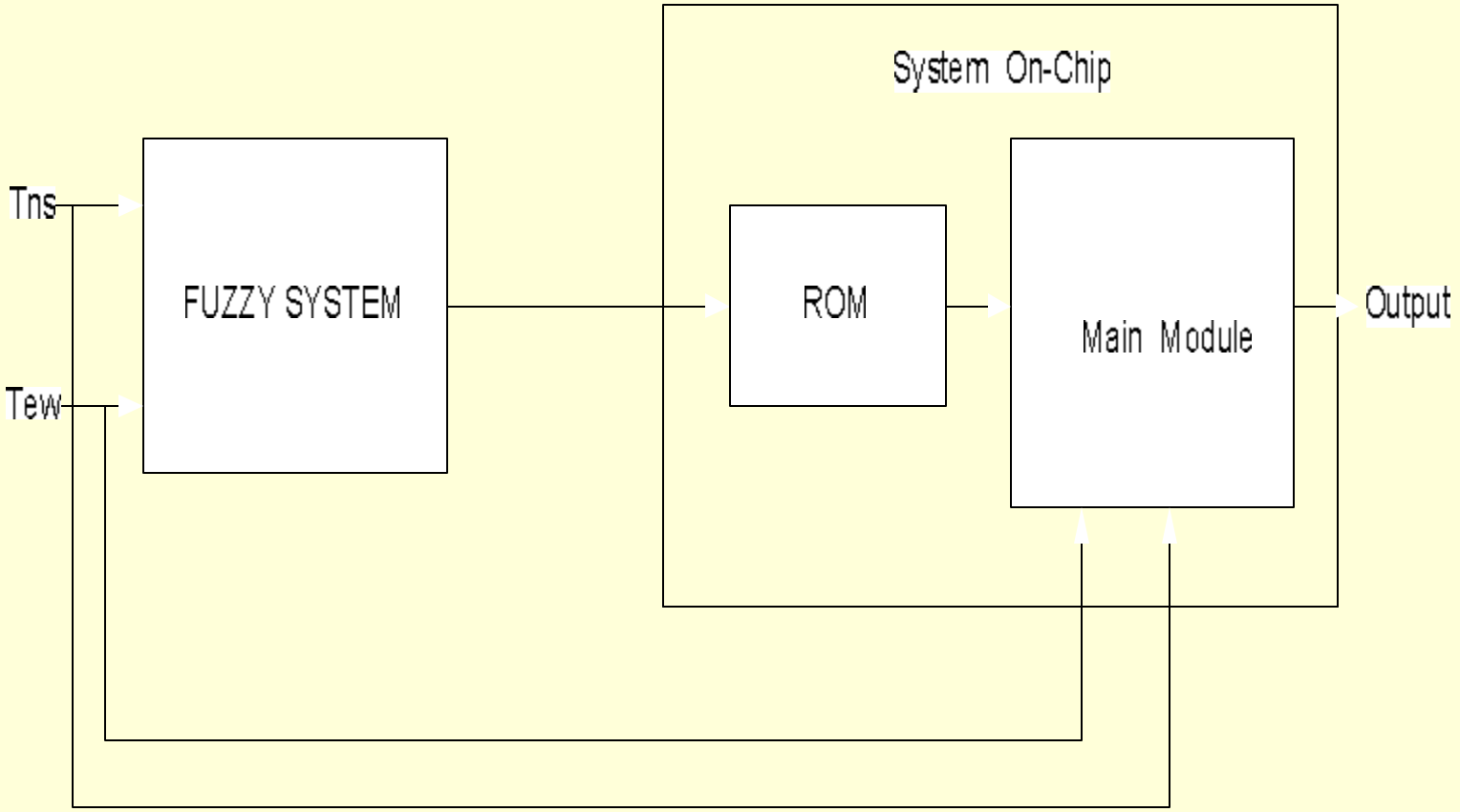
where Ai- Area and Zi- center of area for various fuzzy sets from 1 to n

- Generated MATLAB plot for Fuzzy system with 3 Fuzzy sets
- Wrote a MATLAB code that generates plot for n Fuzzy sets

MATLAB plot for Fuzzy System with Three Fuzzy sets



Block Diagram of the System



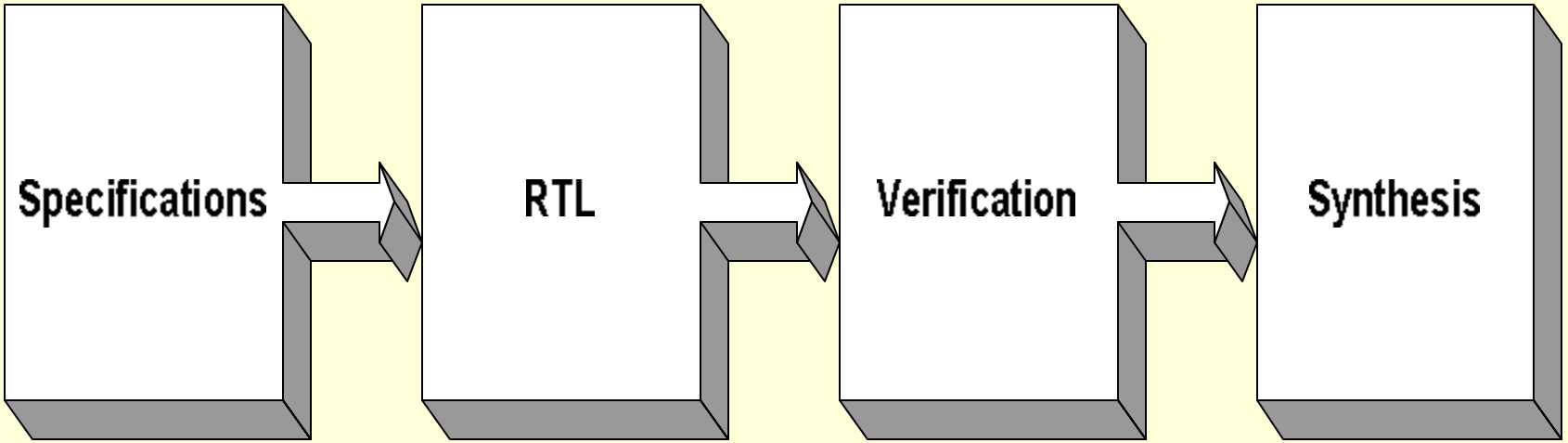
Tns- Traffic in North-South Direction
Tew- Traffic in East-West Direction

ROM Implementation

Addr	+0
0	0000000010001110
1	0000000010001110
2	0000000010001110
3	0000000010001110
4	0000000010001110
5	0000000010001110
6	0000000010001110
7	0000000010001110
8	0000000010001110
9	0000000010001110
10	0000000010001110
11	0000000010001110
12	0000000010001110
13	0000000010001110
14	0000000010001110
15	0000000010001110
16	0000000010001110
17	0000000010001110
18	0000000010001110
19	0000000010001110
20	0000000010001110
21	0000000010001110
22	0000000010001110
23	0000000010001110
24	0000000010001110
25	0000000010001110
26	0000000000000100
27	0000000011111011
28	0000000010000101
29	0000000010110001
30	0000000010001001
31	0000000000011000
32	0000000001100101
33	0000000001111001
34	0000000001010111
35	0000000000000110
36	0000000010001000
37	0000000011100001
38	0000000000010100
39	0000000000100011
40	0000000000001111
41	0000000011011001
42	0000000010000001
43	0000000000000111
44	0000000001101010
45	0000000010101000
46	0000000010111101
47	0000000010100110
48	0000000001011011
49	0000000011010011
50	0000000000000000

- Output values that are obtained by using MATLAB plot are stored in the ROM.
- The output of the ROM is given to the main programming module.

Different Stages of Hardware Implementation



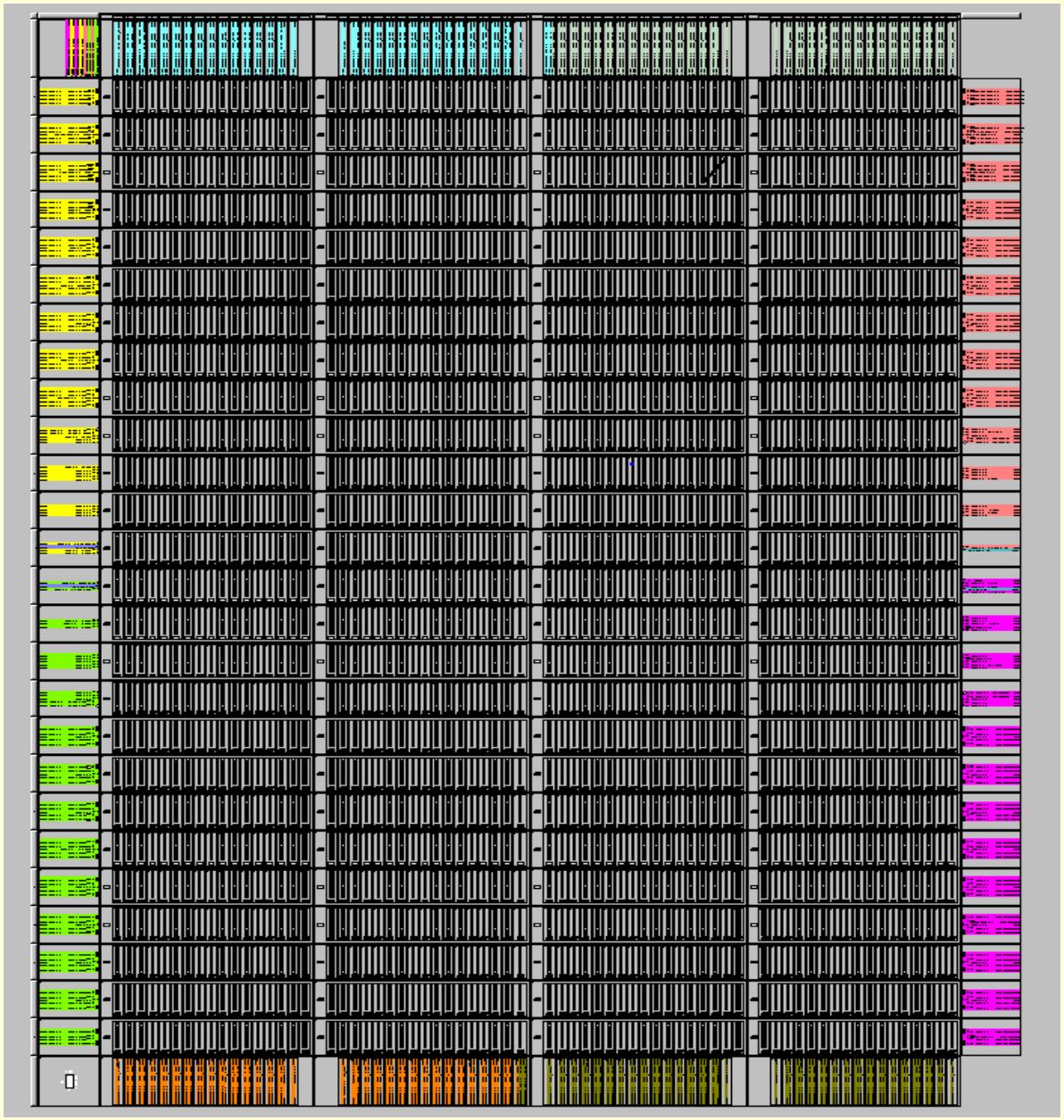
RTL Design

- Developing block- and chip-level specifications
- Developing and optimizing new Register Transfer Level (RTL) code
- Developing intelligent Testbenches
- Improving verification methodology and coverage

FPGA Device Family

- QUARTUS II version 3.0 software is used for compilation and simulation purpose.
- APEX II FPGA device was chosen because it provides high bandwidth and high performance.
- The device has 16,640 to 67,200 logic elements (LEs) and from 416 to 1,120Kbits of memory device.

APEX II SOPC

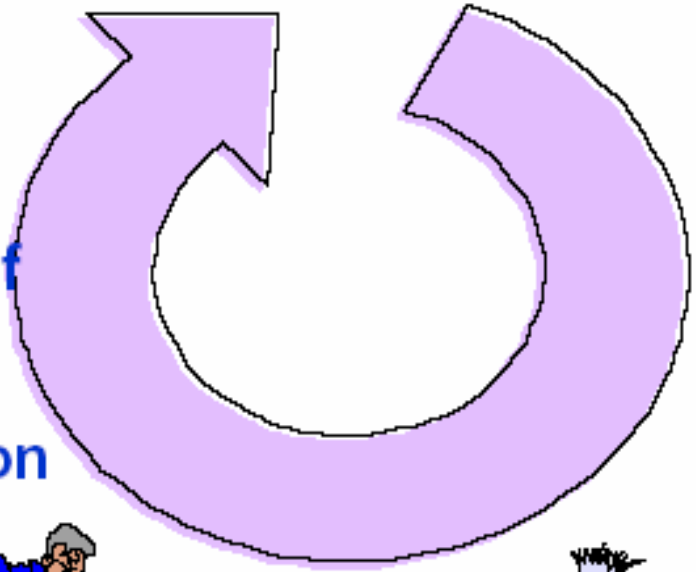


Verification

Design : specify and enter the design intent

Verify:

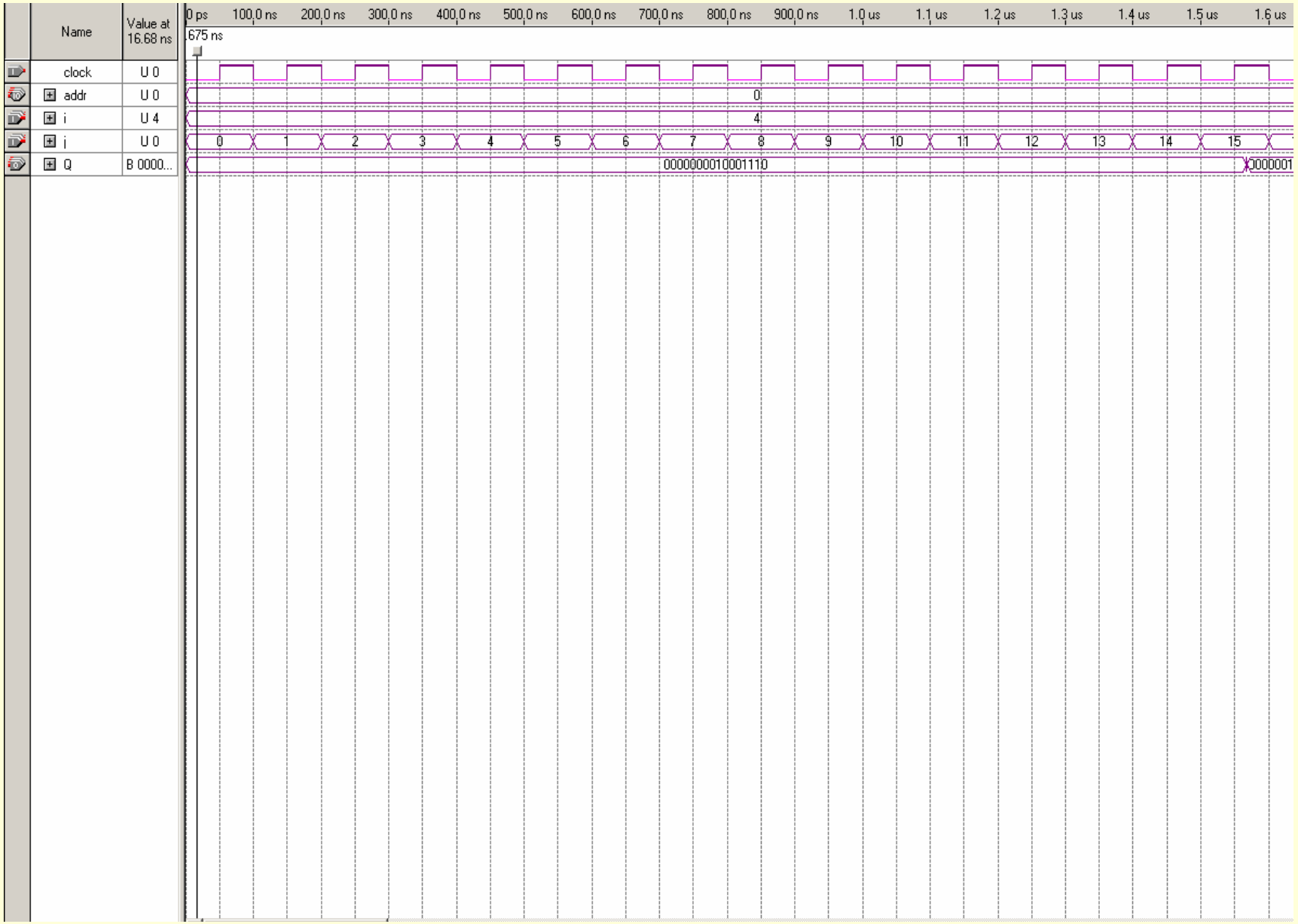
verify the correctness of design and implementation



Implement:

refine the design through all phases

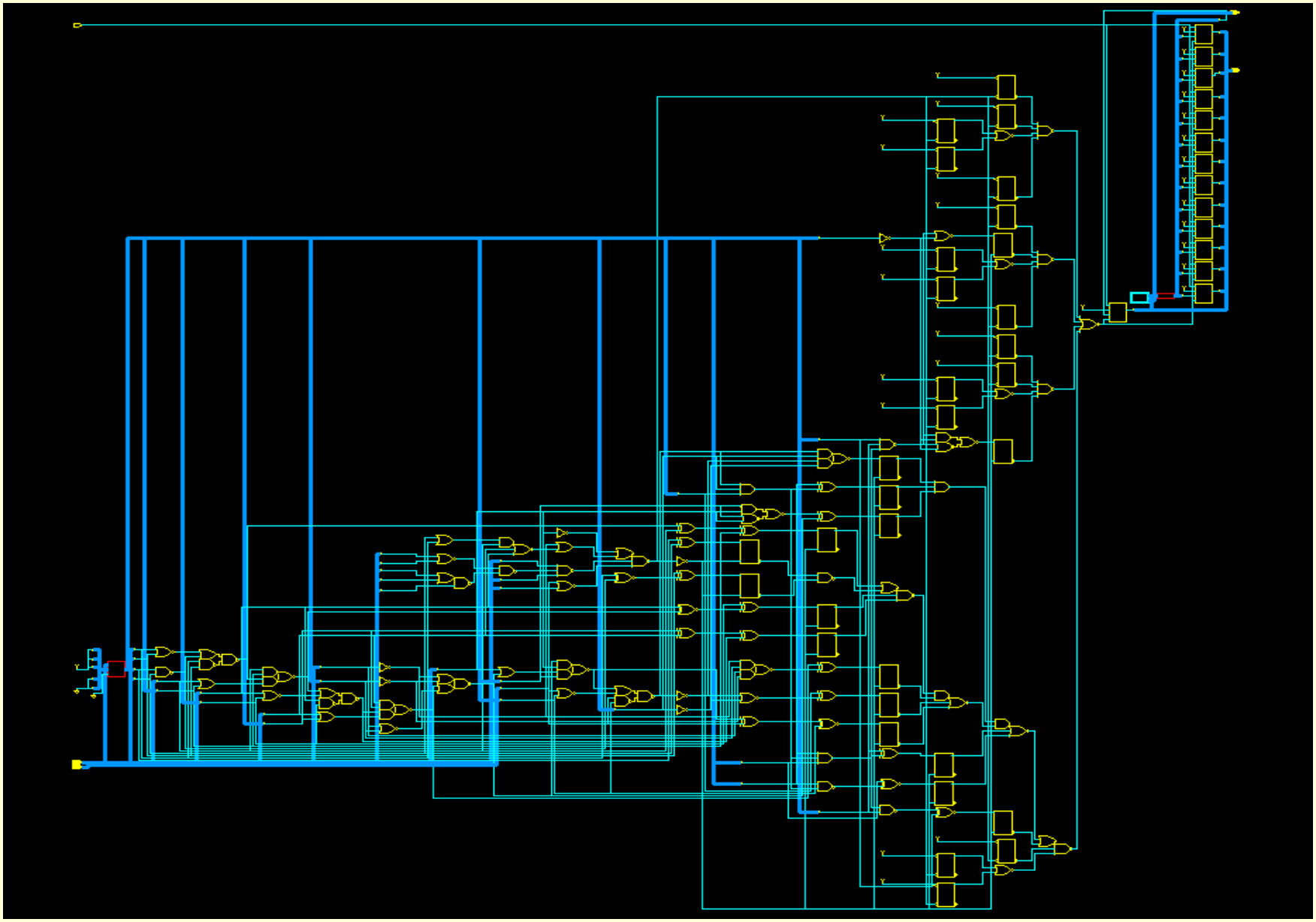
Testbench Waveforms



Synthesis

- Synopsys Design Compiler is used to perform logic synthesis.
- Design is optimized for Area as well as Timing.
- Synopsys Design Analyzer tool is used to obtain gate level representation of the circuit and generate report files for area, timing and power dissipation.

Gate Level Representation of the circuit



Conclusion

- The high level description of the fuzzy rule base is converted to VHDL code.
- To simplify the process we design the fuzzy controller using MATLAB.
- The output of the MATLAB program is stored in the ROM for use in the VHDL code.
- A complete description of the system is assembled in VHDL and is synthesized using VHDL-based logic synthesis.
- Synopsys design compiler is used to perform logic synthesis.