SoC-NTM (T-DNC/NTM-SoC)

QueenField

0. Introduction

0.0. DO-254

0.0.1. HARDWARE PLANNING PROCESS

0.0.1.1. Plan for Hardware Aspects of Certification

0.0.1.2. Hardware Design Plan

0.0.1.3. Hardware Validation Plan

0.0.1.4. Hardware Verification Plan

0.0.1.5. Hardware Configuration Management Plan

0.0.1.6. Hardware Process Assurance Plan

0.0.2. HARDWARE DESIGN PROCESS

0.0.2.1 Requirements Capture Process

0.0.2.2 Conceptual Design Process

0.0.2.3 Detailed Design Process

0.0.2.4 Implementation Process

0.0.2.5 Production Transition Process

0.0.3. VALIDATION AND VERIFICATION PROCESS

0.0.3.1 Validation Process

0.0.3.2 Verification Process

0.0.4. CONFIGURATION MANAGEMENT PROCESS

0.0.5. PROCESS ASSURANCE

0.0.6. CERTIFICATION LIAISON PROCESS

0.0.7. HARDWARE DESIGN LIFECYCLE DATA

0.0.7.1 Certification Authority

0.0.7.2 Certification Reviews

0.0.7.3 Scheduling of Reviews

0.0.8. ADDITIONAL CONSIDERATIONS

0.0.8.1 Previously Developed Hardware

0.0.8.2 Commercial Components Usage

0.1. Best Practices

1. Mechanics

2. Information

2.1. Bit

2.2. Logic Gate

2.2.1. YES/NOT Gate

2.2.2. AND/NAND Gate

2.2.3. OR/NOR Gate

2.2.4. XOR/XNOR Gate

2.3. Combinational Logic

2.3.1. Arithmetic Circuits

2.3.2. Logic Circuits

2.4. Finite State Machine

2.5. Pushdown Automaton

3. Neural Network

3.1. Feedforward Neural Network

3.2. Long Short Term Memory Neural Network

3.3. Transformer Neural Network

4. Turing Machine

4.1. Neural Turing Machine

4.1.1. Feedforward Neural Turing Machine

4.1.2. LSTM Neural Turing Machine

4.1.3. Transformer Neural Turing Machine

4.2. Differentiable Neural Computer

4.2.1. Feedforward Differentiable Neural Computer

4.2.2. LSTM Differentiable Neural Computer

4.2.3. Transformer Differentiable Neural Computer

7.1.1.2. Component diagram

7.1.1.3. Composite diagram

7.1.1.4. Deployment diagram

7.1.1.5. Object diagram

7.1.1.6. Package diagram

7.1.1.7. Profile diagram

7.1.2. Behavioral UML diagrams

7.1.2.1. Activity diagram

7.1.2.2. Communication diagram

7.1.2.3. Interaction diagram

7.1.2.4. Sequence diagram

7.1.2.5. State diagram

7.1.2.6. Timing diagram

7.1.2.7. Use diagram

7.2. Source

7.2.1. Ada Language

7.2.2. C Language

7.3. Model

7.3.1. VHDL

7.3.2. Verilog

7.4. Validation

7.4.1. VHDL

7.4.2. Verilog

7.5. Design

7.5.1. VHDL

7.5.2. Verilog

7.6. Verification

7.6.1. OSVVM-VHDL

7.6.1.1. OSVVM Checker

7.6.1.2. OSVVM Stimulus

7.6.1.3. OSVVM Testbench

7.6.2. UVM-Verilog

7.6.2.1. UVM Agent

7.6.2.2. UVM Driver

7.6.2.3. UVM Enviroment

7.6.2.4. UVM Monitor

7.6.2.5. UVM Scoreboard

7.6.2.6. UVM Sequence

7.6.2.7. UVM Sequencer

7.6.2.8. UVM Subscriber

7.6.2.9. UVM Test

7.6.2.10. UVM Testbench

7.6.2.11. UVM Transaction