Logic gates functions emulation with DSF

Damian Fascio

Working methodology:

Data preloaded into database.

Elements used:

ep (numerical value 0, represents binary number 0).

epX (numerical value 1, represents binary number 1).

Sequence types used:

One-dimensional and two-dimensional sequences.

1. **YES** / **BUFFER** gate:

Truth table:

0|0

1|1

Sequence example #1

step: 1

input: - > NO PRELOAD DATA

database: []

operator: **EMP**

output: -

value: 0

remain: -

=== LOGIC GATE FUNCTION ===

step: 2

input: **ep** > INPUT 0

database: [**ep**]

operator: **ONL**

output: **ep** > OUTPUT 0

value: 0

remain: -

> DELETE DATABASE from step 2

step: 2

input: **epX** > INPUT 1

database: [**epX**]

operator: **ONL**

output: **epX** > OUTPUT 1

value: 1

remain: -

==========

2. **NOT** gate:

Truth table:

0|1

1|0

Sequence example #1

Sequence **A**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep**, **epX** > PRELOAD DATA

database: [**ep**, **epX**]

operator: **RAN**

output: **ep**

value: 0

remain: **epX**

=== LOGIC GATE FUNCTION ===

step: 3

input: **ep** > INPUT 0

database: [**ep**, **epX** | **ep**]

operator: **MAJ**

output: **ep**

value: 0

remain: **epX**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX** (sequence A step 3 remain)

database: [**epX**]

operator: **ONL**

output: **epX** > OUTPUT 1

value: 1

remain: -

> DELETE DATABASE from step 3

step: 3

input: **epX** > INPUT 1

database: [**ep**, **epX** | **epX**]

operator: **MAJ**

output: **epX**

value: 1

remain: **ep**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep** (sequence A step 3 remain)

database: [**ep**]

operator: **ONL**

output: **ep** > OUTPUT 0

value: 0

remain: -

==========

3. **AND** gate:

Truth table:

00|0

01|0

10|0

11|1

Sequence example #1

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep** > PRELOAD DATA

database: [**ep**]

operator: **ONL**

output: **ep**

value: 0

remain: -

=== LOGIC GATE FUNCTION ===

step: 3

input: **ep**, **ep** > INPUT 0, 0

database: [**ep** | **ep**, **ep**]

operator: **ONR**

output: **ep** > OUTPUT 0

value: 0

remain: -

> DELETE DATABASE from step 3

step: 3

input: **ep**, **epX** > INPUT 0, 1

database: [**ep** | **ep**, **epX**]

operator: **MAJ**

output: **ep** > OUTPUT 0

value: 0

remain: **epX**

> DELETE DATABASE from step 3

step: 3

input: **epX**, **ep** > INPUT 1, 0

database: [**ep** | **epX**, **ep**]

operator: **MAJ**

output: **ep** > OUTPUT 0

value: 0

remain: **epX**

> DELETE DATABASE from step 3

step: 3

input: **epX**, **epX** > INPUT 1, 1

database: [**ep** | **epX**, **epX**]

operator: **MAJ**

output: **epX** > OUTPUT 1

value: 1

remain: **ep**

==========

4. **OR** gate:

Truth table:

00|0

01|1

10|1

11|1

Sequence example #1

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX** > PRELOAD DATA

database: [**epX**]

operator: **ONL**

output: **epX**

value: 1

remain: -

=== LOGIC GATE FUNCTION ===

step: 3

input: **ep**, **ep** > INPUT 0, 0

database: [**epX** | **ep**, **ep**]

operator: **MAJ**

output: **ep** > OUTPUT 0

value: 0

remain: **epX**

> DELETE DATABASE from step 3

step: 3

input: **ep**, **epX** > INPUT 0, 1

database: [**epX** | **ep**, **epX**]

operator: **MAJ**

output: **epX** > OUTPUT 1

value: 1

remain: **ep**

> DELETE DATABASE from step 3

step: 3

input: **epX**, **ep** > INPUT 1, 0

database: [**epX** | **epX**, **ep**]

operator: **MAJ**

output: **epX** > OUTPUT 1

value: 1

remain: **ep**

> DELETE DATABASE from step 3

step: 3

input: **epX**, **epX** > INPUT 1, 1

database: [**epX** | **epX**, **epX**]

operator: **ONR**

output: **epX** > OUTPUT 1

value: 1

remain: -

==========

5. **NAND** gate:

Truth table:

00|1

01|1

10|1

11|0

Sequence example #1

Sequence **A**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep**, **ep**, **epX** > PRELOAD DATA

database: [**ep**, **ep**, **epX**]

operator: **MAJ**

output: **ep**

value: 0

remain: **epX**

=== LOGIC GATE FUNCTION ===

step: 3

input: **ep**, **ep** > INPUT 0, 0

database: [**ep**, **ep**, **epX** | **ep**, **ep**]

operator: **MAJ**

output: **ep**

value: 0

remain: **epX**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX** (sequence A step 3 remain)

database: [**epX**]

operator: **ONL**

output: **epX** > OUTPUT 1

value: 1

remain: -

> DELETE DATABASE from step 3

step: 3

input: **ep**, **epX** > INPUT 0, 1

database: [**ep**, **ep**, **epX** | **ep**, **epX**]

operator: **MAJ**

output: **ep**

value: 0

remain: **epX**, **epX**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX**, **epX** (sequence A step 3 remain)

database: [**epX**, **epX**]

operator: **ONR**

output: **epX** > OUTPUT 1

value: 1

remain: -

> DELETE DATABASE from step 3

step: 3

input: **epX**, **ep** > INPUT 1, 0

database: [**ep**, **ep**, **epX** | **epX**, **ep**]

operator: **MAJ**

output: **ep**

value: 0

remain: **epX**, **epX**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX**, **epX** (sequence A step 3 remain)

database: [**epX**, **epX**]

operator: **ONR**

output: **epX** > OUTPUT 1

value: 1

remain: -

> DELETE DATABASE from step 3

step: 3

input: **epX**, **epX** > INPUT 1, 1

database: [**ep**, **ep**, **epX** | **epX**, **epX**]

operator: **MAJ**

output: **epX**

value: 1

remain: **ep**, **ep**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep**, **ep** (sequence A step 3 remain)

database: [**ep**, **ep**]

operator: **ONR**

output: **ep** > OUTPUT 0

value: 0

remain: -

==========

6. **NOR** gate:

Truth table:

00|1

01|0

10|0

11|0

Sequence example #1

Sequence **A**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX** > PRELOAD DATA

database: [**epX**]

operator: **ONL**

output: **epX**

value: 1

remain: -

=== LOGIC GATE FUNCTION ===

step: 3

input: **ep**, **ep** > INPUT 0, 0

database: [**epX** | **ep**, **ep**]

operator: **MAJ**

output: **ep**

value: 0

remain: **epX**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX** (sequence A step 3 remain)

database: [**epX**]

operator: **ONL**

output: **epX** > OUTPUT 1

value: 1

remain: -

> DELETE DATABASE from step 3

step: 3

input: **ep**, **epX** > INPUT 0, 1

database: [**epX** | **ep**, **epX**]

operator: **MAJ**

output: **epX**

value: 1

remain: **ep**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep** (sequence A step 3 remain)

database: [**ep**]

operator: **ONL**

output: **ep** > OUTPUT 0

value: 0

remain: -

> DELETE DATABASE from step 3

step: 3

input: **epX**, **ep** > INPUT 1, 0

database: [**epX** | **epX**, **ep**]

operator: **MAJ**

output: **epX**

value: 1

remain: **ep**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep** (sequence A step 3 remain)

database: [**ep**]

operator: **ONL**

output: **ep** > OUTPUT 0

value: 0

remain: -

> DELETE DATABASE from step 3

step: 3

input: **epX**, **epX** > INPUT 1, 1

database: [**epX** | **epX**, **epX**]

operator: **ONR**

output: **epX**

value: 1

remain: -

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **-** (sequence A step 3 remain)

database: []

operator: **EMP**

output: **-** > OUTPUT 0

value: 0

remain: -

==========

7. **XOR** gate:

Truth table:

00|0

01|1

10|1

11|0

Sequence example #1

Sequence **A**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep** > PRELOAD DATA

database: [**ep**]

operator: **ONL**

output: **ep**

value: 0

remain: -

=== LOGIC GATE FUNCTION ===

step: 3

input: **ep**, **ep** > INPUT 0, 0

database: [**ep** | **ep**, **ep**]

operator: **ONR**

output: **ep**

value: 0

remain: **-**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: - (sequence A step 3 remain)

database: []

operator: **EMP**

output: - > OUTPUT 0

value: 0

remain: -

> DELETE DATABASE from step 3

step: 3

input: **ep**, **epX** > INPUT 0, 1

database: [**ep** | **ep**, **epX**]

operator: **MAJ**

output: **ep**

value: 0

remain: **epX**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX** (sequence A step 3 remain)

database: [**epX**]

operator: **ONL**

output: **epX** > OUTPUT 1

value: 1

remain: -

> DELETE DATABASE from step 3

step: 3

input: **epX**, **ep** > INPUT 1, 0

database: [**ep** | **epX**, **ep**]

operator: **MAJ**

output: **ep**

value: 0

remain: **epX**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX** (sequence A step 3 remain)

database: [**epX**]

operator: **ONL**

output: **epX** > OUTPUT 1

value: 1

remain: -

> DELETE DATABASE from step 3

step: 3

input: **epX**, **epX** > INPUT 1, 1

database: [**ep** | **epX**, **epX**]

operator: **MAJ**

output: **epX**

value: 1

remain: **ep**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep** (sequence A step 3 remain)

database: [**ep**]

operator: **ONL**

output: **ep** > OUTPUT 0

value: 0

remain: -

==========

8. **XNOR** gate:

Truth table:

00|1

01|0

10|0

11|1

Sequence example #1

Sequence **A**:

Arbitrary strength: **epX$_2$** = 1,1 **ep** = 1,1

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX$_2$** > PRELOAD DATA

database: [**epX$_2$**]

operator: **ONL**

output: **epX$_2$**

value: 1

remain: -

=== LOGIC GATE FUNCTION ===

step: 3

input: **ep**, **ep** > INPUT 0, 0

database: [**epX$_2$** | **ep**, **ep**]

operator: **MAJ**

output: **ep**

value: 0

remain: **epX$_2$**

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX$_2$** (sequence A step 3 remain)

database: [**epX$_2$**]

operator: **ONL**

output: **epX$_2$** > OUTPUT 1

value: 1

remain: -

> DELETE DATABASE from step 3

step: 3

input: **ep**, **epX** > INPUT 0, 1

database: [**epX$_2$** | **ep**, **epX**]

operator: (**MAJ**) **STR**

output: **epX$_2$**

value: 1

remain: **ep**, **epX** <---------- (alternative remain)

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **ep**, **epX** (sequence A step 3 remain)

database: [**ep**, **epX**]

operator: **STR**

output: **ep** > OUTPUT 0

value: 0

remain: **epX**

> DELETE DATABASE from step 3

step: 3

input: **epX**, **ep** > INPUT 1, 0

database: [**epX₂** | **epX**, **ep**]

operator: (**MAJ**) **STR**

output: **epX$_2$**

value: 1

remain: **epX**, **ep** <---------- (alternative remain)

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX**, **ep** (sequence A step 3 remain)

database: [**epX**, **ep**]

operator: **STR**

output: **ep** > OUTPUT 0

value: 0

remain: **epX**

> DELETE DATABASE from step 3

step: 3

input: **epX**, **epX** > INPUT 1, 1

database: [**epX₂** | **epX**, **epX**]

operator: (**ONR**) **STR**

output: **epX₂**

value: 1

remain: **epX**, **epX** <---------- (alternative remain)

Sequence **B**:

step: 1

input: -

database: []

operator: **EMP**

output: -

value: 0

remain: -

step: 2

input: **epX**, **epX** (sequence A step 3 remain)

database: [**epX**, **epX**]

operator: **ONR**

output: **epX** > OUTPUT 1

value: 1

remain: -

Córdoba, Argentina, October, 25, 2024.