Flip-flop Evaluator

User Manual

Caquilala, Cedric

Famadico, Nichol John

Overview

This program is a freeware that evaluates the next state and outputs of flip-flop circuits. It can handle four different flip-flops: D, T, RS, and JK. It is capable of evaluating circuits with up to two flip-flops of the same kind, two inputs, and one output. It can also simplify complicated equations with operators including AND, OR, and XOR, and even follows precedence rules. It will display the respective state table which includes the present states, input(s), flip-flop input(s), output, and the next states.

Getting Started

This program is console-based. Users can submit their input by typing and pressing the return key.

Inputting Data

1. Initialization

The first input that will be asked is the type of flip-flop. The program supports four different flip-flops: D, T, JK, and RS. Input the type of flip-flop that will be used for the program.

Input type of flip-flop (D, T, JK, RS):

JK

The next input that will be asked is the number of flip-flops. The program supports up to two flip-flops. However, these two flip-flops must be of the same kind. These two flip-flops will be represented by **A** and **B**.

Input number of flip-flops:

2

Next, input the number of input variables that will be used. Up to two input variables may be declared. These will be represented by **x** and **y**.

Input number of input variables:

2

Then, input the number of output variables. Only one or no output variable can be used. For the latter, simply type ‘**0**’.

Input number of output variables:

1

1. Equations

After declaring the number of each part of the circuit, the equations for each of those will now be asked for. For each these inputs that require an equation, simply typing the equation as we normally do is sufficient. The program is capable of understanding complex equations that may even include parentheses as long as there are no syntax errors.

The following operators and operands can be used:

|  |  |  |  |
| --- | --- | --- | --- |
| Operators | | Operands | |
| + | OR | A / B | Flip-flop states |
| \* | AND | JA / KA  RA / SA  DA / TA | Flip-flop inputs |
| @ | XOR | x / y | Inputs  (x & y if there are two inputs) |
| ‘ | NOT | y / z | Outputs  (the output is y if there is one input; z if there are two.) |

**If an output variable will be used**, the program will ask for its equation. The output variable will be represented by **y** if there is one input, and **z** if there are two.

Input the equation for output variable

z=Axy+Bx'y'

The user is then prompted to input the equations for the flip-flops. This will vary depending on the type of flips-flops and the number. D and T flip-flops will ask for one equation each flip-flop, while RS and JK will ask for two for each.

Input the equation for flip-flop A

JA=Bx+By

Input the equation for flip-flop A

KA=B'xy'

Input the equation for flip-flop B

JB=A'x

Input the equation for flip-flop B

KB=A+xy'

Output

After the equations, the program will begin the evaluation. It will display all the user’s input followed by the state table. The state table is divded into up to four parts: The present state & inputs, the flip-flop inputs, the next state, and the output. The program will display all possible cases given the inputs and the flip-flop states in a table.

Type: JK

No. of Flip Flops: 2

No. of input variables: 2

No. of output variables: 1

EQ of output variable 1: z=Axy+Bx'y'

EQ of flip-flop 1: JA=Bx+By

EQ of flip-flop 2: KA=B'xy'

Table: JK

A B x y | JA KA JB KB | A B | z

0 0 0 0 | 0 0 0 0 | 0 0 | 0

0 0 0 1 | 0 0 0 0 | 0 0 | 0

0 0 1 0 | 0 1 1 1 | 0 1 | 0

0 0 1 1 | 0 0 1 0 | 0 1 | 0

0 1 0 0 | 0 0 0 0 | 0 1 | 1

0 1 0 1 | 1 0 0 0 | 1 1 | 0

0 1 1 0 | 1 0 1 1 | 1 0 | 0

0 1 1 1 | 1 0 1 0 | 1 1 | 0

1 0 0 0 | 0 0 0 1 | 1 0 | 0

1 0 0 1 | 0 0 0 1 | 1 0 | 0

1 0 1 0 | 0 1 0 1 | 0 0 | 0

1 0 1 1 | 0 0 0 1 | 1 0 | 1

1 1 0 0 | 0 0 0 1 | 1 0 | 1

1 1 0 1 | 1 0 0 1 | 1 0 | 0

1 1 1 0 | 1 0 0 1 | 1 0 | 0

1 1 1 1 | 1 0 0 1 | 1 0 | 1

Example

For example, we want to get the state table for a circuit with one T flip-flop, one input, and one output. The output follows the equation , while the flip-flops follow the equations and respectively. We can use the program as follows: (green represent the user’s input.)

Input type of flip-flop (D, T, JK, RS):

T

Input number of flip-flops:

2

Input number of input variables:

1

Input number of output variables:

1

Input the equation for output variable

y=(A+B)x'

Input the equation for flip-flop T

A=x

Input the equation for flip-flop T

B=A'x

Type: T

No. of Flip Flops: 2

No. of input variables: 1

No. of output variables: 1

EQ of output variable 1: y=(A+B)x'

EQ of flip-flop 1: A=x

EQ of flip-flop 2: B=A'x

Table: T

A B x | TA TB | A B | y

0 0 0 | 0 0 | 0 0 | 0

0 0 1 | 1 1 | 1 1 | 0

0 1 0 | 0 0 | 0 1 | 1

0 1 1 | 1 1 | 1 0 | 0

1 0 0 | 0 0 | 1 0 | 1

1 0 1 | 1 0 | 0 0 | 0

1 1 0 | 0 0 | 1 1 | 1

1 1 1 | 1 0 | 0 1 | 0

------- end ------