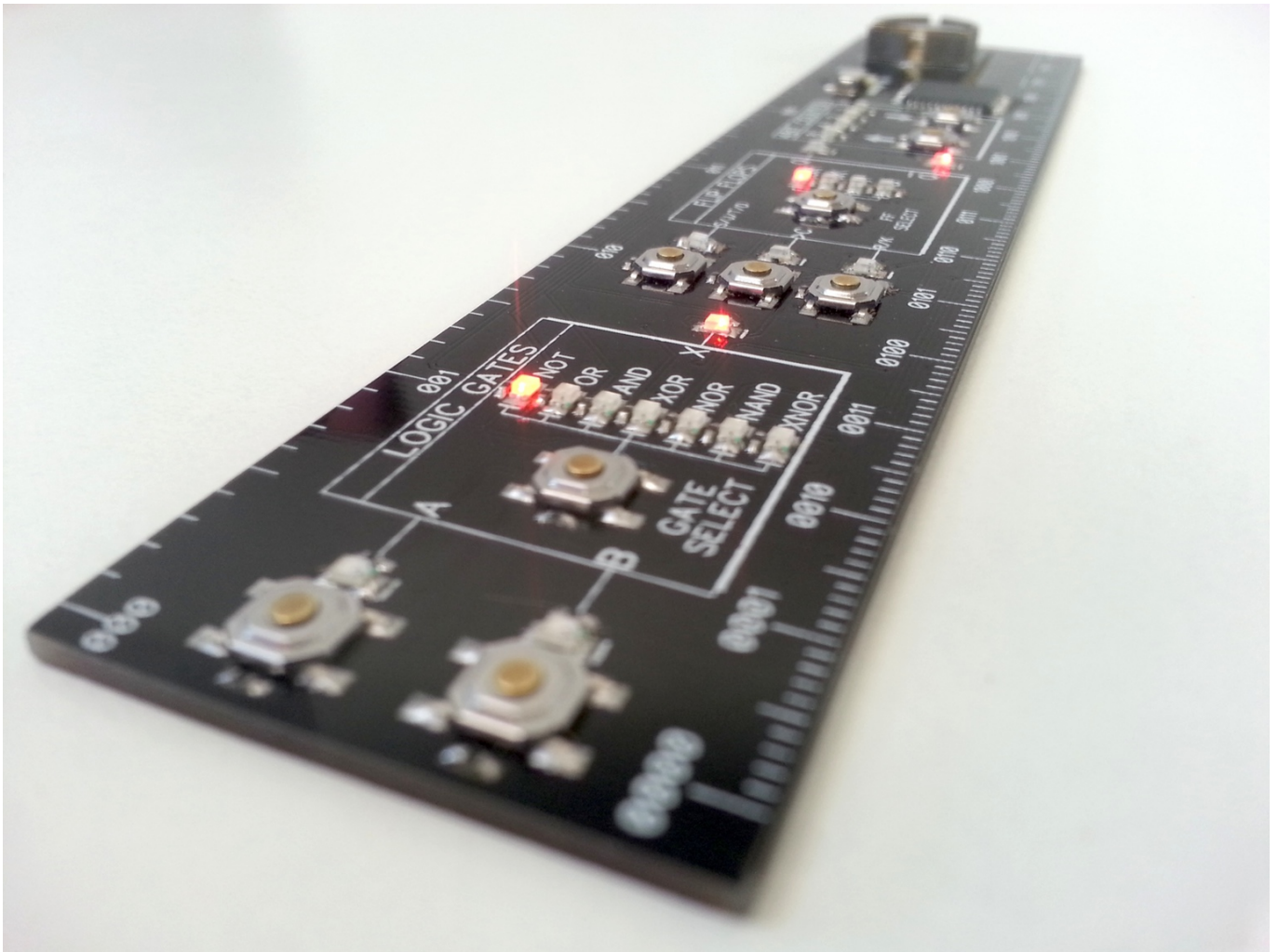


# DigiRule User Manual



## About

The DigiRule is a 15cm (6”) ruler made out of a 1.6mm circuit board. The DigiRule was designed to be a geeky ruler with an inbuilt microcontroller, LED’s, buttons and various other components to enable the user to simulate various digital circuits.

## Functions

1. Logic Gates
  - 1.1. Inverter (Not Gate)
  - 1.2. OR Gate
  - 1.3. AND Gate
  - 1.4. XOR Gate
  - 1.5. NOR Gate
  - 1.6. NAND Gate
  - 1.7. XNOR Gate
2. Flip Flops
  - 2.1. SR Flip Flop (aka RS Flip Flop)
  - 2.2. JK Flip Flop
  - 2.3. Toggle Flip Flop
  - 2.4. Data Flip Flop
3. A four bit up/down binary counter

## Hidden Mode

The DigiRule also has a hidden mode of operation with nine modes of operation. These modes are as follows:

1. Mode 1 – standard mode (the same mode that you would get by simply switching the DigiRule on as normal)
2. Mode 2 – standard mode with a gray code counter in place of the binary counter
3. Mode 3 – sparkly LED’s
4. Mode 4 – seven bit serial in shift register
5. Mode 5 – LED test
6. Mode 6 – Persistence of vision display
7. Mode 7 – binary dice
8. Mode 8 – catch the XOR Gate binary game
9. Mode 9 – 4 bit logic unit.

## How To Use The DigiRule – Standard Mode.

To enter the standard mode of operation, simply turn the DigiRule on with the slide switch without pressing any other buttons.

## Using the Logic Gates

The gate select button enables you to cycle through the seven different logic gates. The truth table on the rear of the DigiRule shows the X output for a given input on the ‘A’ and ‘B’ buttons. For example, if you were to select the AND Gate you will notice that the only combination of inputs for ‘A’ and ‘B’ to give a logic 1 at output ‘X’ is the combination of 1 1.

1. Select your desired logic gate with the gate select button.
2. Input your combination of 1's and 0's on the Logic Gate's 'A' and 'B' inputs.
3. The output LED 'X' will light up whenever the output is true for the given input (see the reverse side of the DigiRule for the logic gate truth tables).
4. Selecting a new logic gate will clear your 'A' and 'B' inputs.

### **Using the Flip Flops**

1. Select your desired flip flop with the flip flop select button.
2. Input your combination of 1's and 0's on the flip flops 'S/J/T/D' and 'R/K' inputs.
3. The outputs Q and notQ will only be affected by the inputs when the clock has a positive edge transition applied to it (when the button is pressed changing a logic 0 to a logic 1) – see reverse side of the DigiRule for the flip flop truth tables.
4. Selecting a new flip flop will clear your inputs and reset the output.

### **Using the Four bit Up / Down Binary Counter**

1. Press the up arrow to increment the binary count by 1. The counter will loop back around to  $0000_2$  if the counter tries to count past  $1111_2$ .
2. Press the down arrow to decrement the binary count by 1. The counter will loop back around to  $1111_2$  if the counter tries to count below  $0000_2$ .

### **How to Use the DigiRule – Hidden Mode**

To enter the hidden mode of operation, hold down the logic gates 'A' button and then turn the DigiRule on. The user can then cycle through the various modes from  $0001_2$  to  $1001_2$  using the 4bit counter up/down arrows. The logic gates 'B' button is then used to select a desired mode.

#### **Using Mode 1 ( $0001_2$ )**

This mode is identical to the standard mode of operation.

#### **Using Mode 2 ( $0010_2$ )**

This mode is identical to mode 1 except that the 4bit up/down counter now functions as a 4bit gray code counter.

#### **Using Mode 3 ( $0011_2$ )**

This mode presents the user with some random sparkling LED's. The user can then increase and decrease the speed at which the LED's sparkle by using the 4bit binary up/down arrow buttons.

#### **Using Mode 4 ( $0100_2$ )**

This mode is a seven bit serial in shift register. The contents of the register are shown on the seven logic gate LED's.

1. The serial data input is logic gate input 'A'
2. The positive edge clock is the gate select button
3. The master reset is logic gate input 'B'

### **Using Mode 5 (0101<sub>2</sub>)**

This mode simply turns on LED's to test each of the for functionality.

### **Using Mode 6 (0110<sub>2</sub>)**

This mode is a simple POV (persistence of vision) display. The user needs to either shake the DigiRule left and right really fast to separate the individual lines of information coming from the logic gates LED's or alternatively you can keep the DigiRule still and shake your head.

1. Pressing the logic gate input 'B' button will cycle through the three different images.

### **Using Mode 7 (0111<sub>2</sub>)**

This mode is a random dice that gives the user a random binary number between 1 and 6 (0001<sub>2</sub>) and (0110<sub>2</sub>).

2. Pressing the logic gate input 'B' button will 'roll' the dice.

### **Using Mode 8 (1000<sub>2</sub>)**

This mode is a simple game whereby the user needs to press the logic gate select 'B' button while the XOR LED is lit. if successful, the binary counter will count up by one and the speed of the logic gate LED's will increase. The idea is to get to the last level which is level 15 (1111<sub>2</sub>). If the button is pressed while any other LED is lit, all points are lost and the game returns to the starting speed.

### **Using Mode 9 (1001<sub>2</sub>)**

This mode is a 4-bit logic unit. The user can select the desired logic function with the logic gates select button. Then a combination of 1's and 0's can be entered via the 4-bit up/down counter using the up/down arrows. This data can then be saved to the four bit 'A' register by pressing logic gate input 'A' or it can be saved to the four bit 'B' register by pressing logic gate input 'B'. The answer to the selected logic function for the 'A' and 'B' registers will be shown on the flip flop select LED's.