

Binary Clock and Stopwatch using Arduino

INTRODUCTION

This documentation is an implementation on Binary Clock and Stopwatch using Arduino. It will be operated using buttons. The clock will be having a layout hh:mm:ss. It will be having pausing and resuming options. The clock will be displayed on the LED screen.

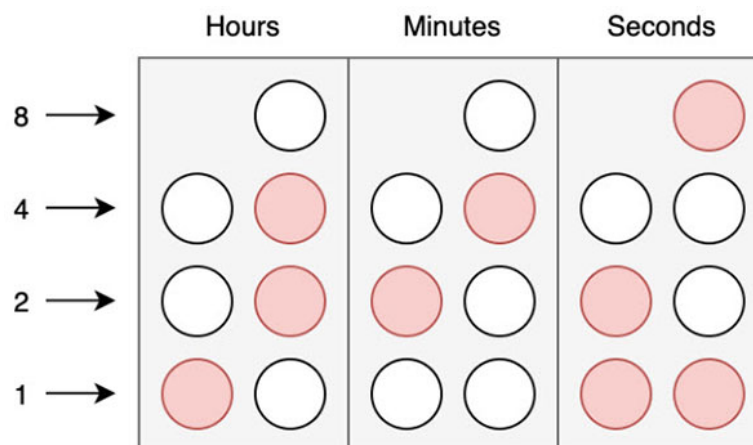
MATERIALS

- Arduino UNO - 1
- Red 5mm LED - 10
- Yellow 5mm LED - 10
- Green 5mm LED - 10
- NPN Transistor - 8
- 680 Ohm Resistor - 8
- 150 Ohm Resistor - 6
- Buttons - 5

SETUP OF DEVICE

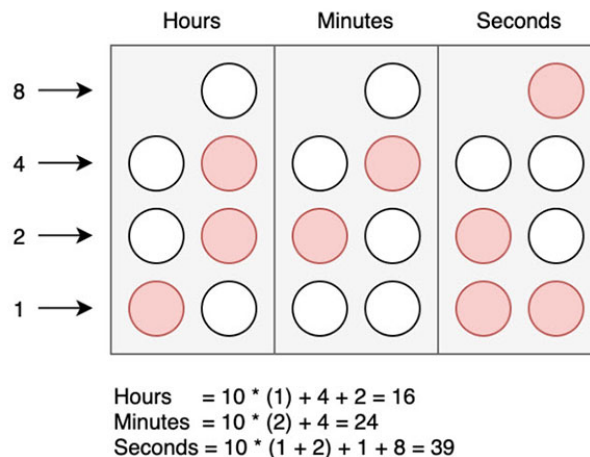
How to Read a Binary Clock:

This project used a 6x4 LED matrix that is common in larger binary clocks such as wall clocks. Small watch-style devices may use slightly different arrangements to save space. In any case, in this project, the LEDs are arranged as follows:



This picture shows an example of a binary clock. The red circles represent lit LEDs.

As we can see, we can divide the 6x4 matrix into three equal parts, each containing two rows of LEDs. The hours are displayed on the left, the minutes are displayed in the middle two columns, and the current seconds are displayed in the last two columns. The left column of each section represents the tens digit, and the second column represents the ones digit of the last digit. Similarly, each row represents a power of two. To read the current time, look at each of the three segments that start with the time. Add the value for each lit LED in the left row of the time segment and multiply the result by 10. Then add the value of the LED in the second column to the result. Repeat this process for minutes and seconds:

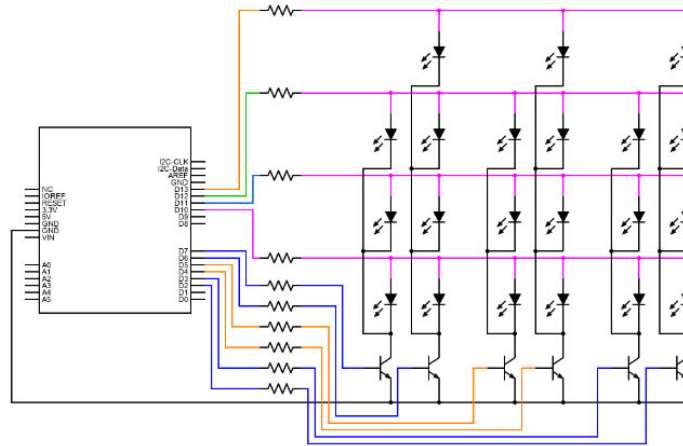


Use the provided formulas to determine the time shown on the LED matrix.

So, in this example, the displayed time is 16:24:39, or four PM, 24 minutes, and 39 seconds. Most binary clocks use the 24-hour format, sometimes also referred to as military time. However, you can simplify your design and omit two LEDs if you decide to build one that only uses the 12-hour format.

The Schematic Diagram:

As mentioned earlier, this project uses a custom 6x4 LED matrix to display the time. Note that the first column of each block does not need to indicate an 8, so there is no LED to indicate an 8. The following scheme describes how to connect the LEDs in the matrix.



This image shows the schematic diagram of this project

Connect the anodes of all LEDs in a row. Then connect the cathodes of all the LEDs to the column. Then connect each column to the collector pin of the NPN transistor. We need to add a resistor to the base of each transistor. Also note that there is only one current limiting resistor per row. In this configuration, each transistor controls one column in the matrix. This means that Arduino scans the columns in the matrix and only one column is active at a time.

However, Arduino can activate multiple lines at the same time. Therefore, adding a resistor to the lead will connect the resistor to each active LED. Otherwise, multiple LEDs will use a single current limiting resistor and the brightness of consecutive LEDs will vary depending on the number of LEDs lit.

REFERENCE

<https://www.digikey.com/en/maker/projects/build-an-arduino-based-binary-stopwatch-with-tinkercad/203054c5ba3a4605a9d556acec7b165b>