AriksWorld - Clock No. 8

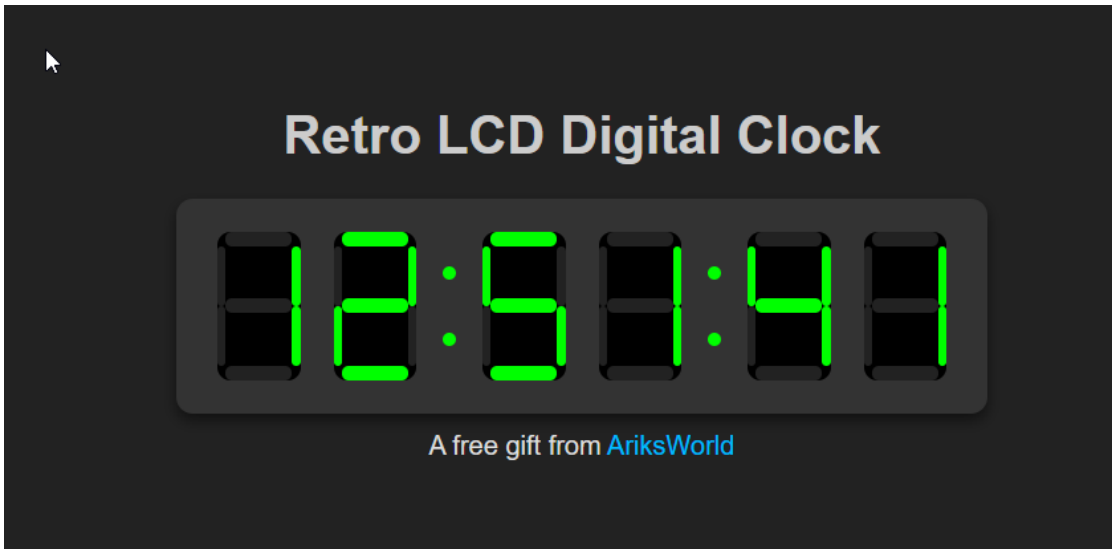
Retro LCD Digital Clock

Thank you for visiting AriksWorld and using my free clocks! This first clock is a simple and easy-to-use **retro LCD digital clock**, designed for both functionality and elegance.

I love clocks. To me, they are more than just tools for telling time - they are beautiful design elements that can enhance any space, whether physical or digital. Clocks bring a sense of order and rhythm to our lives, subtly reminding us of the passage of time while adding a touch of sophistication.

In web design, clocks serve both aesthetic and practical purposes. They create an engaging, interactive experience for users, making a website feel more dynamic and alive. A clock can also tell a story about the website and its owners. For example, a binary clock might reflect a preference for mathematics or programming. A well-designed clock can reinforce a brand's identity, provide real-time updates, and improve usability by helping users keep track of time while browsing. Whether minimalist, artistic, or highly functional, clocks add a timeless appeal to any digital interface.

Developing all these clocks wasn't an easy task. At times, I had to switch from CSS-based design to SVG design. However, the tools used should not affect your experience with each clock. I truly hope you enjoy every single one!



How to Use This Clock

You can use this clock on your computer by running the core HTML file in your browser. You can also add it to your website. However, integrating this clock into a website requires some understanding of HTML, CSS, and JavaScript. If you're not familiar with these, you can seek help from AI tools or online resources. You can also use an `<iframe>` to embed the clock into your existing HTML design. In my opinion, starting with this ready-made clock is a better approach than building one from scratch.

Here's how to get started:

1. **Try It Out First** – Copy the core HTML code into a new file on your computer (e.g., `clock.html`) using a text editor like Notepad++. Then, open the file in your browser to see how the clock works. If all you need is a clock for personal use on your computer, you can skip ahead to the next section.

2. **Embed with an iframe (Optional and Easy!) -** If you're looking for the easiest way to display this clock on your website without editing your existing HTML too much, using an iframe is a great solution. It allows you to embed the clock just like you would embed a YouTube video.

   Here's how to do it:

   1. First, upload the clock HTML file (e.g., clock.html) to your web server or hosting platform.
   2. Then, copy and paste the following iframe code into your webpage where you want the clock to appear:

```
<iframe src="clock.html" width="300" height="120" style="border:
none;"></iframe>
```

   Note:

   - Adjust the width and height values as needed to fit your design.
   - Make sure the `src` attribute correctly points to the location of your uploaded clock file.
   - If you're using an external service (like GitHub Pages or another file host), replace `clock.html` with the full URL.

This method requires **zero CSS or JavaScript editing**, making it perfect for quick use or adding the clock to platforms with limited design access. Once you've made your adjustments, save the file, upload it to your server, and enjoy your customized clock!

3. **Integrate It into Your Website** – If you like how the clock looks, you can:
    - **Embed it into an existing webpage** by copying the relevant `<div>` and JavaScript sections into your website's HTML file. Be mindful of CSS styles to avoid conflicts with your existing design.
    - **Use it as a foundation for a new webpage** by building around this structure.

4. **Adjust the Placement** – You'll likely want to position the clock differently to fit your website's layout. This can be done using CSS styles such as `position`, `margin`, or `flexbox`.

5. **Customize the Look** – You might want to tweak the design elements, including:
    - **Background color** – Match it with your website's theme.
    - **Clock hands (if applicable)** – Modify their color, thickness, or style.
    - **Clock size** – Scale it up or down based on your needs.

6. **Clean Up the Code** – If you're adding this clock to an existing site, remove any redundant or conflicting code, such as extra `<head>` sections or duplicate CSS rules.

Here is the HTML core code:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8" />

  <title>AriksWorld - Retro LCD Digital Clock</title>

  <style>

    /* Basic Reset */

    * {

      margin: 0;

      padding: 0;
```

```
  box-sizing: border-box;
}

body {
  font-family: Arial, sans-serif;
  background: #222; /* Dark background for a retro vibe */
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  min-height: 100vh;
  color: #fff;
}

h1 {
  margin-bottom: 20px;
  color: #ccc;
}

/* Container for the entire clock display */
.clock-container {
  display: flex;
  align-items: center;
  justify-content: center;
  gap: 10px;
```

```
  background: #333;
  padding: 20px;
  border-radius: 10px;
  box-shadow: 0 4px 8px rgba(0,0,0,0.5);
}

/* Each digit is a 7-segment display inside a .digit wrapper */
.digit {
  position: relative;
  width: 50px;
  height: 90px;
  background: #000;
  margin: 0 5px;
  border-radius: 8px;
  display: flex;
  justify-content: center;
  align-items: center;
}

/*
  We'll name the segments a, b, c, d, e, f, g.
  Each is an absolutely positioned rectangle,
  which we'll turn "on" or "off" via a CSS class.
*/
.segment {
```

```
  position: absolute;
  background-color: #222; /* Off color */
  transition: background-color 0.2s;
}
.on {
  background-color: #0f0; /* ON color (greenish) */
}

/* Segment positions:
    a
  f   b
  f   b
    g
  e   c
  e   c
    d
*/
.segment.a {
  top: 0;
  left: 10%;
  width: 80%;
  height: 10%;
  border-radius: 4px;
}
.segment.b {
```

```
  top: 10%;
  right: 0%;
  width: 10%;
  height: 40%;
  border-radius: 4px;
}
.segment.c {
  bottom: 10%;
  right: 0%;
  width: 10%;
  height: 40%;
  border-radius: 4px;
}
.segment.d {
  bottom: 0;
  left: 10%;
  width: 80%;
  height: 10%;
  border-radius: 4px;
}
.segment.e {
  bottom: 10%;
  left: 0;
  width: 10%;
  height: 40%;
```

```
  border-radius: 4px;
}
.segment.f {
  top: 10%;
  left: 0;
  width: 10%;
  height: 40%;
  border-radius: 4px;
}
.segment.g {
  top: 45%;
  left: 10%;
  width: 80%;
  height: 10%;
  border-radius: 4px;
}

/* The colon (:) between HH:MM and MM:SS */
.colon {
  width: 10px;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: space-around;
  height: 80px;
```

```
    }
    .dot {
      width: 8px;
      height: 8px;
      background: #0f0;
      border-radius: 50%;
    }

    /* Footer note */
    .note {
      font-size: 1rem;
      color: #ddd;
      margin-top: 10px;
    }
    .note a {
      color: #0bf;
      text-decoration: none;
    }
    .note a:hover {
      text-decoration: underline;
    }
  </style>
</head>
<body>
```

```
<h1>Retro LCD Digital Clock</h1>

<!-- The clock display: HH : MM : SS -->
<div class="clock-container" aria-live="polite">
  <!-- Hour Tens -->
  <div class="digit" id="digitH1">
    <div class="segment a"></div>
    <div class="segment b"></div>
    <div class="segment c"></div>
    <div class="segment d"></div>
    <div class="segment e"></div>
    <div class="segment f"></div>
    <div class="segment g"></div>
  </div>

  <!-- Hour Ones -->
  <div class="digit" id="digitH2">
    <div class="segment a"></div>
    <div class="segment b"></div>
    <div class="segment c"></div>
    <div class="segment d"></div>
    <div class="segment e"></div>
    <div class="segment f"></div>
    <div class="segment g"></div>
  </div>
```

```
<!-- Colon -->
<div class="colon">
  <div class="dot"></div>
  <div class="dot"></div>
</div>

<!-- Minute Tens -->
<div class="digit" id="digitM1">
  <div class="segment a"></div>
  <div class="segment b"></div>
  <div class="segment c"></div>
  <div class="segment d"></div>
  <div class="segment e"></div>
  <div class="segment f"></div>
  <div class="segment g"></div>
</div>

<!-- Minute Ones -->
<div class="digit" id="digitM2">
  <div class="segment a"></div>
  <div class="segment b"></div>
  <div class="segment c"></div>
  <div class="segment d"></div>
  <div class="segment e"></div>
```

```
  <div class="segment f"></div>
  <div class="segment g"></div>
</div>

<!-- Colon -->
<div class="colon">
  <div class="dot"></div>
  <div class="dot"></div>
</div>
<!-- Second Tens -->
<div class="digit" id="digitS1">
  <div class="segment a"></div>
  <div class="segment b"></div>
  <div class="segment c"></div>
  <div class="segment d"></div>
  <div class="segment e"></div>
  <div class="segment f"></div>
  <div class="segment g"></div>
</div>
<!-- Second Ones -->
<div class="digit" id="digitS2">
  <div class="segment a"></div>
  <div class="segment b"></div>
  <div class="segment c"></div>
  <div class="segment d"></div>
```

```
    <div class="segment e"></div>
    <div class="segment f"></div>
    <div class="segment g"></div>
  </div>
</div>
<div class="note">
  A free gift from
  <a href="https://ariksworld.com" target="_blank">AriksWorld</a>
</div>
<script>
  /*********************************************************
   * A 7-segment map for digits 0..9
   * We'll store booleans (or 1/0) for each segment: a,b,c,d,e,f,g
   * Order: [a, b, c, d, e, f, g]
   *********************************************************/
  const DIGIT_MAP = {
    '0': [1,1,1,1,1,1,0],
    '1': [0,1,1,0,0,0,0],
    '2': [1,1,0,1,1,0,1],
    '3': [1,1,1,1,0,0,1],
    '4': [0,1,1,0,0,1,1],
    '5': [1,0,1,1,0,1,1],
    '6': [1,0,1,1,1,1,1],
    '7': [1,1,1,0,0,0,0],
    '8': [1,1,1,1,1,1,1],
```

```
  '9': [1,1,1,1,0,1,1]
};
// Utility: zero-pad hours, minutes, seconds to 2 digits
function pad(num) {
  return num < 10 ? '0' + num : '' + num;
}
/**
 * Update a single digit's segments based on a number 0..9
 * digitElement: the .digit container
 * digitValue: string "0".."9"
 */
function setDigit(digitElement, digitValue) {
  const segments = digitElement.querySelectorAll('.segment');
  const mapArray = DIGIT_MAP[digitValue] || [0,0,0,0,0,0,0];
  // Each segment a..g gets turned on or off
  mapArray.forEach((onOff, idx) => {
    if (onOff) {
      segments[idx].classList.add('on');
    } else {
      segments[idx].classList.remove('on');
    }
  });
}
function updateClock() {
  const now = new Date();
```

```
    let hh = now.getHours();
    let mm = now.getMinutes();
    let ss = now.getSeconds();
    // Convert to strings with leading zeros
    hh = pad(hh); // e.g. "07"
    mm = pad(mm); // e.g. "03"
    ss = pad(ss); // e.g. "09"
    // Hours tens, hours ones
    setDigit(document.getElementById('digitH1'), hh[0]);
    setDigit(document.getElementById('digitH2'), hh[1]);
    // Minute tens, minute ones
    setDigit(document.getElementById('digitM1'), mm[0]);
    setDigit(document.getElementById('digitM2'), mm[1]);
    // Second tens, second ones
    setDigit(document.getElementById('digitS1'), ss[0]);
    setDigit(document.getElementById('digitS2'), ss[1]);
  }
  // Update every second
  setInterval(updateClock, 1000);
  // Initialize immediately
  updateClock();
 </script>
</body>
</html>
```

At AriksWorld, we believe in making technology fun, simple, and accessible**.**
This clock is just one of the many free tools we offer. Whether you're looking for unique clocks, interactive games, engaging eBooks, or creative tools, we might have something for you.

Beyond that, we are continuously developing new content, adding more free tools, eBooks, and clocks. I encourage you to visit AriksWorld.com from time to time to explore what's new. Most of our content is completely free - **no cost, no registration, no hassle!**

While all my free tools are great (in my opinion), let me recommend a few:

- Image Certificate Tool – Generate a unique certificate for any image, adding a 'paper-trail' to your copyright claims with a secure SHA-256 code.
- Fix-Color Tool – Instantly convert your image to a one fixed color. Reminds me of the blue period of Pablo Picasso.
- Leonardo Mirror - The Leonardo Mirror is a free tool that transforms your text into a beautifully mirrored image, complete with floral decoration and Leonardo da Vinci's signature.

You are free to use this clock anywhere you like. I would appreciate if you keep the **AriksWorld** link, but it's entirely up to you.

Thank you, and enjoy this clock,

Leon (Arik) Schenkler

Have questions or need help? Feel free to contact me at **breaththinkwrite@gmail.com**