



## DIY Digital Clock With 7 Segment LED Display

 by Being Engineers

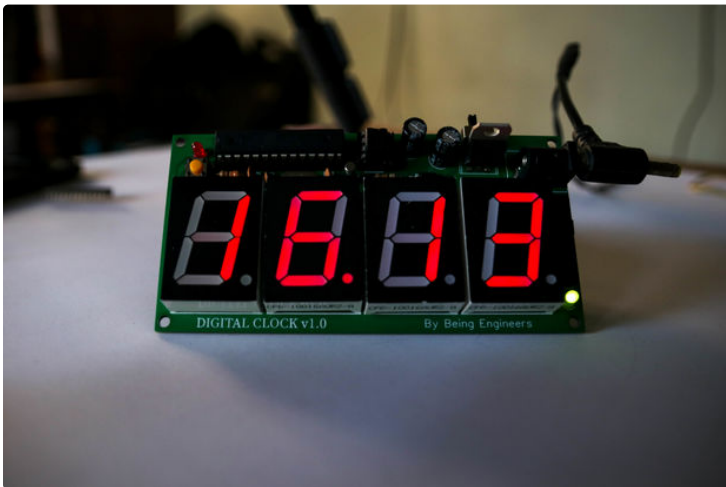
Hey guys, Welcome to Being Engineers. Hope you all are going great. This will be a short and simple project about how to make a digital clock using Arduino. Previously We have made a binary clock using charlieplexing concept. You can find the link to that tutorial in the later part of this instructables. There are countless number of ways to make a digital clock using Arduino or say any kind of microcontroller. The main visible change is the way the time is presented to the viewer. You can use a LCD screen, OLED screen or even make a completely new display technique which will be out of the box. But I want the look of a more traditional digital clock that uses 7 segment displays to portray the digit on the front panel. So I will be using 7 segment display for this project. So without further ado let's start this process.

BTW in our youtube channel we have uploaded a video on this same topic. So if you are not feeling like reading through this page, then do check out this video -

[//www.youtube.com/embed/6DLmpZpUM\\_Q](https://www.youtube.com/embed/6DLmpZpUM_Q)

DIY Binary clock using Arduino - <https://www.instructables.com/id/How-to-Make-a-Bin...>

Do follow us here in Instructables and subscribe to our Youtube channel - [www.youtube.com/c/being\\_engineers1](https://www.youtube.com/c/being_engineers1)



## Step 1: Gather the Required Components

Gather these items before starting the project -

- Atmega328p Microcontroller X 1
- DS1307 RTC IC X 1
- 1 inch Common Anode 7 Segment LED Display X 4
- 2N3904 NPN Transistor X 4
- 7805 Voltage Regulator X 1
- 16 MHz Crystal Oscillator X 1
- 32.746 KHz Crystal Oscillator X 1
- Pushbutton X 1
- Slide On/Off Switch X 1
- 3mm LED X 2
- Dc Female jack X 1
- 330E Resistor X 2
- 150E Resistor X 7
- 100K Resistor X 4
- 10K Resistor X 3
- 6.8K Resistor X 1
- 220uF Caps X 2
- 22pF Caps X 2
- 100nF Caps X2
- 1N4007 Diode X 1
- 8 pin & 28 pin IC Socket X 1 each
- 3v CR2032 battery
- 9V-12V DC 1A Power Supply

Except these you will need soldering equipment.

BOM - <https://bit.ly/2FAbbfi>

---

## Step 2: Draw the Circuit Diagram for the Clock

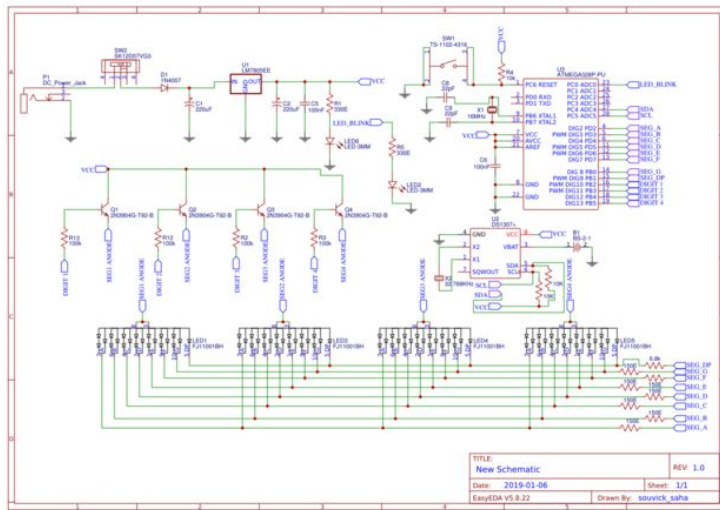
After you have gathered all the required components, it's time to draw the schematic for the project.

I use EasyEDA to draw circuit diagram for this kind of projects. It is super simple to draw circuits here with its easy to interact controls and tools. It makes life a lot easier afterwards. So this is the final circuit that I came up with. You can find the circuit diagram in the attached PDF.

I will use a DIY Arduino on board as a microcontroller. If you aren't sure about what I am talking, follow the links below.

Circuit Diagram - <https://bit.ly/2TVzSXB>

Make DIY Arduino Uno - <https://bit.ly/2BWU0RC>

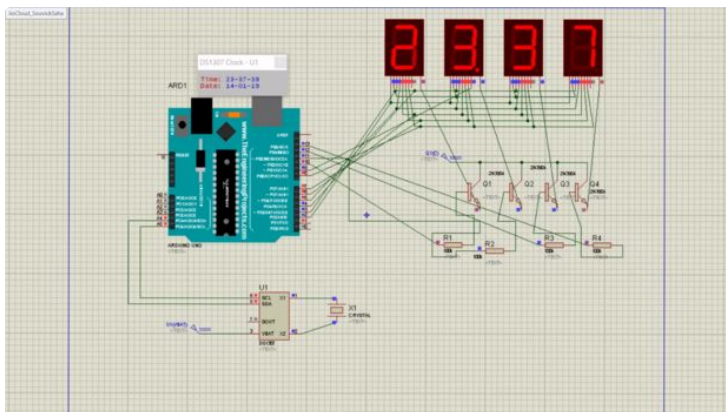


### Step 3: Simulate the Circuit

Before designing a PCB and finalizing a product it's always a good idea to simulate the circuit to check if everything works as it was intended to do or not. I will recommend you to use proteus for simulation. The available library is vast for Proteus.

In this picture you can see I have simulated the same circuit in ISIS. The result also looks good.

Proteus Design files - <https://bit.ly/2sx2URr>



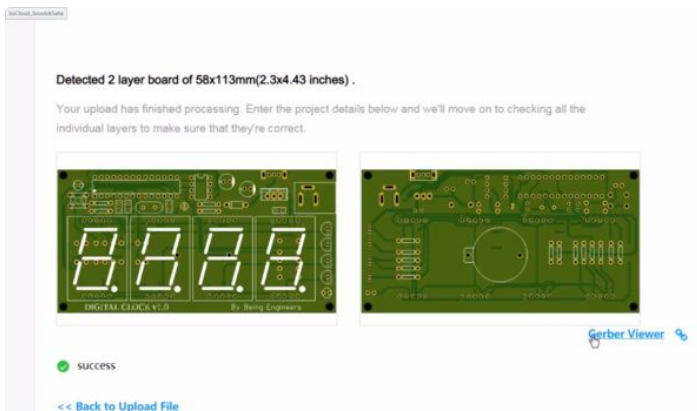
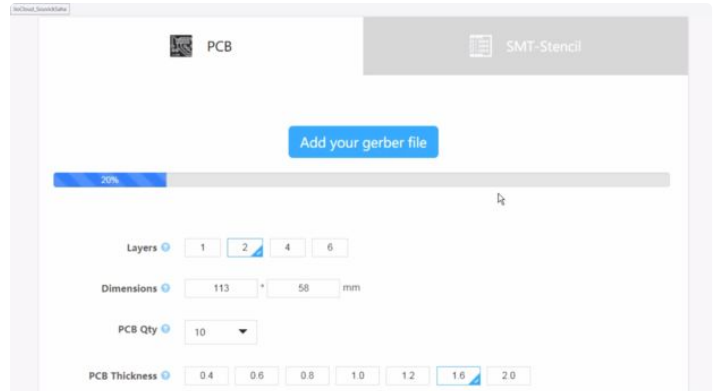
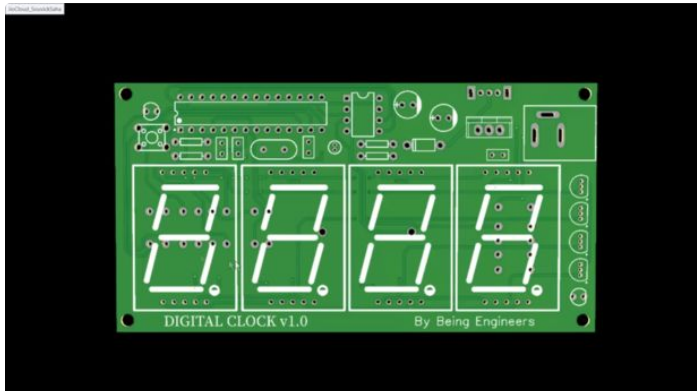
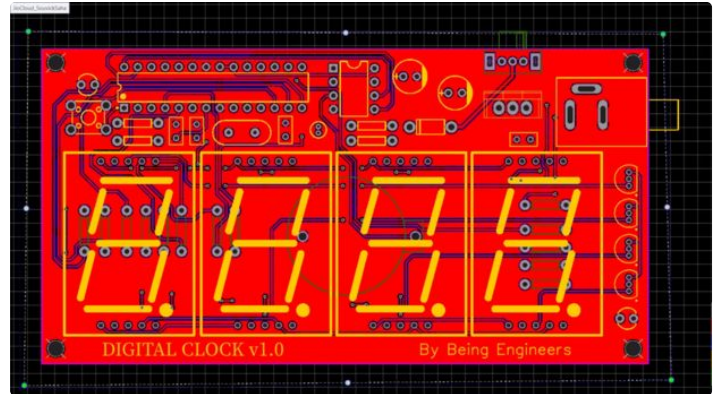
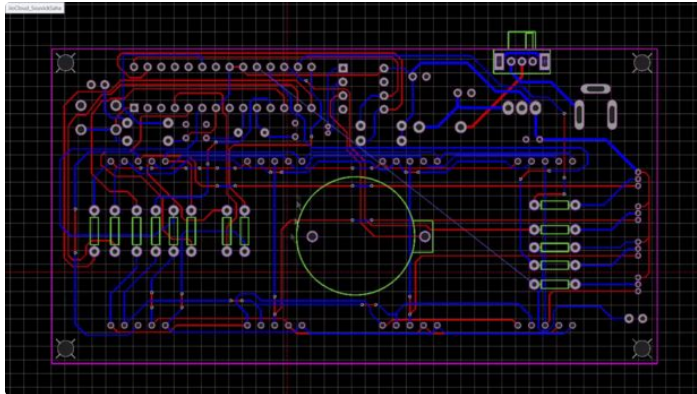
## Step 4: Design the PCB and Order

Once you are sure about the circuit implementation it's time for designing the PCB. I used [JLCPCB](#) for manufacturing the PCBs for me. It's a counterpart of EasyEDA, that helps a lot with circuit modification and updating PCB with a new components. For this step you must have some basic knowledge about designing a PCB. Even though the portal is very much user friendly for first time user, you can end up doing something wrong and that will cause your PCBs to be unusable. So be sure about the design

before generating a gerber file from the EasyEDA platform. From there you can place order directly on [JLCPCB](#), they offers a great deal for prototyping your PCBs.

PCB PDF in 1:1 Scale - <https://bit.ly/2Dd6TZS>

PCB Gerber File - <https://bit.ly/2TV3PXL>

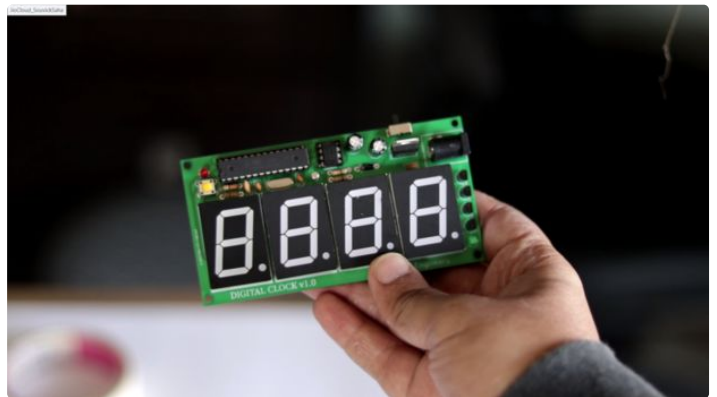
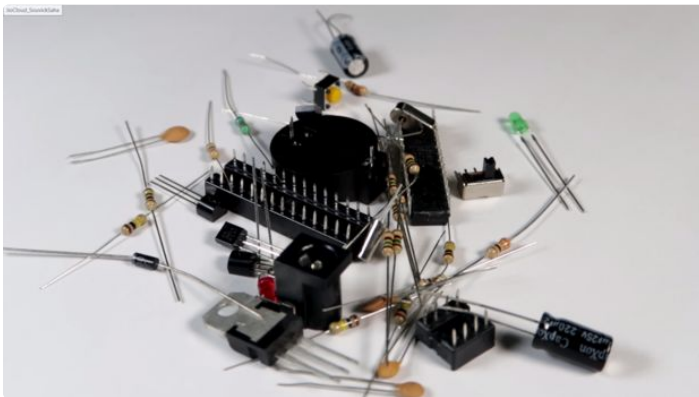


## Step 5: Solder All the Components on Board

Once you have received the PCBs, it's time to solder the components on it. This step is very straightforward. Just make a print out of the circuit diagram and BOM. Having a physical copy at the time of soldering helps a lot. Place each component on the PCB as the diagram tells. All these are through-hole components and that will result in a better as well as easier soldering experience. You don't need any soldering station for this job.

After soldering, check the continuity of each pad. Make sure that there are no solder bridges shorting two pins. Also note the orientation of the items before inserting the components in their place.

One thing you may find odd is that all the components are on the top layer of the PCB except some resistors and the 3V battery. I did this intentionally to reduce the size of this board. That's completely your choice, as long as the connections are proper everything will work. Another thing you may find odd is that there are two LEDs on board. The one at the right is the power indicator that is connected with 7805 with a current limiting resistor. The second is also a 3mm LED that is connected to the pin A0 of the microcontroller. I can use it with a blink program to check if the onboard DIY Arduino works or not. This is how I like it to be. You can remove this LED from your PCB if you want.





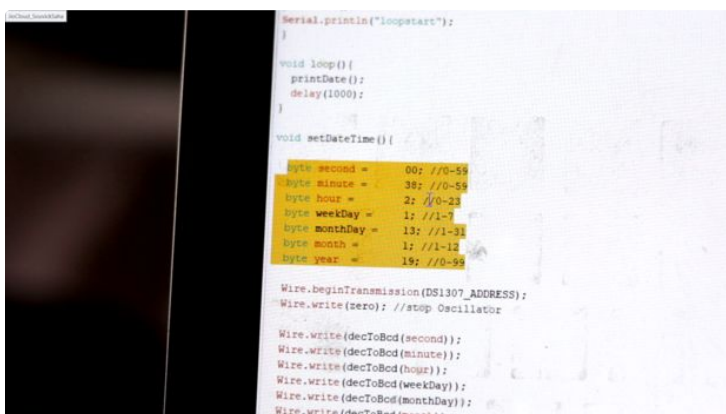
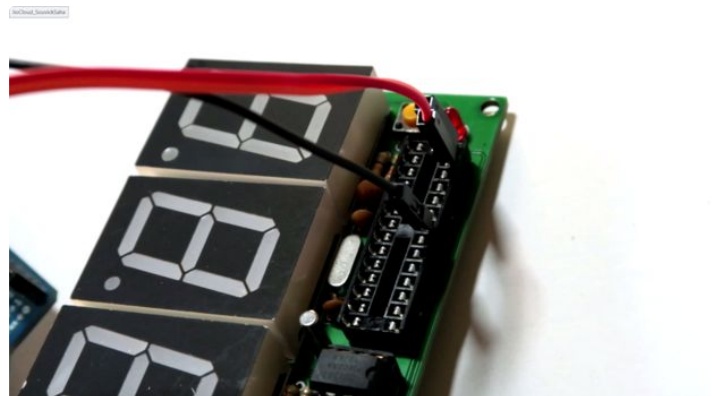
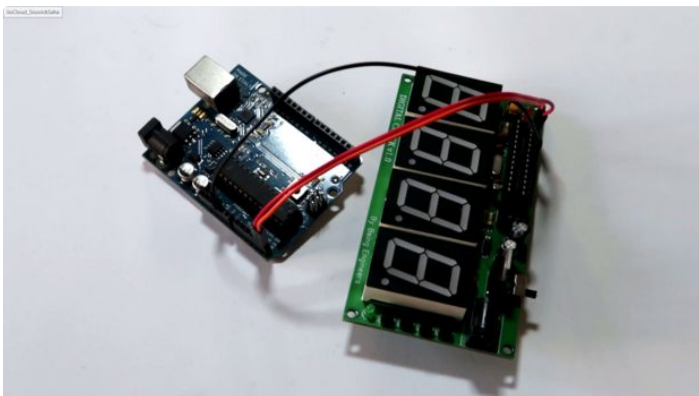
## Step 6: Set the Proper Time/Date in the RTC IC

Now you need to feed the correct time and date into the RTC module. Do the following steps -

- Insert the DS1307 IC in the 8 pin IC socket with the notch pointing outwards from the PCB.
  - Place a 3V CR2032 coin cell at the battery holder in the back.
  - Power up the PCB by a 9-12V DC supply.
  - Take a working Arduino and make the following connections with the PCB.
    - Arduino SCL(A5) to 28 pin IC Socket pin 28.
    - Arduino SDA(A4) to 28 pin IC Socket pin 27.
  - Then common the ground of both Arduino and PCB's ground.
- 
- Connect the arduino to the laptop by a proper USB cable and open Arduino IDE.
  - Open the sketch named "save\_time.ino". In the code you have to change the time and date to the correct value.
  - Upload the sketch to the Arduino Uno and that way the new date and time will be saved to the new DS1307 IC.
  - Once done remove the connections and power off the clock.

So now the clock is running in the DS1307 and the 3V battery will power up the IC all time even if the main power is off.

Arduino code to set time - <https://bit.ly/2FqVJmk>



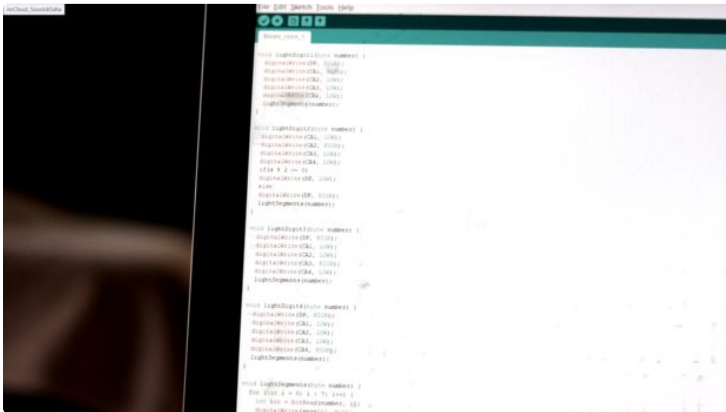
---

## Step 7: Upload the Code in the Microcontroller

In this step we have to upload the main program to the microcontroller.

- Open the sketch "Digital\_clock.ino" in Arduino IDE and upload the code to the Arduino.
- Remove the IC using a tweezer or flat head screwdriver.
- Insert the IC in the 28 pin IC socket on the PCB. The notch will be facing outwards.
- Insert the DC adapter to the Dc socket.
- Slide the switch to power up the board.

Arduino digital clock code - <https://bit.ly/2HhSi39>



---

## Step 8: It's DONE!

Now if you have everything correctly till now, the LED segments should light you with numbers and will present you the correct time. Do check if 1 minutes of this clock matches perfectly with 1 minute of a standard commercial clock. Also if you find the clock is not keeping up with the actual time, then you can change the 3V coin cell and try again.

That's it. The project is completed hereby. If you liked this project then please let us know by liking this post or commenting your inputs down below. Also follow us here in Instructables and subscribe to our youtube

channel. I will see you in my next tutorial. Till then goodbye and take care.

Instructable Profile - [https://www.instructables.com/member/Being%20Engin...](https://www.instructables.com/member/Being%20Engineers1)

Youtube channel - [www.youtube.com/c/being\\_engineers1](http://www.youtube.com/c/being_engineers1)

Thanks.

