# A COMPLETE DIGITAL CLOCK WITH IR REMOTE USING ARDUINO NANO

## Components used :

| | | |
|---|---|---|
| 1. | Arduino Nano R3/ V3 | 1 No. |
| 2. | Female edge connector for mounting Arduino ( 40 pin) | 1 No. |
| 3. | RTC module DS3231 | 1 No |
| 4. | 1 inch Common Anode 7 segment Display for Hour and Minute | 4 Nos |
| 5. | 0.5 inch Common anode 7 segment display for Seconds | 2 Nos |
| 6. | 3 mm LEDs for blinking dots display | 2 Nos. |
| 7. | Copper clad Mica sheet for making PCB 15cm X 20 CM | 1 No. |
| 8. | Infra red receiver TSOP1738 | 1 no. |
| 9. | UM 66 music generator IC | 1 no. |
| 10. | Continuous tone piezo buzzer | 1 No. |
| 11. | 2 inch small speaker 8 ohm , 0.5 w | 1 No. |
| 12. | Resistor 220 Ohm ¼ w | 1 No. |
| 13. | Resistor 150 Ohm ¼ w | 8 Nos. |
| 14. | Resistor 100 Ohm ¼ w | 1 No. |
| 15. | IR Remote control unit | 1 No. |

## FEATURES:

1. **Very Compact in size with all functions and all functions can be accessed through IR remote**
2. **Hour/ Minute /Second and Room Temperature Display**
3. **Real time and Alarm time can be set using IR remote very easily.**
4. **Alarm time once set is stored in the flash memory of Arduino, so that , the Alarm time set will not lost on unplugging the clock from Power or on power failure.**
5. **Alarm when blown can be switched off using remote control. Also, if snooze button of remote is pressed the alarm will blow again after the set snooze time.**
6. **The clock will display room temperature when the button for the same on remote is pressed.**

7. **Musical Hour chime on every hour, duration of which can be set.**
8. **Clock display can be switched off and on with single pressing of a button on remote.**
9. **A return to normal clock mode button is provided in the remote to return from Alarm set/ time set with out making changes or to toggle between temperature / time display modes.**
10. **The last 2 digits (second displays) will light as "AL" when in Alarm setting mode and "st" while in time set mode and"0C" while displaying temperature.**

## Construction:

Please refer the circuit diagram attached. For making the project compact and usable for normal use, I have made the project using a PCB made at home itself. I have drafted PCB layout of the clock with a free PCB routing software DipTrace which is very convenient to use. Since it is difficult to make multi layered PCBs at home , the PCB layout was made as single sided PCB. Layout of PCB drawn with Dip Trace is also attached. For making PCB, I have printed the Layout using a normal inkjet printer. After that pasted a masking tape over the entire copper side of the mica sheet. Over the pasted masking tape pasted the printed PCB layout using Fevicol. The White area on the layout was carefully cut and peeled off with a sharp cutting Knife /razor blade. Then immersed the mica sheet in Ferric chloride Solution to remove the exposed copper parts and complete the PCB. The result was very good , and cheap too. Other methods of PCB making like using photo resist etc a which are suitable for printing of larger no of PCBs. Please see the PCB made with the above method.

All the components were soldered directly in to the PCB, but can also be made in general purpose PCB if desired. The Arduino nano was put the board using 2 pieces of 15 pin female edge connectors. A normal 5V , 2A mobile phone adaptor is using to power the clock

## Working:

The clock uses 6 Nos of common anode Seven segment displays , four of which are 1 inch in size used for the display of hour and minutes. The last two 0.5 inch size common anode displays are used to display the seconds in normal mode and mode status in other modes like alarm set/ time set / temperature etc.

All the corresponding segments of all the six displays are connected together and connect to seven pins of Arduino . Also Common anodes of each of the six displays are connected to another six pins of Arduino. For display of the blinking dots , two 3 mm red LEDs are connected in parallel and connected

to a pin of Arduino . Thus 14 pins of Arduino are connected to the display. Current limiting resistors are used in series with all common anode pins of the all the displays. Time multiplexing is method is used to display the time , in which, Arduino is programmed to light each digit one by one for a small time, at a very faster rate so that all the displays appear as lighted at the same time. An Infra Red sensor is connected to one of the pins of Arduino to receive the signals from the remote and do the corresponding functions assigned in the sketch.

I have used a **3231RTC Module** For the keeping real time in the project. I have opted this module though a bit costly over other RTCs , since this is having very high precision and other features like temperature measuring , Flash memory etc. Any Discarded remote of TV/Video/Audio etc can be used as remote control , but I used a cheap small remote purchased from a component shop which cost about 35 Indian Rupees(less than half a Dollar).

I have earlier published a Clock project, and that was my first project on Arduino. That project has only limited in features( even with out a facility for time setting) due to constraints in getting sufficient pins on Arduino nano to write the program. But by adding an InfraRed remote, it give me flexibility of incorporating all the features that requires ( may be more if required), with out the need of extra pins and also avoiding slide and push switches. Also IR remote make the clock very compact and convenient to set and use. I am publishing this project as a new one , since the previous project's sketch was extensively changed to incorporate IR remote and adding additional features, and also it is a finished project.

The working principle of the clock is that ,the program read the time from the RTC and displays it on the seven segment multiplexed displays. Also the temperature stored within the DS3231 RTC module is read and displayed while in temperature display mode.

Any Remote control can be linked with the clock , for which you need to set code send by the remote on pressing each key in to the sketch. For finding the code you may run the sketch on a computer with Serial Monitor on and removing // on the lines 187 and 188 of the sketch. On pressing each key on the remote the corresponding code will be shown on the serial monitor. Write down the code and use it to edit the appropriate code in the sketch. I will append the IR code used in my sketch for each key of the remote I used for a guidance.

The Hour chime is made with a low cost 3 pin UM66 melody generator IC and with a single BC147 transistor amplifier , and a small 2 inch sized speaker all readily available in the market. The Hour chime will shut of after a time which can be set in the program . with a few additional lines in the stretch the chime can be switched off at night (between a time interval) if required.

A continuous tone active piezo buzzer is used to sound alarm , and the same is converted as beeper ( beep- beep like tone ) with appropriate lines in the sketch .The alarm duration can be set within the sketch. The snooze time can also be set within the sketch.

The Alarm time, once set will be written in to the flash memory of the Arudino and it will remain even after a power failure or switch off/on .

Due to shortage of sufficient no of pins, the blinking dots are connected to D1 pin, which is being used by the Arudino for serial communication thorough USB or other. So to work the blinking dots , disable all serial communication lines in the program once finished.

The Clock is programmed as 12 Hour clock, but can be modified as 24 hour if required with minor modification. The Alarm setting time is in kept in 24 hour mode to know whether the alarm is set to AM or PM.

The casing of the Clock was made from an old discarded power extension chords enclosure which was modified and painted with spray paint.. Actually the PCB size was selected to suit the case perfectly.

## Functions on Remote:

1. **DISPLAY ON/OFF** : pressing the button switch off and switch on the seven segment displays , useful at nights
2. **TEMP** :Pressing of this key will display the room temperature in degree centigrade. In this mode the last 2 segments shows as"0C" . Pressing NOR MOD key on remote will bring back the display to show real time
3. **ALM SET** : by pressing this button the required alarm time can be entered using the numeric keys on remote. .After entering the Alarm time , press the enter key to register /write the Alarm set time to the flash memory of Arduino , and return to normal time mode. Pressing NOR MOD will just return to normal clock mode with out doing any thing. While in Alarm set mode the last 2 digits of the displays shows as "AL" . Alarm time has to be set in 24 hour mode only.
4. **ALM OFF** When the Alarm beeps at the set time, by pressing this key, the alarm can be switched off. If Alarm off button is not pressed , the Alarm will continue up to some time set in the sketch (which can be changed) and after that switch off automatically
5. **SNZ ON:** While alarm beeps, if this button is pressed, the alarm beeps again after a snooze interval (which can be set in the sketch). Snooze button can be pressed again and the alarm can be switched off by pressing ALM OFF button.

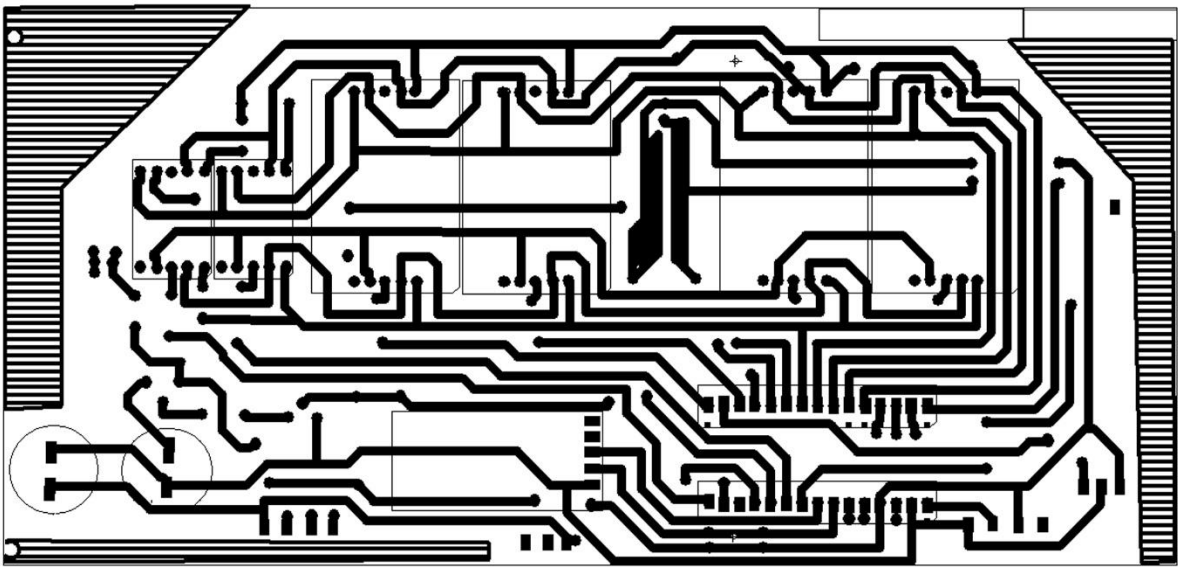
6. **TIME SET** : Pressing this button will bring the display to time set mode. In this set mode the last 2 digits of the displays shows as “st”. Time to be set can be entered using the numeric keys on remote followed by pressing of **ENTER** key. Also can be returned to Normal clock mode by pressing NORM MOD button with out doing any thing.

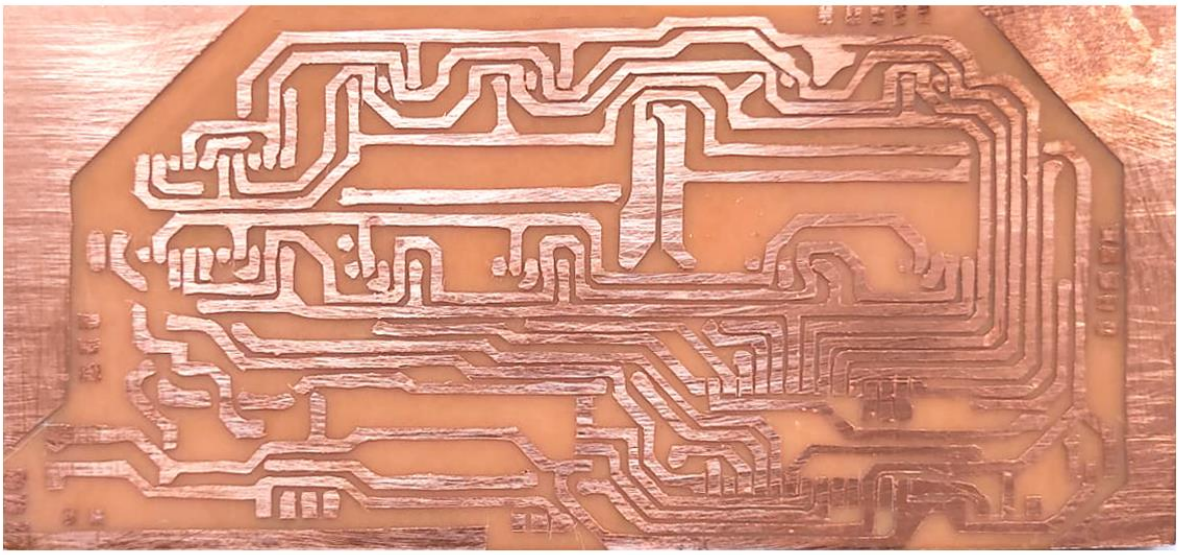
## IR codes used in the sketch corresponding to the IR remote I have used:

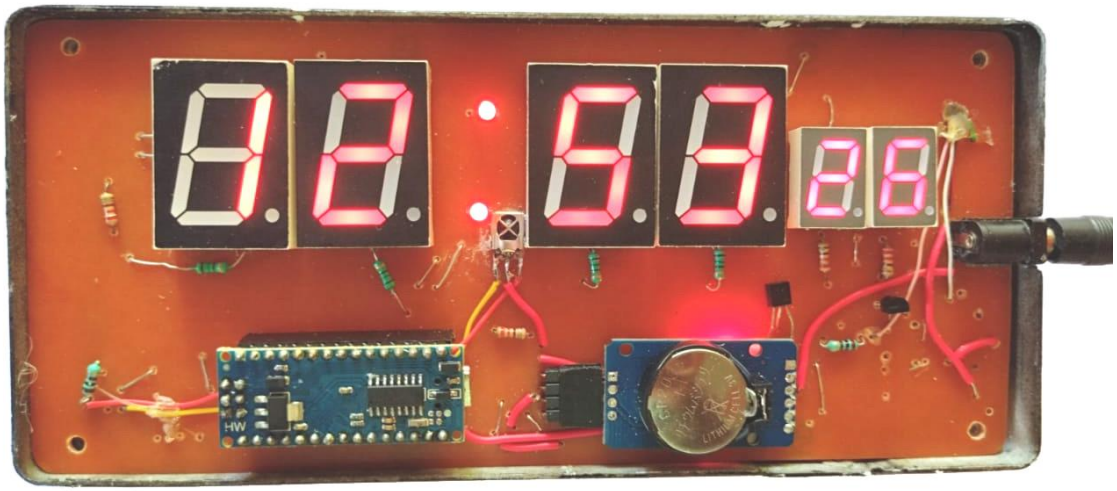
| KEY | | IR CODE |
|---|---|---|
| DISPLAY ON/OFF | : | 0x1FE48B7 |
| TEMP | : | 0x1FE7887 |
| ALM SET | : | 0x1FE40BF |
| TIME SET | : | 0x1FEC03F |
| ENTER | : | 0x1FE20DF |
| ALM OFF | : | 0x1FEA05F |
| SNZ ON | : | 0x1FE609F |
| NOR MOD | : | 0x1FE58A7 |
| 0 | : | 0x1FEE01F |
| 1 | : | 0x1FE50AF |
| 2 | : | 0x1FED827 |
| 3 | : | 0x1FEF807 |
| 4 | : | 0x1FE30CF |
| 5 | : | 0x1FEB04F |
| 6 | : | 0x1FE708F |
| 7 | : | 0x1FE00FF |
| 8 | : | 0x1FE9867 |
| 9 | : | 0x1FE9867 |


Modify the sketch with appropriate IR code of the remote using

## PCB LAYOUT DRAWN WITH DIPTRACE AND ACTUAL PCB ETCHED







**ASSEMBLED PCB**


**FINISHED CLOCK**


**TIME SETTING MODE**


**ALARM SETTING MODE**


**TEMERATURE DISPLAY**

**AJITH KUMAR. B**