Digital Clock with Alarm system.

Team Members: Sayali Bhosekar (09026003) , PrithikaVageeswaran (09D26010)

Components: Two 8*8 LED Matrix display (to display the digital clock), DS 1307 to compute time, date and year, piezo buzzor, aurdino board, switches

Working Principle: To provide various options in the digital clock, following switches are used:

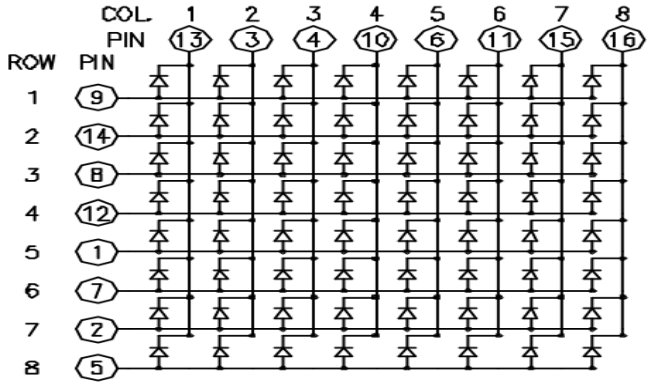
Switch 1 (S_1)	:	alarm on	(to initiate the alarm set option)
Switch 2 (S_2)	:	alarm off	(to put alarm off)
Switch 3 (S_3)	:	Increment	(to increase the value at selected segment)
Switch 4 (S_4)	:	Next	(to set the next digit for alarm time)
Switch 5 (S_5)	:	Snooze	(to snooze alarm)

We use a driver IC Max7219 to operate the 8*8 LED matrices. The advantage of this is that Max7219 uses only 3 pins of microcontroller and thus saves the number of pins required.

Another feature of the clock is the snooze function. Everytime the snooze function is pressed on a motor rotates. The snooze switch is connected to the motor and the switch moves away from the user on rotating the motor.

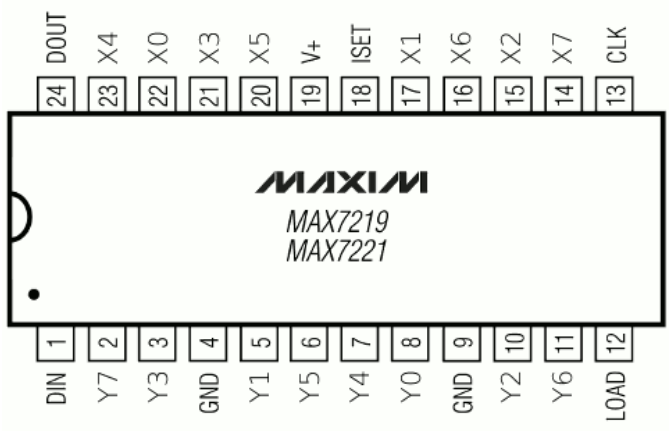
Pin Diagrams of the components used:

a) Pin diagram of 8 cross 8 LED matric



For one colour LED matrix there is no distinction between common cathode and common anode LED matrices. To make a particular LED glow make the row number pin High and the column pin Low. We can control all the LEDs in this way.

b) Max7219 IC:

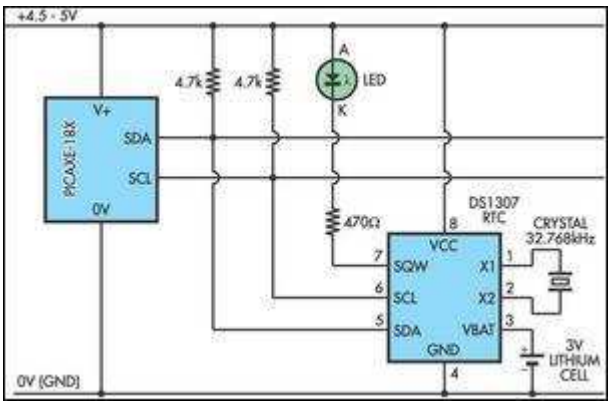


Data in pin goes to pin number 12 of microcontroller

CLK pin 13 is connected to the pin 11 of microcontroller

LOAD pin 12 is connected to pin 10

c) DS 1307 circuit.



A real time clock is basically just like a watch - it runs on a battery and keeps time when there is a power outage. Using an RTC, you can keep track of long timelines, even if you reprogram your microcontroller or disconnect it from USB or a power plug.

Arduino has a built-in timekeeper **millis()** and there are also timers built into the chip that can keep track of longer time periods like minutes or days. However , **millis()** only keeps track of time since the Arduino was last powered - . That means that when the power is turned on, the millisecond timer is set back to 0. The Arduino doesn't know that it's 'Tuesday' or 'March 8th', all it can tell is 'It's been 14,000 milliseconds since I was last turned on'. We

will need to have consistent timekeeping that doesn't reset when the Arduino battery dies or is reprogrammed**.** Thus, we include a seperate RTC.The RTC chip is a specialized chip that just keeps track of time.

Work Division:

Sayali Bhosekar:- Digital Clock, snooze alarm working

PrithikaVageeswaran: Alarm clock functions.