MAKERSPACE

DIGITAL CLOCK

Members:

- Vivek Adajania
- Sachin Sharma
- Jatin
- Anurag

Mentors:

- Dhruv Patel
- Millie Bhanani
- Bhumil Depani

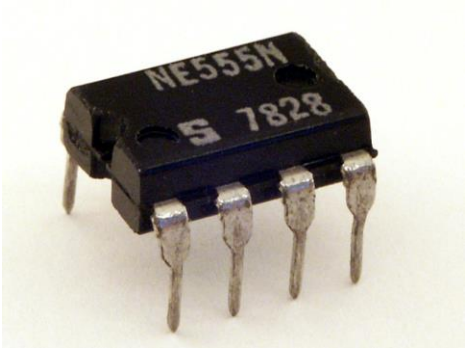
Index

Problem Statement:

To design a clock using 7 segment displays showing hours, minutes and seconds.

Solutions:

1.Atmega’s Internal Clock
2.Real Time clock-DS1307 module
3.555 timer(NE555N)



Solution 1(Atmega's internal clock)

Working: Atmega128 8-bit timer and interrupts.

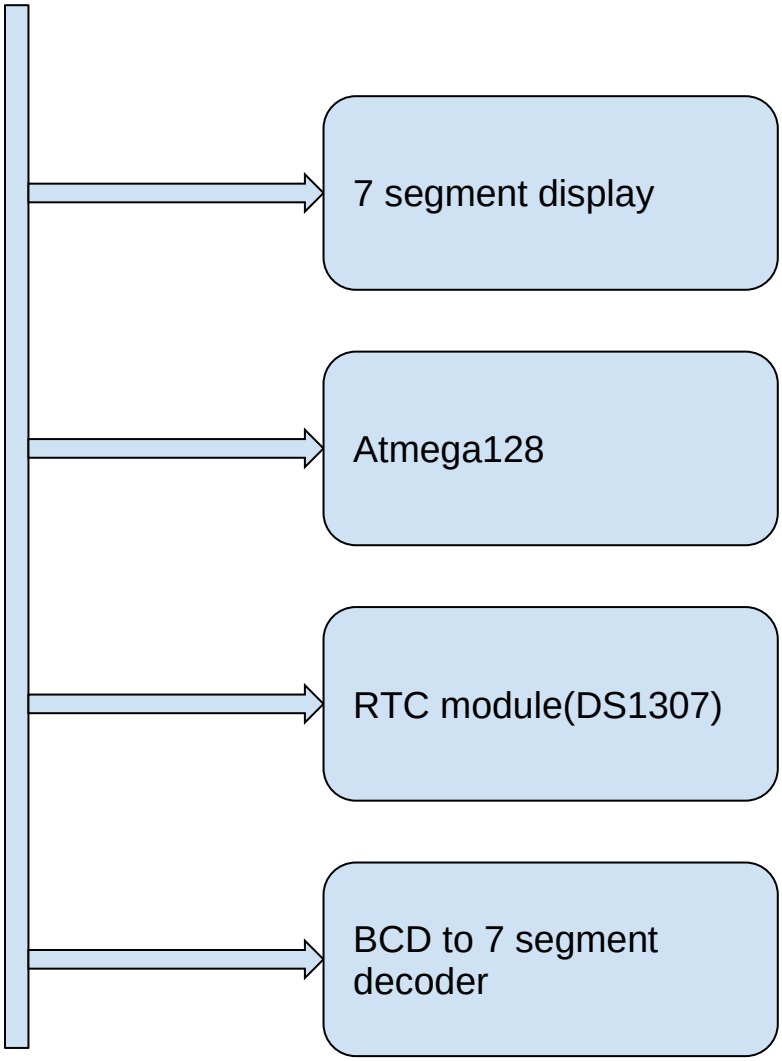
Problem Faced:

A timer count of 1.2s was obtained using 8-bit timer.
An error of 0.2s and accumulation of it by time thus wrong time will be shown.

Refer page number 92 in datasheet.

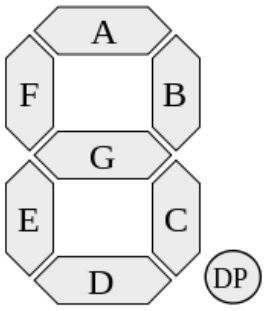
Solution 2(RTC)

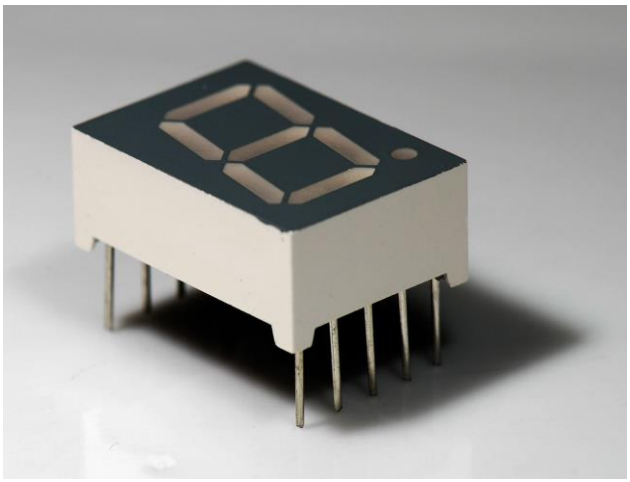
Hardware overview



7 segment display

A **seven-segment display** (**SSD**) is a form of electronic display device for displaying decimal numerals.





Truth Table:

Segments Inputs							7 Segment Display Output
a	b	c	d	e	f	g	
0	0	0	0	0	0	1	0
1	0	0	1	1	1	1	1
0	0	1	0	0	1	0	2
0	0	0	0	1	1	0	3
1	0	0	1	1	0	0	4
0	1	0	0	1	0	0	5
0	1	0	0	0	0	0	6
0	0	0	1	1	1	1	7
0	0	0	0	0	0	0	8
0	0	0	0	1	1	0	9

Atmega128

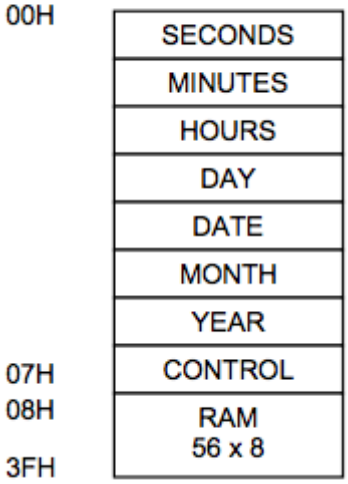
The high-performance, low-power Microchip 8-bit AVR RISC-based microcontroller combines 128KB of programmable flash memory, 4KB SRAM, a 4KB EEPROM, an 8-channel 10-bit A/D converter, and a JTAG interface for on-chip debugging. The device supports throughput of 16 MIPS at 16 MHz and operates between 4.5-5.5 volts.
By executing instructions in a single clock cycle, the device achieves throughputs approaching 1 MIPS per MHz, balancing power consumption and processing speed.

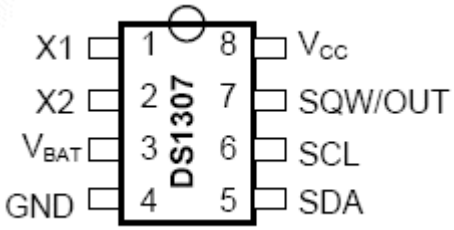
Real Time Clock Module(DS1307)

A real-time clock (RTC) is an integrated circuit that keeps track of the current time.

We have to just initialize the RTC module, it will run on it's own then we have to just read data from it.
For communication between microcontroller and RTC, we have used I2C communication.
The memory address of RTC is given below:

Figure 2

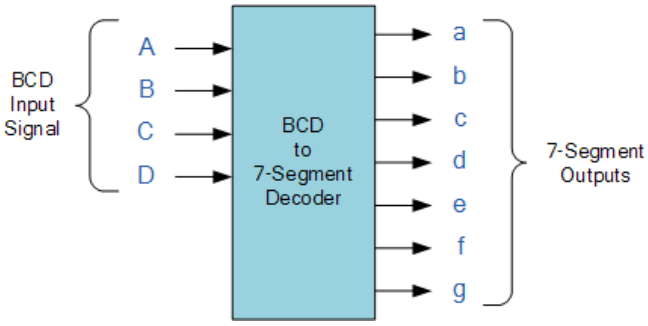




1. **Vbat-Backup battery**
2. **Vcc-5v**
3. **GND-0v**
4. **SDA & SCL-Interfacing pins**
5. **X1-x2 crystal oscillator**
6. **SQWout-to obtain waveform**

BCD to 7 segment decoder

- Every 7-segment needs 7 inputs in order to display something.
- We have 6 7-segments(hh:mm:ss) so we need 42 inputs.
- One can't use 42 input pins from microcontroller as it will be hectic in soldering and some microcontrollers won't even have 42 pins.



Binary Inputs				Decoder Outputs							7-Segment Display Outputs
D	C	B	A	a	b	c	d	e	f	g	
0	0	0	**0**	1	1	1	1	1	1	0	0
0	0	0	**1**	0	1	1	0	0	0	0	**1**
0	0	**1**	0	1	1	0	1	1	0	1	**2**
0	0	1	1	1	1	1	1	0	0	1	3
0	**1**	0	0	0	1	1	0	0	1	1	**4**
0	1	0	1	1	0	1	1	0	1	1	5
0	1	1	0	1	0	1	1	1	1	1	6
0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	0	1	1	1	1	1	1	1	**8**
1	0	0	1	1	1	1	1	0	1	1	9

Software overview

- UART(Universal Asynchronous Receiver-Transmitter)
- I2C(Two wire interface)

UART-Universal Asynchronous Receiver-Transmitter

- UART is serial communication to the transfer data at a particular bitrate/baudrate.
- We used UART to receive the RTC's data(time) in XCTU.

I2C- Two wire interface

- Serial communication protocol.
- Everything works on just two lines(Clock line and Data line).
- It is synchronous.

Reference: here