T Clock 11

Analog electronics clock

Dr. Tõnis

Introduction

As it seems, I like to make different clocks. I have built and designed numerous electronic and mechanical clocks and this one is another one. My first electronics clock required several iterations and I learned a lot.

The presented design is improvement of the older design – it is smaller and cheaper to build than the previous versions. Additionally, I have documented the process much better this time.

The clock is electronic, without any microcontroller. The time is generated from a 32.768 kHz crystal and by counting the crystal oscillations the time could be shown. The numbers are constructed with LEDs in a seven-segment display formation.

In the following the BOM is introduced, thereafter the design is introduced and eventually the assembly process is shown.

Have fun!

BOM

- Everything through hole components (I got everything from Aliexpress)
  - 74HC393N - 8 pcs
  - 74HC32N - 3 pcs
  - 74HC08N - 3 pcs
  - 74LS47N - 6 pcs
  - NE555N - 1 pcs
  - 8-bit switch - 3 pcs
  - 6mm button - 2 pcs
  - Resistor 10k - 9 pcs
  - Resistor 1M - 5 pcs
  - Resistor 1k - 1 pcs
  - Resistor 560Ω - 52 pcs (follow the comments in the end, I used 560Ω)
  - Capacitor 100n - 15 pcs
  - Capacitor 16p - 1 pcs
  - Capacitor 8p - 1 pcs
  - 32.768kHz crystal - 1 pcs
  - Led - 128 pcs (you can use any color you like, 3 or 5mm LEDs, I used 5mm)
  - M3 screws (>5mm) and nuts - 4 pcs
  - I would strongly recommend using IC sockets instead of directly soldering the components

Design explanation

- The first thing what we need is some kind of a clock signal.
- The ideal would be to have a 1 second reference signal that we could use for the second indication.
- Obviously the accuracy is very important for us. Thus, the clock signal should be stable and relatively accurate.
- In the electronic clocks usually 32768 Hz crystal is used as a reference and we could use the same one.
- The 32768 Hz signal is very useful as it is 2^{15} (2^{15} = 32768), meaning we could divided it easily and in the end we have a 1Hz signal.
- First, how to use the crystal to generate the 32768 Hz signal?

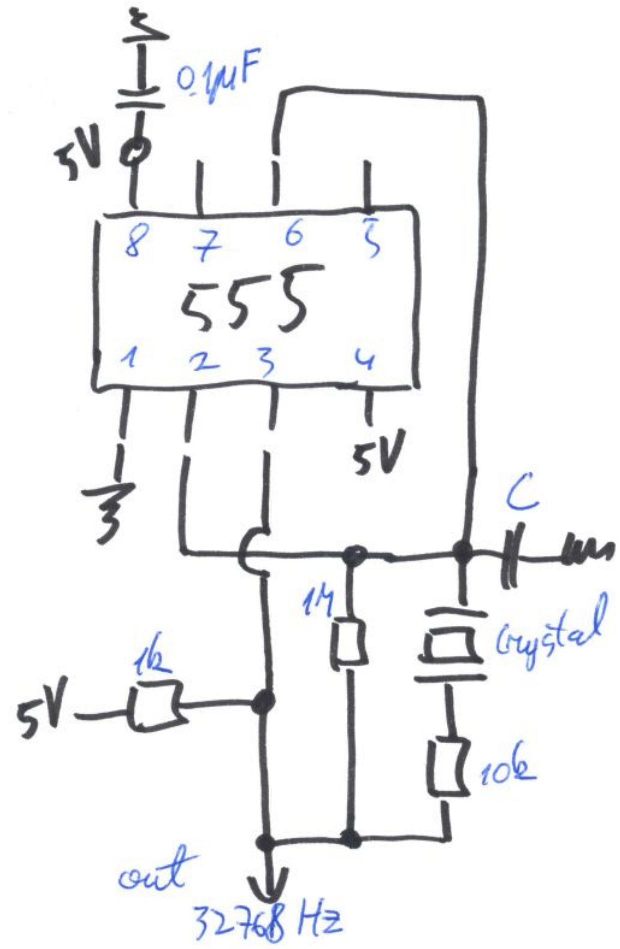
Design explanation - generating 32768 Hz signal

- After some online research several different methods could be found:
- Using CD4060 chip – Link1, Link2
- Using 555 timer – Link3
- Using 74LS04/74HC04 (inverter) chip – Link4, Link5, Link6

- I didn't have the CD4060 chip, thus couldn't test the shown circuits
- I had 74HC04 laying around and tested the circuits shown under Link4, Link5 and Link6. They all require an Op-Amp for a rectangular pulse generation and I was not very impressed with the circuit. I did not want to use an Op-Amp.
- The 555 timer was producing good results. I tested NE555 and TLC555 chip and both give the same result, despite often the CMOS (TLC) version is recommended.

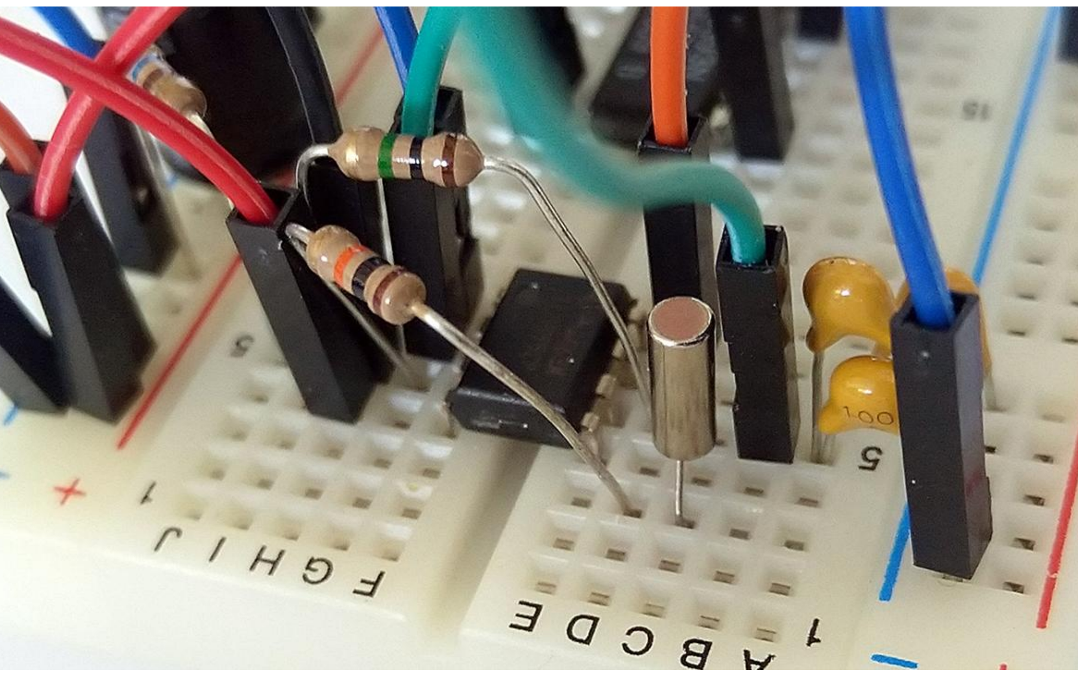
Design explanation - generating 32768 Hz signal

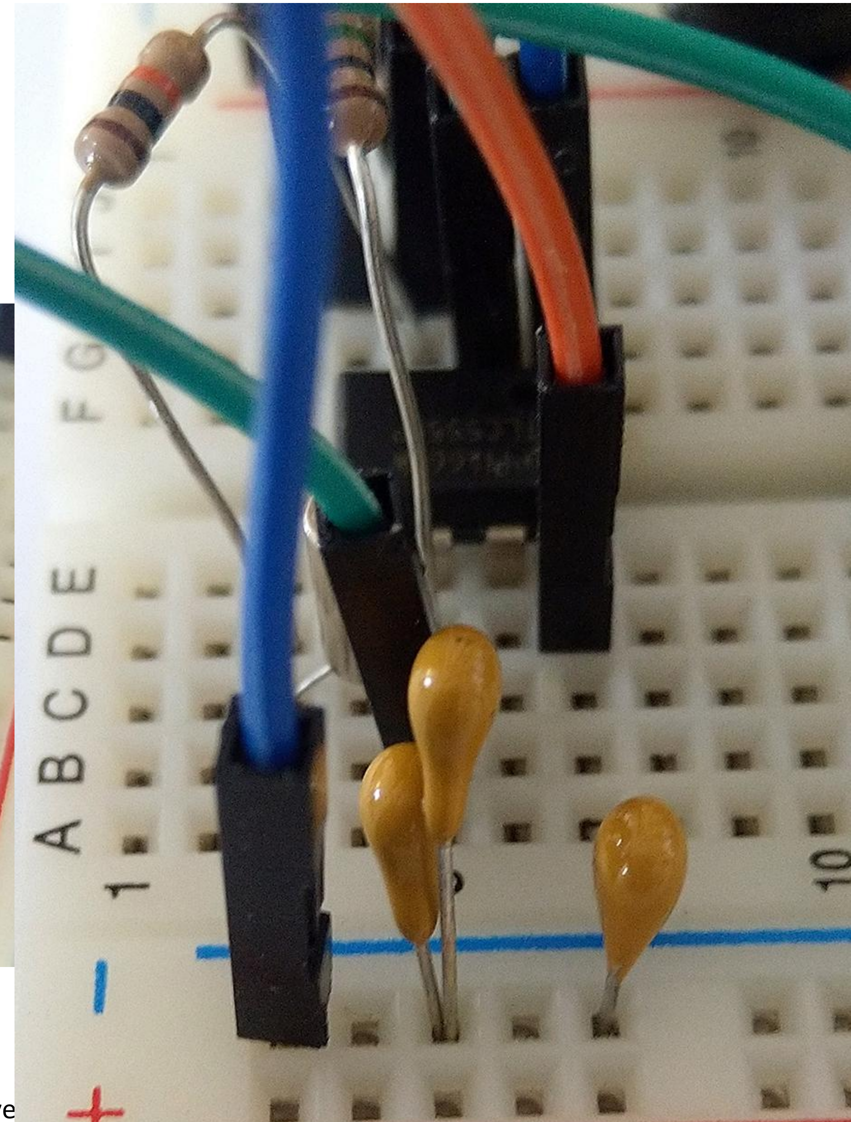
- The layout schema can be seen in the right
- The output frequency was measured with an oscilloscope
- Keep in mind, the crystal is very sensitive device and one should follow the datasheet very carefully. Thus, the output frequency was very strongly depending on the capacitance used with the crystal.
    - No C (C = 0) -> f = 69.5 kHz
    - C = 16pF -> f = 33.229 kHz
    - C = 10pF -> f = 33.545 kHz
    - C = 8pF -> f = 32.771 kHz
    - C = 5pF -> f = 32.771 kHz
- I had 16 pF and 10 pF capacitors, thus I connected them in series or parallel to get the wished capacitance values.
- 3 Hz error means one second error after 32768 / 3 = 10922.7 sec = 3.03 h → after every 3h the clock will be 1 second off.
- Sounds good enough



Design explanation - generating 32768 Hz signal

Breadboard test circuit

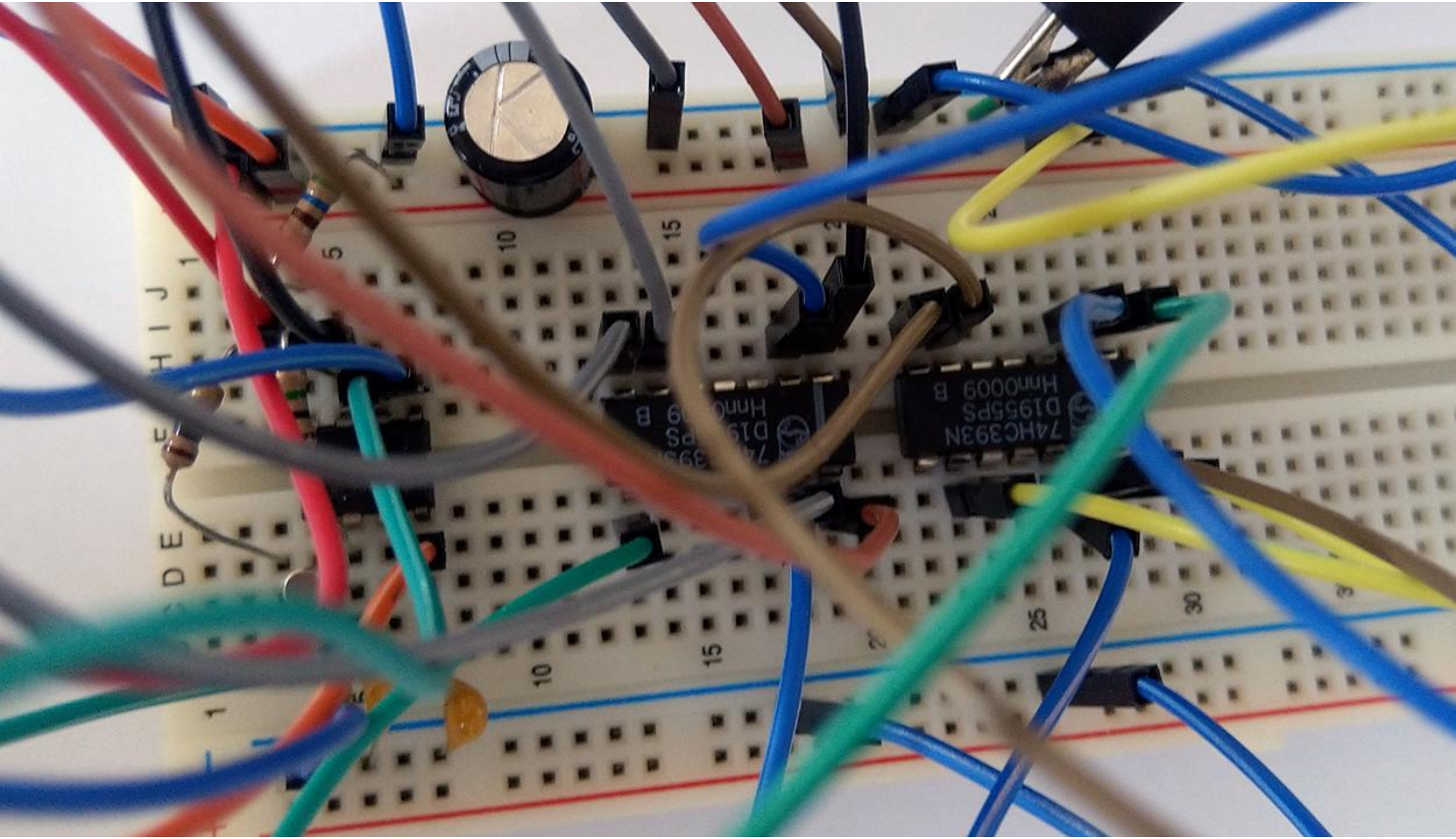




Design explanation - generating 1 Hz signal

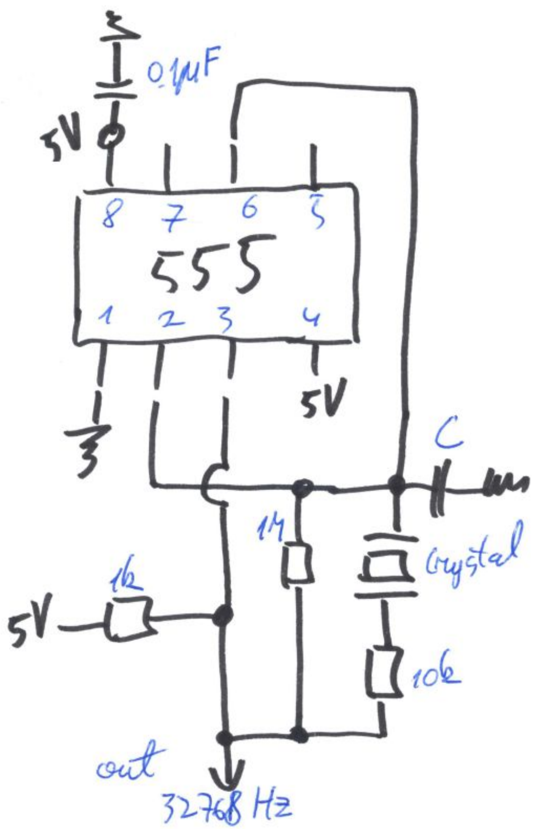
- Now having the 32768 Hz signal we could consider different methods how to slow it down.
- The simplest is to use a counter.
- We use a 4 bit counter as the same counter will be needed later for counting the seconds, minutes and hours.
- More about digital counters can be found under these links: Link1, Link2, Link3
- A 4 bit counter divides our input frequency 2^4 = 16 times.
- In order to achieve 2^{15} = 32768 division we need 4 counters.
- There are many counters in the market and I have chosen 74HC393 in this circuit.
- The 74HC393 has two parallel circuits, thus we only need two chips.
- In the following figure a breadboard test circuits can seen (together with the 555 chip and 32768 Hz generator).

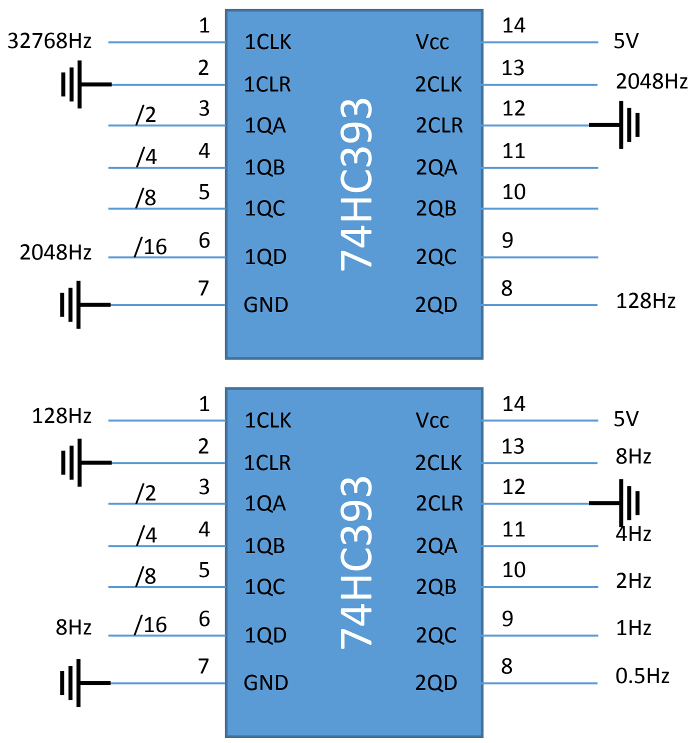
Design explanation - generating 1 Hz signal



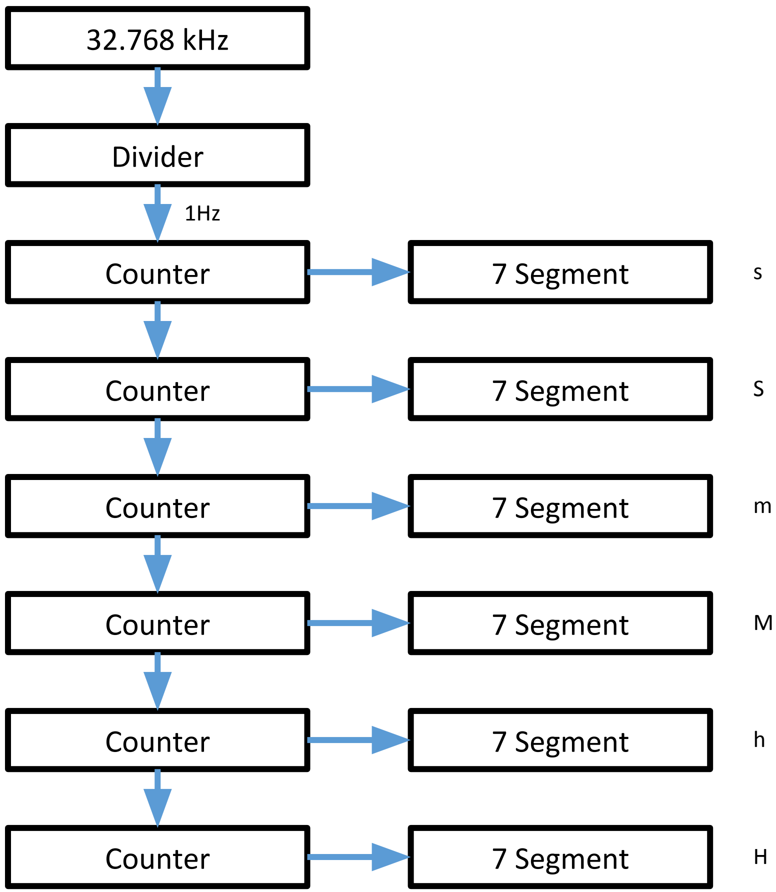
Design explanation - generating 1 Hz signal

- The layout schema can be seen in the right
- With only two 74HC393 chips a 1 Hz signal can be generated.
- Now with the 1 Hz signal one could plan the whole clock logic structure.





Design explanation - logic flow structure



- On the left one could see the logic flow structure
- Without much discussion the used components are
  - Divider: 74HC393
  - Counter: 74HC393
  - 7 segment: SN74LS47
- As can be seen, all of the digits requires the same components and are followed by the same logic.
- We could divide the time as following: HH:MM:SS ⟶ h H : m M : s S
- In the next stage we look how we could build a second panel:
  - Seconds indicator
  - Tens of the seconds indicator
  - Resetting the counters at the correct point

Design explanation - second indication

- There are 60 seconds in one minute, thus
  - The second counter should count from 0 to 9 and then reset.
  - Tens of the second should count from 0 to 5 and then reset.
- The counter 74HC393 datasheet: [Link]
- The counter outputs can be seen on the right
- If we want to count up to 9 then when the counter reaches to 10 we need to make the reset.
- If the second counter makes reset we need to send the counting pulse to the tens counter.
- To reset the counter 74HC393 we need a HIGH signal on the CLR pin.

COUNT	OUTPUTS			
	Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H
10	H	L	H	L
11	H	L	H	H
12	H	H	L	L
13	H	H	L	H
14	H	H	H	L
15	H	H	H	H

Design explanation - second indication

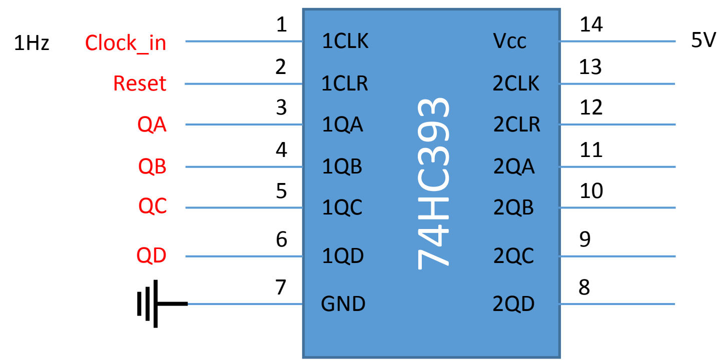
- If the counter reads 10:
  - Qd is HIGH
  - Qc is LOW
  - Qb is HIGH
  - Qa is LOW
- When we observe the table we can notice that if Qd is HIGH the count is at least 8 → important to determine the count 10 (we don't care about the count as long Qd is LOW).
- Between 8 and 11 Qc is LOW → not interesting to determine the count 10 (we should never reach the point when Qc is HIGH).
- Between 10 and 11 is Qb HIGH → important to determine the count 10.
- Qa state not interesting to determine the count 10.
- Consequently, if Qd is HIGH and Qb is HIGH we have count 10 (we will make the reset before we reach count 11).

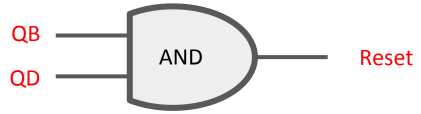
COUNT	OUTPUTS			
	Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H
10	H	L	H	L
11	H	L	H	H
12	H	H	L	L
13	H	H	L	H
14	H	H	H	L
15	H	H	H	H

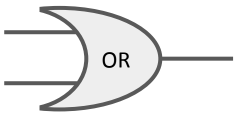
Design explanation - second indication

- Clock_in for the seconds is the 1Hz signal
- If Qb and Qd are both HIGH (number 10) we reset the counter.

- OR element is for the later applications.







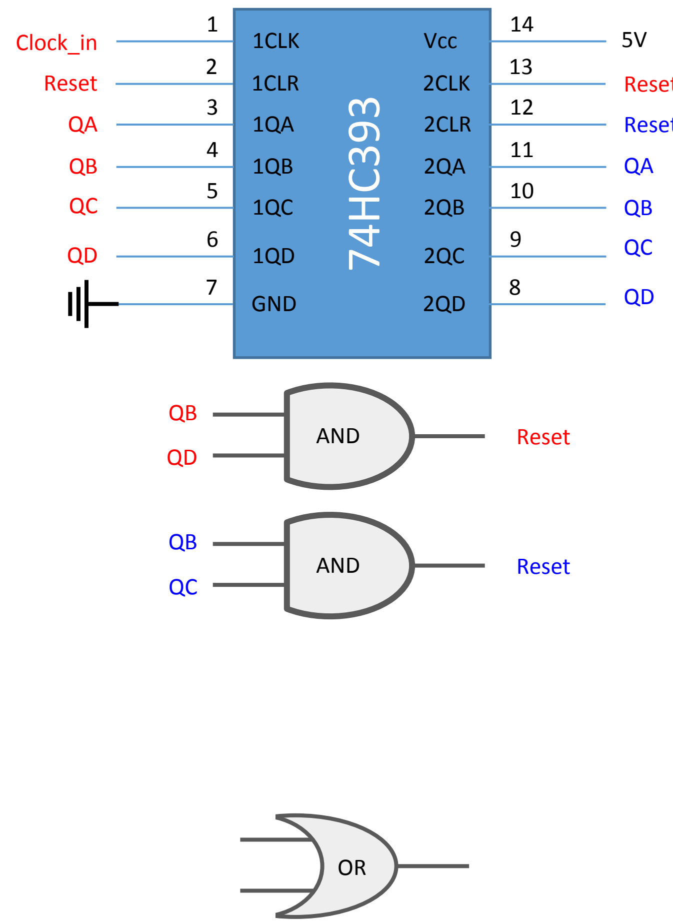
Design explanation - second indication

- If the counter reads 6:
    - Qd is LOW
    - Qc is HIGH
    - Qb is HIGH
    - Qa is LOW
- When we observe the table we can notice that if Qd is HIGH the count is at least 8 → not interesting to determine the count 6 (Qd is never HIGH).
- Between 4 and 7 Qc is HIGH → important to determine the count 6.
- Between 6 and 7 is Qb HIGH → important to determine the count 6.
- Qa state not interesting to determine the count 6.
- Consequently, if Qc is HIGH and Qb is HIGH we have count 6 (we reset the counter before we reach higher counts).

COUNT	OUTPUTS			
	Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H
10	H	L	H	L
11	H	L	H	H
12	H	H	L	L
13	H	H	L	H
14	H	H	H	L
15	H	H	H	H

Design explanation - second indication

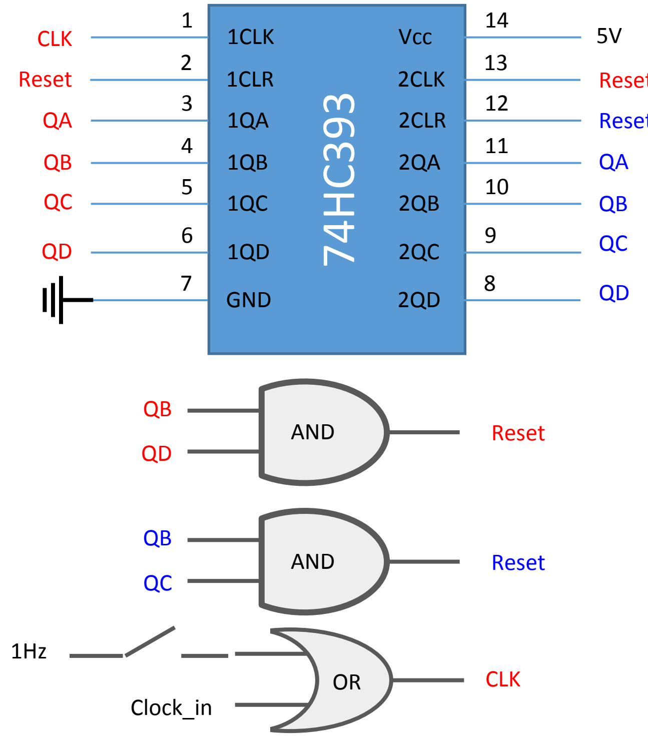
- Clock_in for the tens of seconds is the Reset from the seconds
- If Qb and Qc are both HIGH (number 6) we reset the counter.
- Reset signal is for the next stage Clock_in (minute Clock_in)



- OR element is for the later applications.

Design explanation - minute indication

- The minute indication works exactly as the second indication.
- One small addition would be needed compared the second - time adjustment.
- To adjust time we introduce a button that feeds 1Hz signal to the minute CLK. With this small add on we can adjust time manually.
- It should be done with an OR element.
- Reset signal is for the next stage Clock_in (hour Clock_in)



Design explanation - hour indication

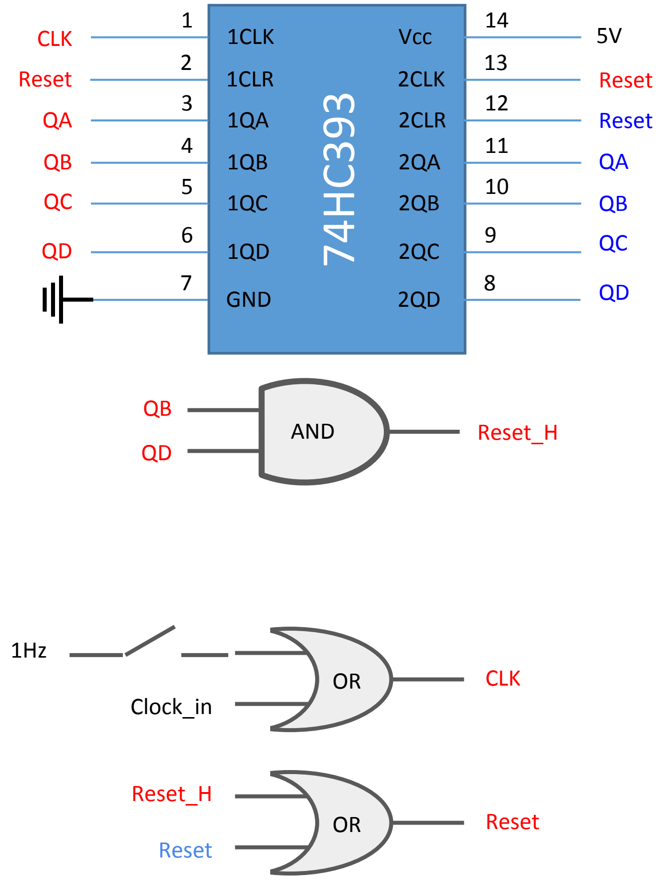
- The hour indication is slightly more complicated as the minute/second panel
- 12 hour time indication:
  - Hour indication (h - in the panel) should reset when :
    - H = 10 (that part we already know)
    - h H = 13
  - Tens of Hour indication (H - in the panel) should reset when :
    - h H = 13 → H can be only 0 or 1
- 24 hour time indication:
  - Hour indication (h - in the panel) should reset when :
    - H = 10 (that part we already know)
    - h H = 25
  - Tens of Hour indication (H - in the panel) should reset when :
    - h H= 25 → H can be only 0, 1 or 2

COUNT	OUTPUTS			
	Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H
10	H	L	H	L
11	H	L	H	H
12	H	H	L	L
13	H	H	L	H
14	H	H	H	L
15	H	H	H	H

h H : m M : s S

Design explanation - hour indication

- The hour (H-panel) indication works exactly as the second/minute (M- or S-panel) indication.
- Similar to the minutes we need time adjustment possibility.
- To adjust time we introduce a button that feeds 1Hz signal to the hour CLK.
- Additionally we need the possibility to reset the hour counter when hH = 12 or hH = 24
- Note that the 12h system the hours always go 11, 00, 01. With a standard electronic clock it goes 11, 12, 01.



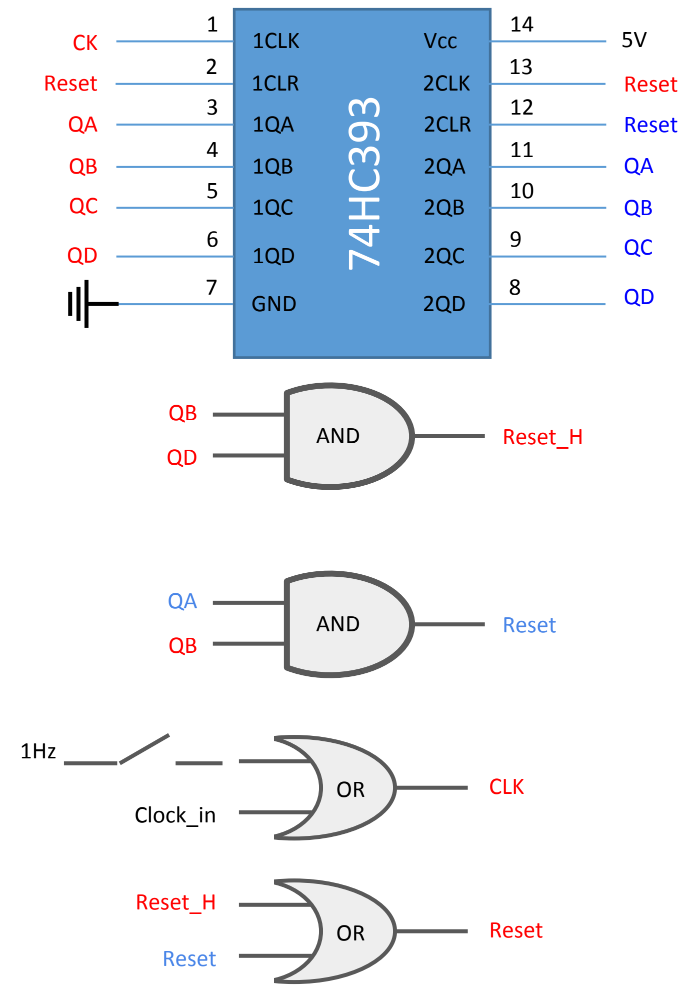
Design explanation - hour indication

- 12 hour time indication:
    - Tens of hour should be reseted if Hh = 12 ⟶
        - h - QA is HIGH
        - H - QB is HIGH
- 24 hour time indication:
    - Tens of hour should be reseted if Hh = 24⟶
        - h - QB is HIGH
        - H - QC is HIGH
-

COUNT	OUTPUTS			
	Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H
10	H	L	H	L
11	H	L	H	H
12	H	H	L	L
13	H	H	L	H
14	H	H	H	L
15	H	H	H	H

Design explanation - hour indication

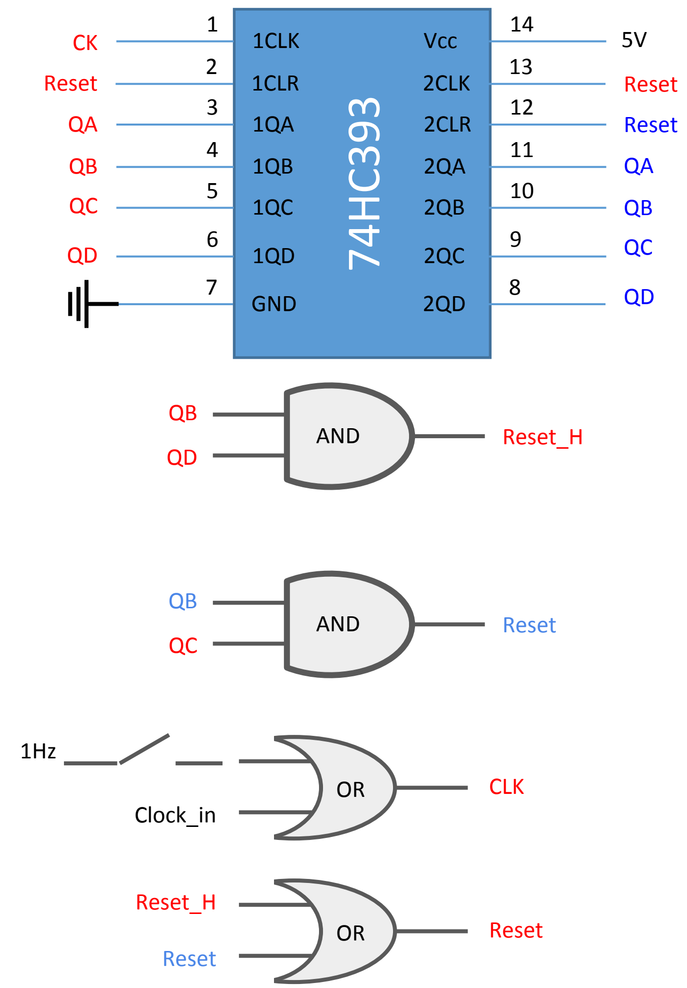
- 12 hour time indication:
  - Tens of hour should be reseted if hH = 12 →
    - h - QA is HIGH
    - H - QB is HIGH



h H : m M : s S

Design explanation - hour indication

- 24 hour time indication:
  - Tens of hour should be reseted if Hh = 25⟶
    - h - QB is HIGH
    - H - QC is HIGH



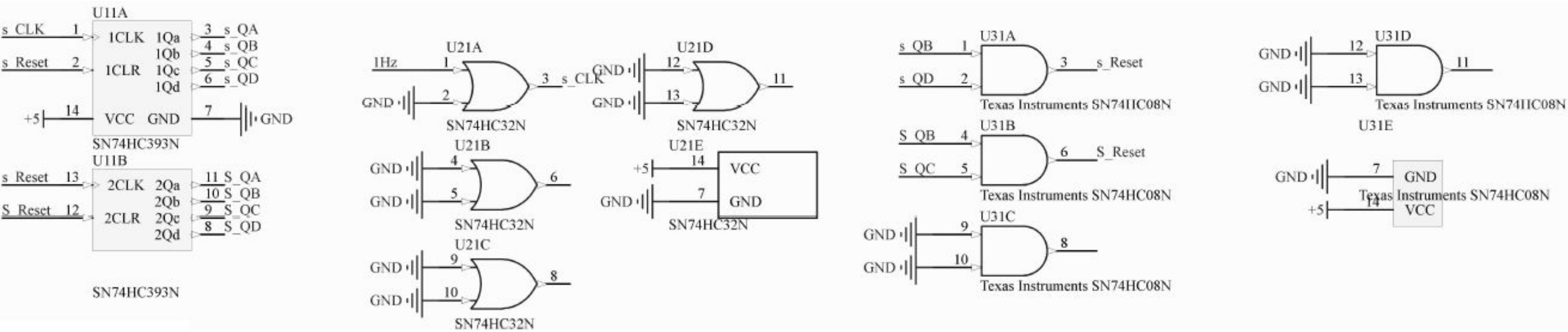
Design explanation

- As we can see the design of the second and minute panels are almost the same.
- The hour panel is slightly different, nevertheless still very similar to the other two.
- Thus, we could use it and make unified design for each panel:
  - The simplification will help us to reduce the PCB size
  - The PCB price goes drastically down
  - The unified panel design schematic is together with the Gerber file.
- In the following the schematics of second, minute and hour panel is shown

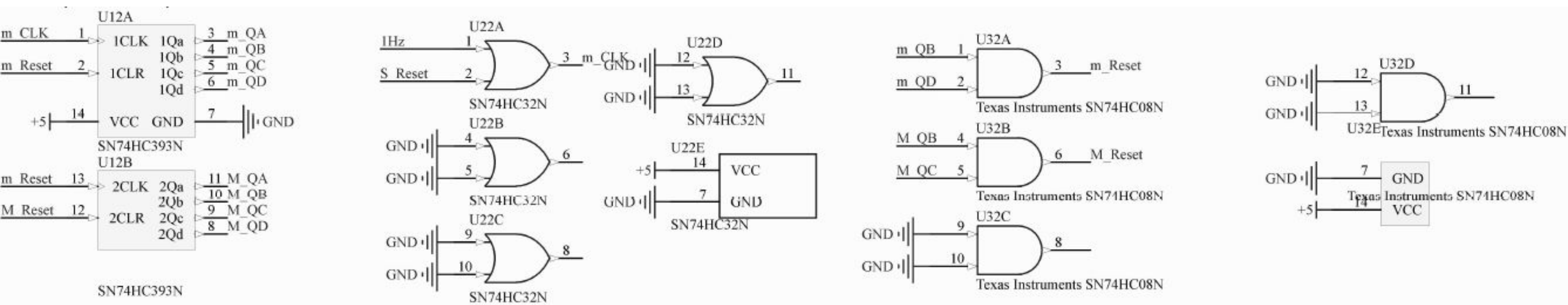
Design explanation

Note: not all necessary components are shown! Missing are some some switches and pull down resistors.

Second panel



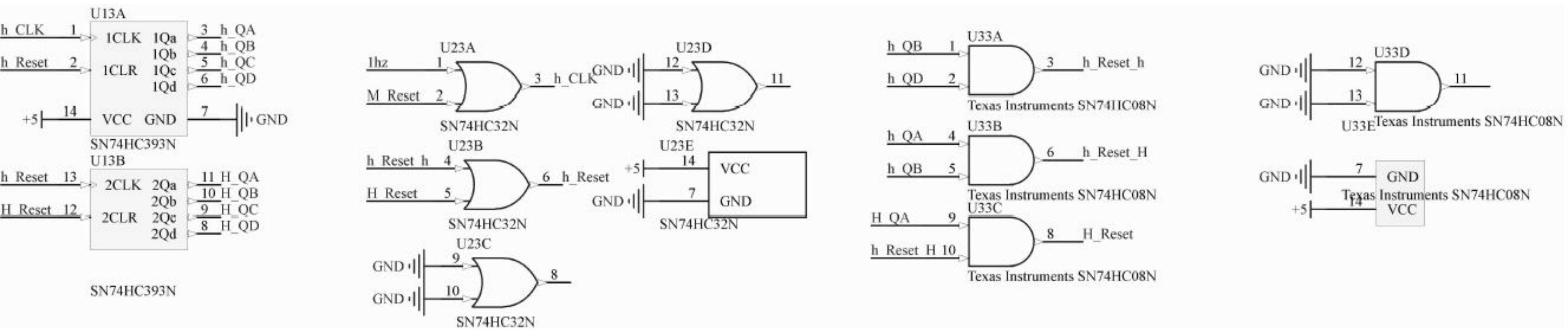
Minute panel



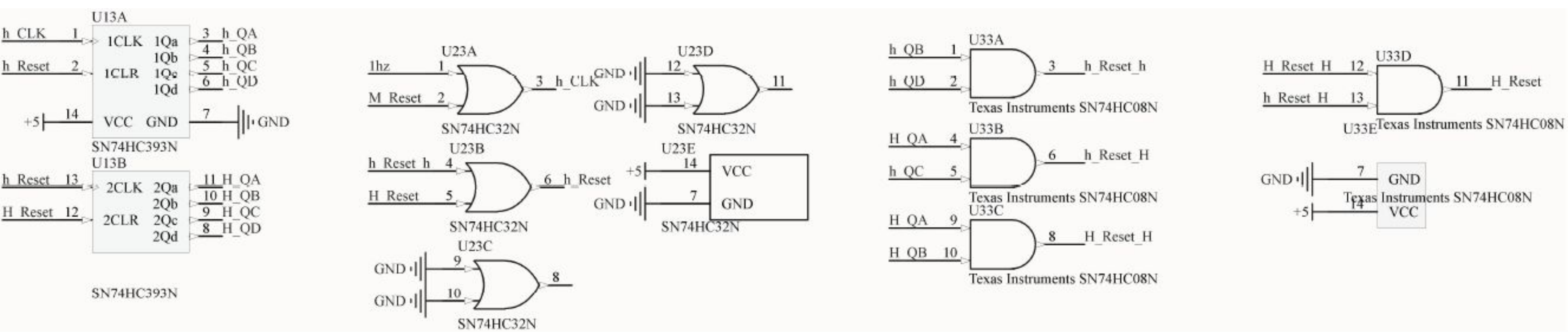
Design explanation

Note: not all necessary components are shown! Missing are some some switches and pull down resistors.

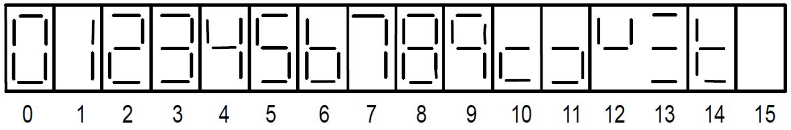
12h Hour panel



24h Hour panel



Design explanation - 7 segment



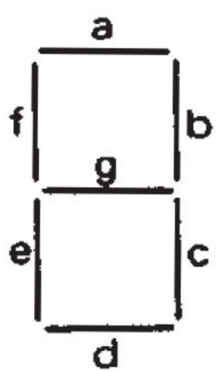
NUMERICAL DESIGNATIONS — RESULTANT DISPLAYS

Images from 74LS47 datasheets:

- Link1
- Link2

DECIMAL OR FUNCTION	INPUTS LT	RBI	D	C	B	A	BI/RBO	OUTPUTS a	b	c	d	e	f	g	NOTE
0	H	H	L	L	L	L	H	L	L	L	L	L	L	H	A
1	H	X	L	L	L	H	H	H	L	L	H	H	H	H	A
2	H	X	L	L	H	L	H	L	L	H	L	L	H	L	
3	H	X	L	L	H	H	H	L	L	L	L	H	H	L	
4	H	X	L	H	L	L	H	H	L	L	H	H	L	L	
5	H	X	L	H	L	H	H	L	H	L	L	H	L	L	
6	H	X	L	H	H	L	H	H	H	L	L	L	L	L	
7	H	X	L	H	H	H	H	L	L	L	H	H	H	H	
8	H	X	H	L	L	L	H	L	L	L	L	L	L	L	
9	H	X	H	L	L	H	H	L	L	L	H	H	L	L	
10	H	X	H	L	H	L	H	H	H	H	L	L	H	L	
11	H	X	H	L	H	H	H	H	H	L	L	H	H	L	
12	H	X	H	H	L	L	H	H	L	H	H	H	L	L	
13	H	X	H	H	L	H	H	L	H	H	L	H	L	L	
14	H	X	H	H	H	L	H	H	H	H	L	L	L	L	
15	H	X	H	H	H	H	H	H	H	H	H	H	H	H	
BI	X	X	X	X	X	X	L	H	H	H	H	H	H	H	B
RBI	H	L	L	L	L	L	L	H	H	H	H	H	H	H	C
LT	L	X	X	X	X	X	H	L	L	L	L	L	L	L	D

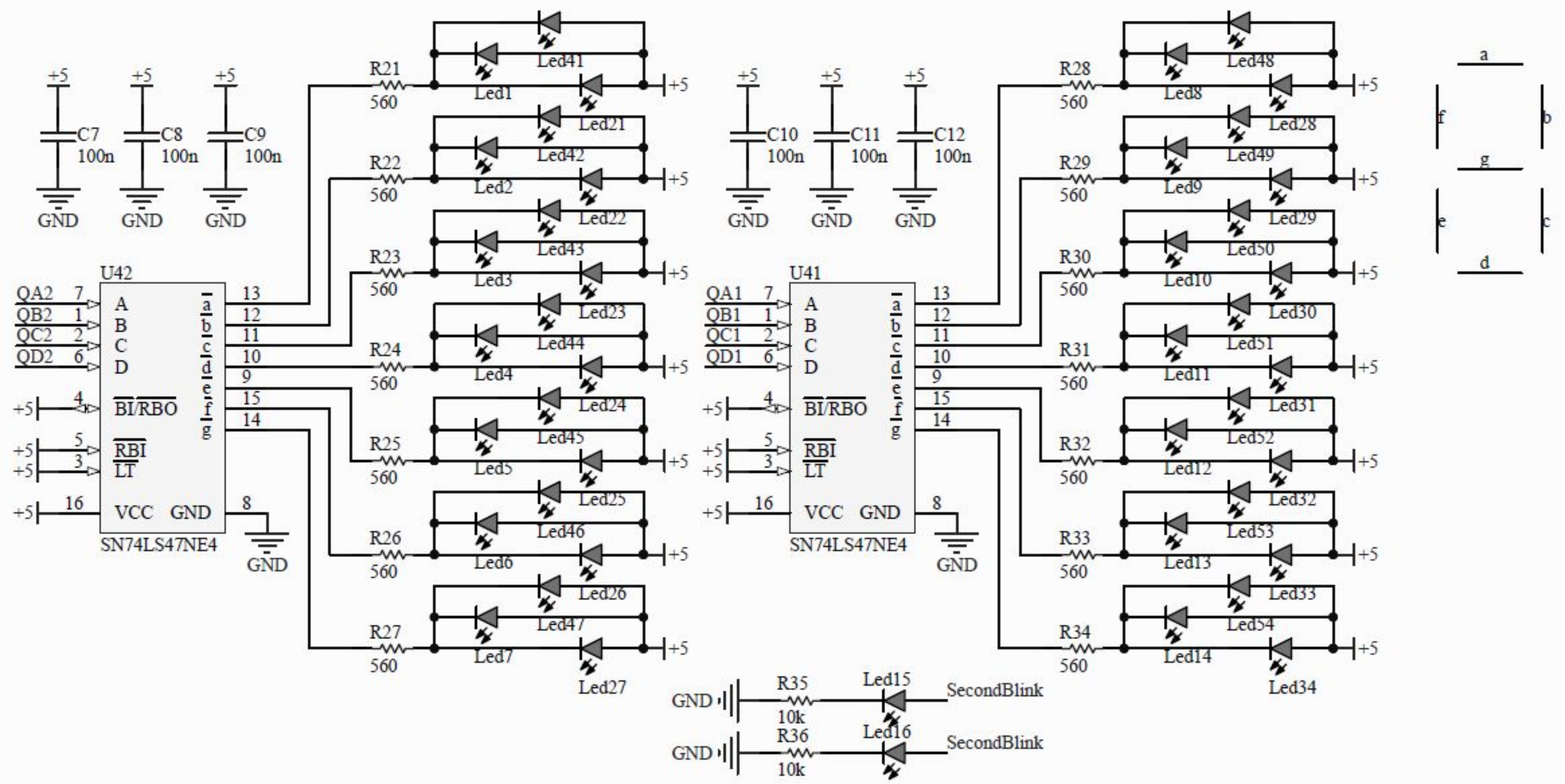
H = HIGH Voltage Level
L = LOW Voltage Level
X = Immaterial



SEGMENT IDENTIFICATION

Design explanation - 7 segment

LEDs



Design explanation - maximum allowed voltage

- Working voltage levels:
  - NE555 → 4.5 - 18V
  - 74HC393 → 2 - 6V
  - 74HC32 → 2 - 6V
  - 74HC08 → 2 - 6V
  - 74LS47 → 4.75 - 5.25V (output pins can handle more)
- Here we can see that the operating voltage is 5V and that’s what we will be using.
- No voltage regulators on the board, thus only 5V is allowed!
- I used an old USB cable

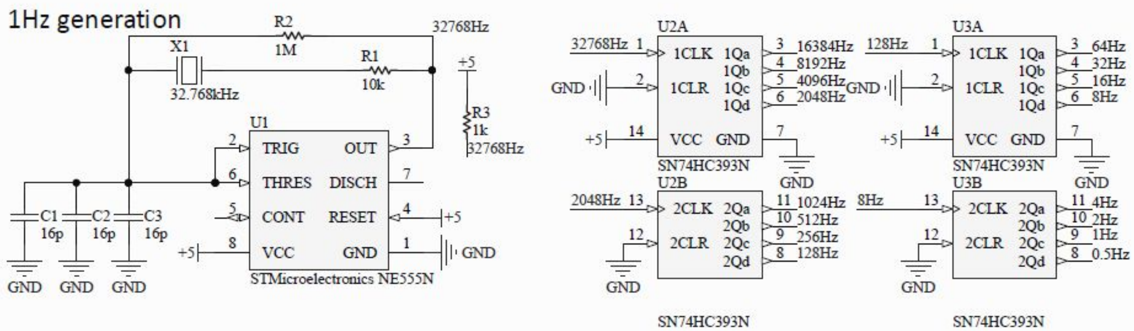
Design explanation - power supply requirements

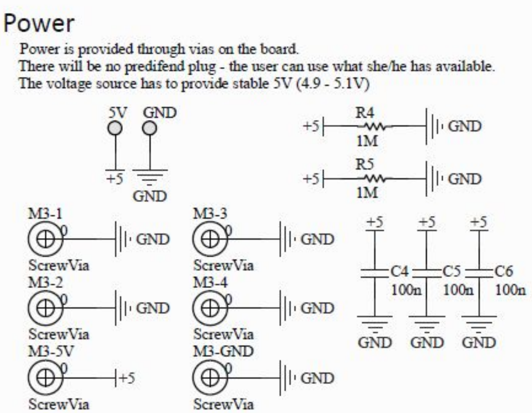
- Each LED segment current depends on the used resistor (R_{21} - R_{36}):
  - $I_{LED} = V / R = (V_{in} - V_{fLED}) / R = (5 - 1.5) / R = 3.5 / R$ [$V_{fLED} = 1.5V$ is an approximation!]
  - $R = 560\Omega \rightarrow I_{LED} = 3.5 / 560 = 6.25$ mA
  - $R = 200\Omega \rightarrow I_{LED} = 3.5 / 200 = 17.5$ mA
- Total LED current (theoretical as it will never happen that all LEDs are ON → 88:88:88):
  - I_{max_LED} = Nr_segments * Nr_numbers * $I_{LED} = 7 * 6 * I_{LED} = 42 * I_{LED}$
  - $R = 560\Omega \rightarrow I_{max\_LED} = 42 * 6.25$ mA = 0.263 A
  - $R = 200\Omega \rightarrow I_{max\_LED} = 42 * 17.5$ mA = 0.735 A
- We can assume that the calculated current is the max current consumed by the whole board.
- Computer USB port can provide max 0.5A, thus the LED resistors (R_{21} - R_{36}) have to be more than 320Ω if a computer USB port is used.
- If a regular phone charger is used LED resistors (R_{21} - R_{36}) can be as low as 200Ω.
- If lower resistor values are used the LEDs shine brighter!

Design explanation - PCB

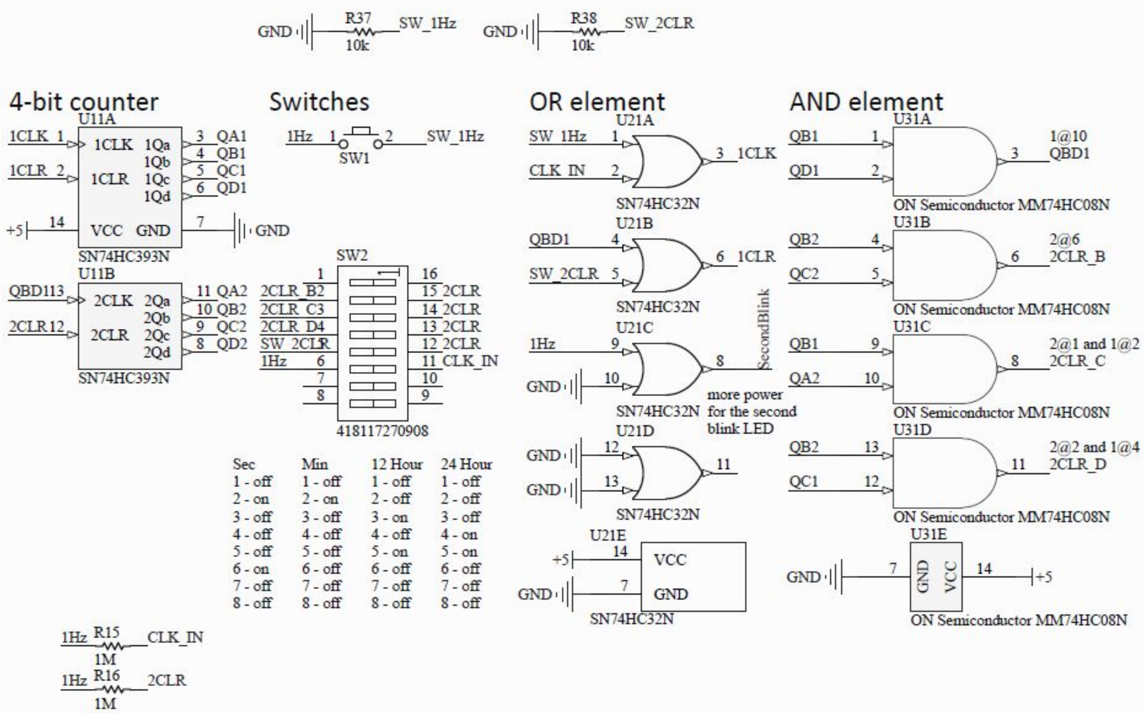
- As one could see from the different schemas that they are very similar.
- It would be beneficial to combine all the panels on one PCB.
- I have designed a 100x100mm PCB that contains a 1Hz signal generation and one panel (hH, mM or sS). Like this I was able to order 5 cheap PCB from china. Obviously one could make one large PCB, but the price goes drastically up.
- The PCBs are built up very similarly, but only the second indication PCB has the 1Hz circuit on it.
- Hour and minute PCB has the tactile switch.
- The components left and right from the lines are placed in the end. First all the components between the lines are soldered.

Design explanation - Final Schema - 1 Hz generation



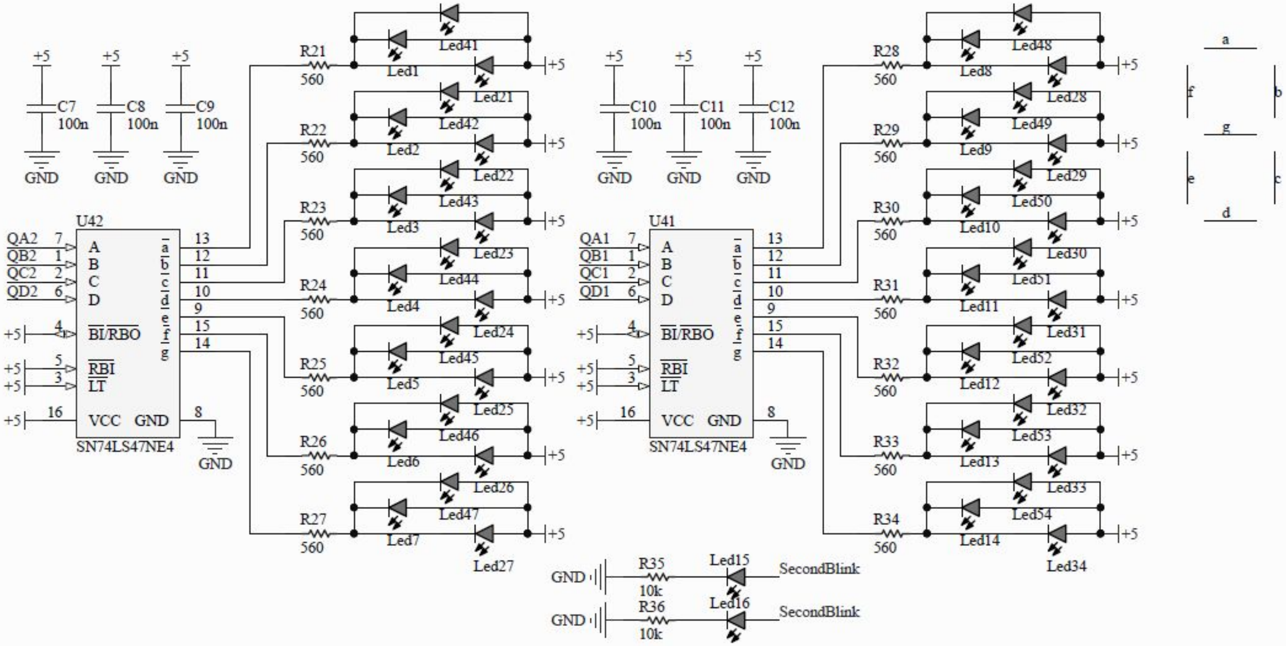


Design explanation - Final Schema - counters

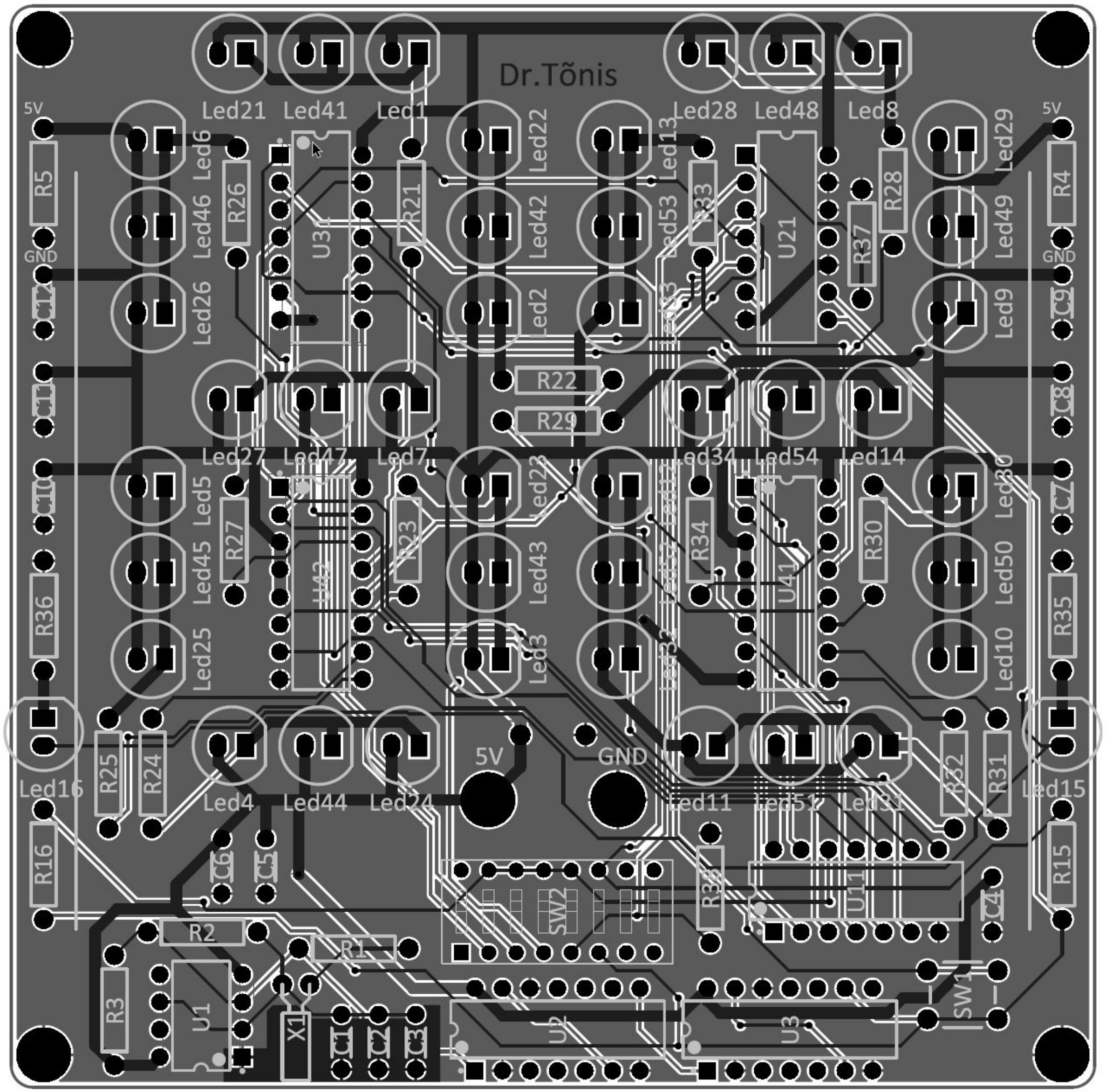


Sec	Min	12 Hour	24 Hour
1 - off	1 - off	1 - off	1 - off
2 - on	2 - on	2 - off	2 - off
3 - off	3 - off	3 - on	3 - off
4 - off	4 - off	4 - off	4 - on
5 - off	5 - off	5 - on	5 - on
6 - on	6 - off	6 - off	6 - off
7 - off	7 - off	7 - off	7 - off
8 - off	8 - off	8 - off	8 - off

Design explanation - Final Schema - LEDs



Design explanation - Final PCB



Design explanation - PCB how to solder (1)

- Start with the PCB for the seconds.
- Solder everything except:
  - R4 - R5
  - C7 - C12
  - R15 - R16
  - R35 - R36
  - Led15 - Led16
  - SW1
  - C3
- If you have an oscilloscope the you could play around with C1-C3 and try different combinations to achieve the closest frequency of 32’768Hz.
- I would start by soldering only 1 LED for each segment.
- After applying power and adjusting SW2 one could use it as an Hour, a Minute or a Second PCB.
- If the second counter goes as expected solder the remaining segment LEDs.

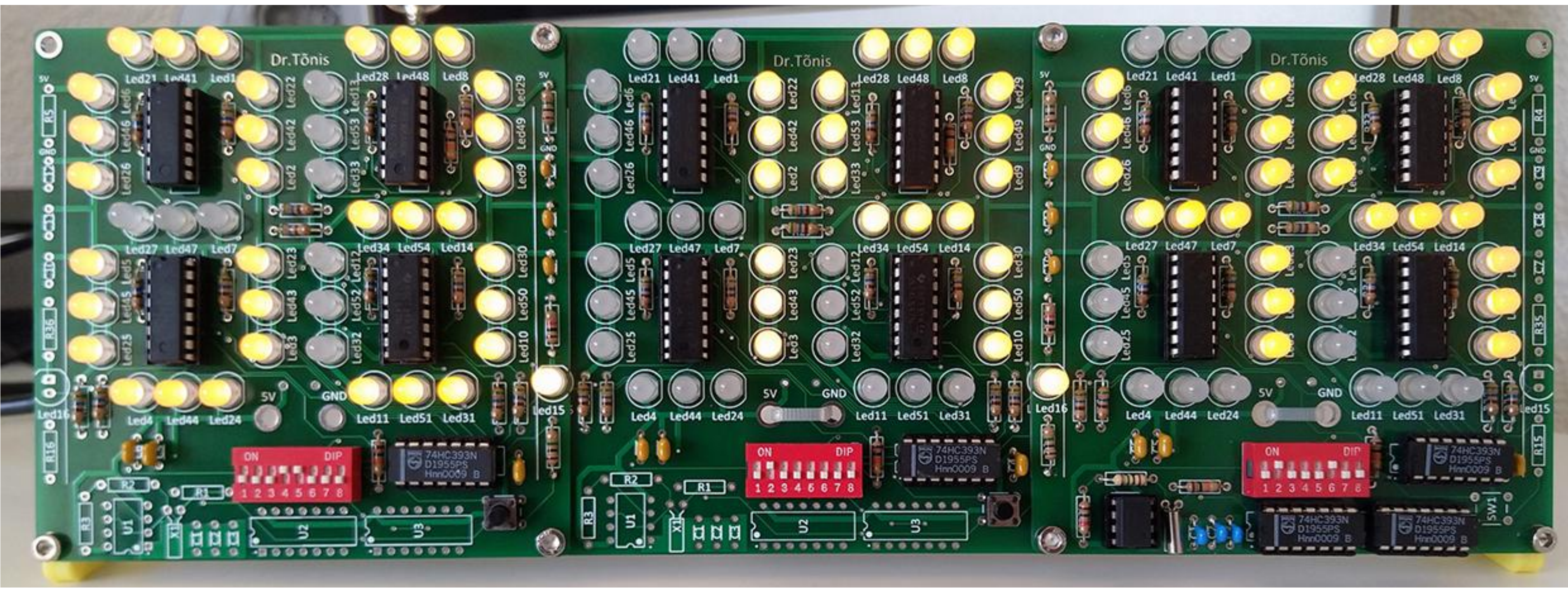
Design explanation - PCB how to solder (2)

- Now we can build the Hour and the Minute PCB
- Solder everything except:
    - R4 - R5
    - C7 - C12
    - R15 - R16
    - R35 - R36
    - Led15 - Led16
    - C1 - C3
    - R1 - R3
    - U1 - U3
    - X1
- Screw the panels together (look the following image), but don't tighten the screws too much. Now it’s important that the panels stay together, but one can still slightly move them.

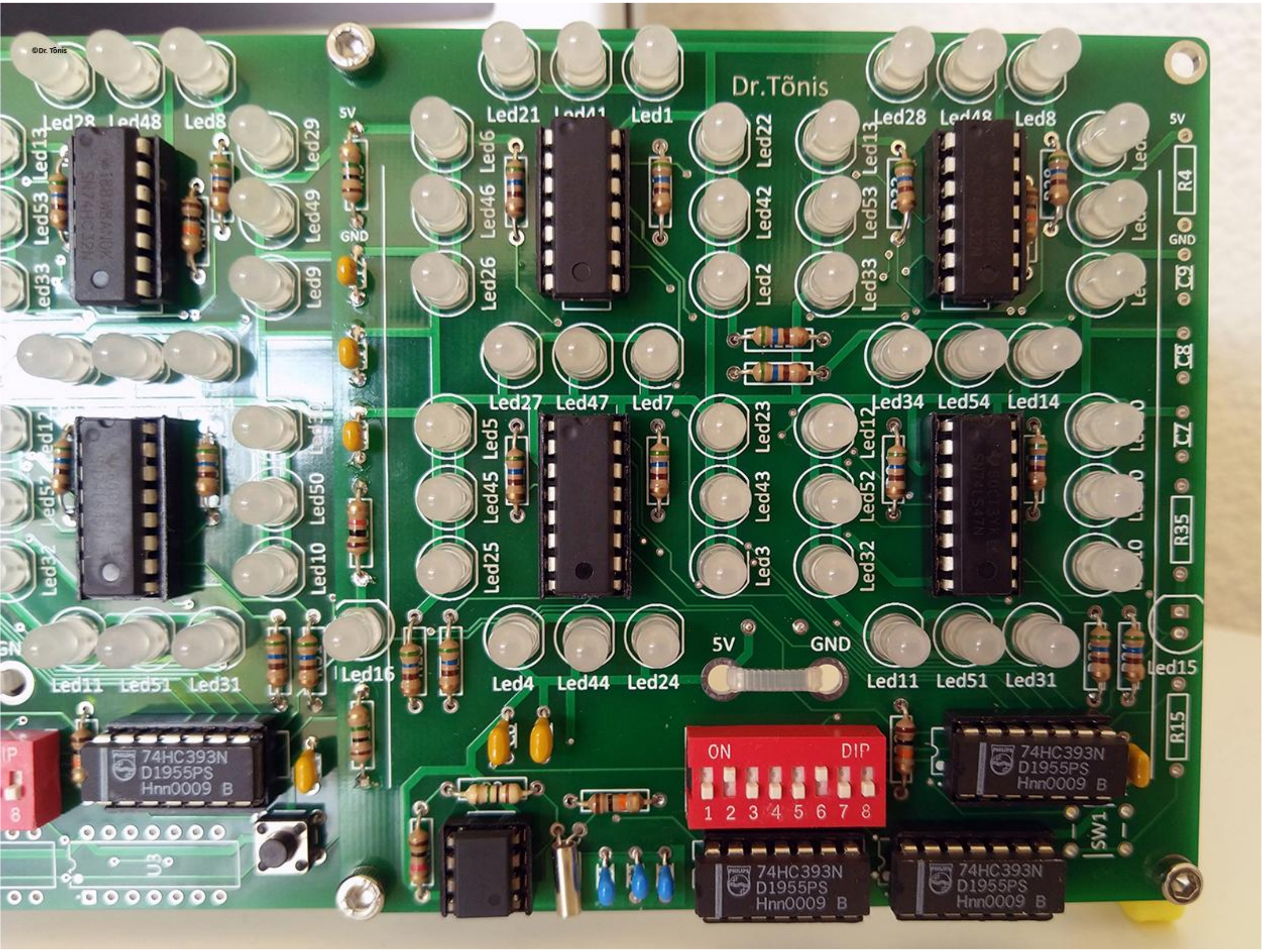
Design explanation - PCB how to solder (3)

- Solder the missing components only on the Minute PCB (look the following image):
    - R4 - R5
    - C7 - C12
    - R15 - R16
    - R35 - R36
    - Led15 - Led16
- Solder the components from both sides. So that there is a good contact with between the component and both PCBs.

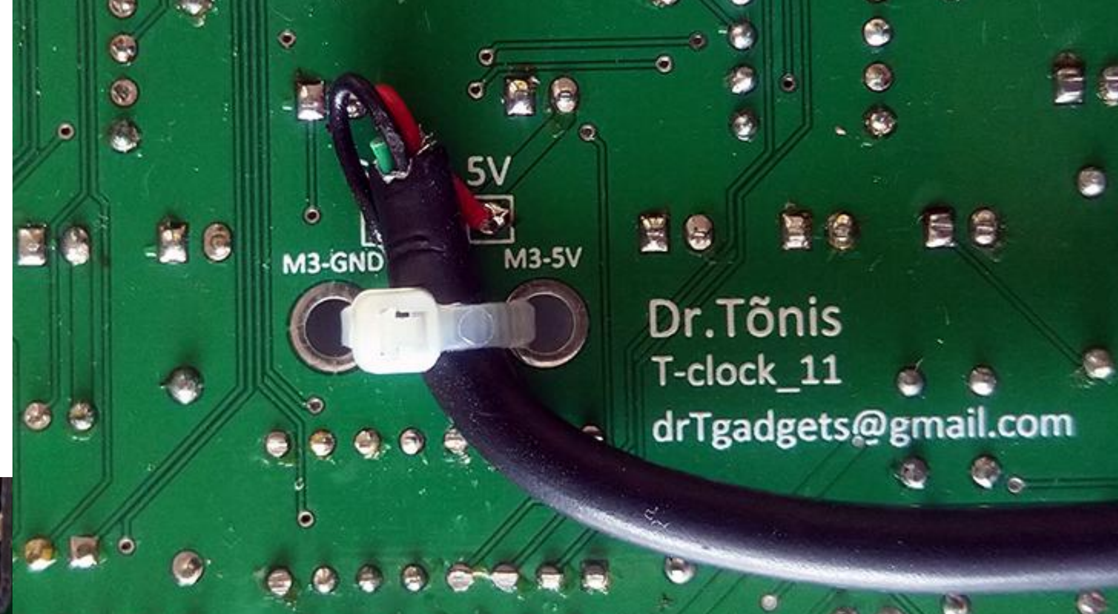
Final Clock

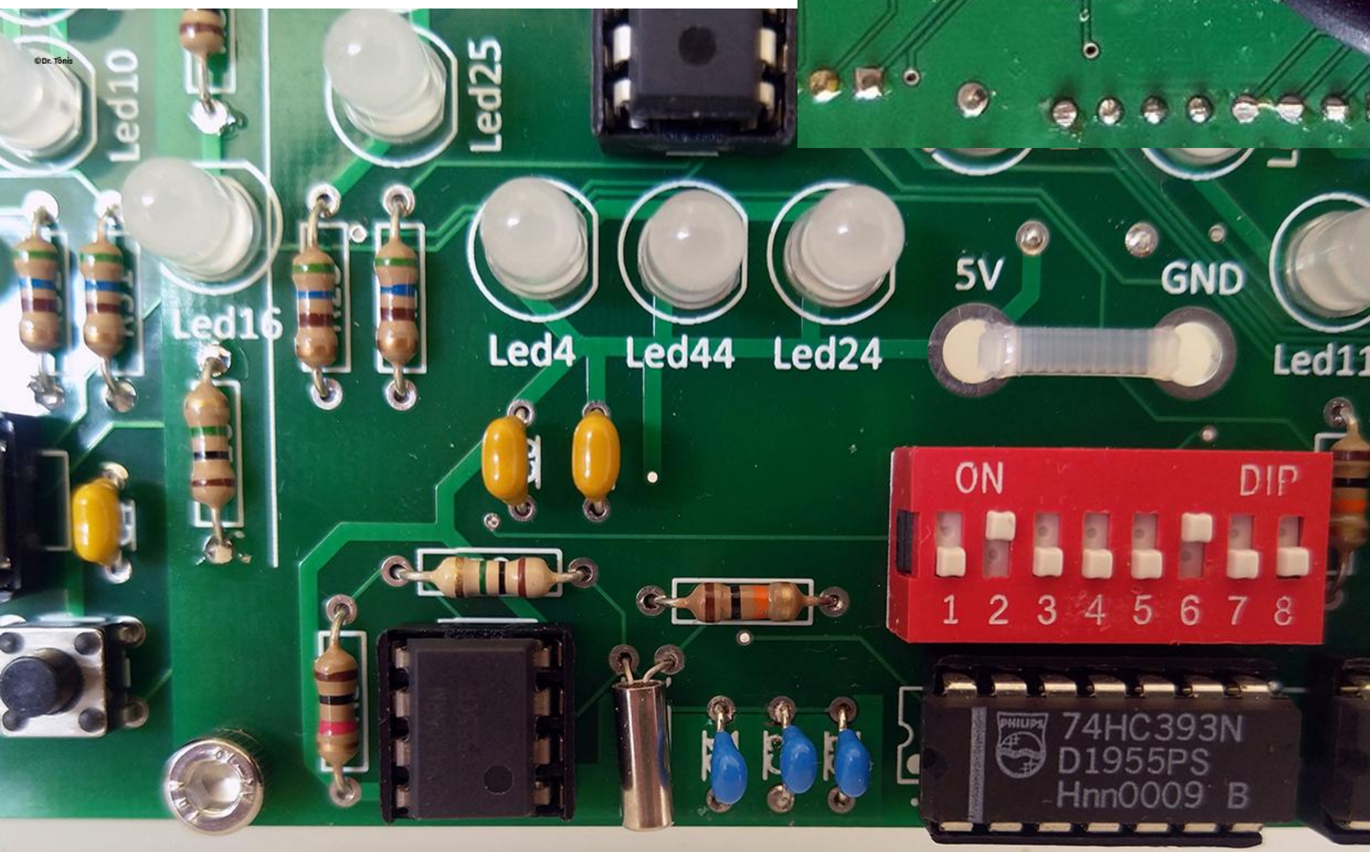


Final Clock



Final Clock


5V
M3-GND
M3-5V
Dr.Tõnis
T-clock_11
drTgadgets@gmail.com


Led10
Led25
Led16
Led4
Led44
Led24
5V
GND
Led11
ON
DIP
1 2 3 4 5 6 7 8
74HC393N
D1955PS
Hnn0009 B

Conclusion

- I hope this instructable was helpful and interesting to read.
- Maybe it can inspire other people to use counters or build their own clock.

- If you would like to purchase the PCB gerber files then follow this link to my etsy shop:
  https://www.etsy.com/shop/DrTonis?ref=seller-platform-mcnav

Contact

The most convenient way to contact me is the email: drTgadgets@gmail.com

If you are interested to see other projects from me: Dr. Tõnis’ blog

or my Instructables page: Dr. Tõnis Instructables

Homepage:

Email: