INSTITUTO TECNOLÓGICO DE BUENOS AIRES

# Trabajo Práctico de Laboratorio Nro. 2

NOWIK, Ariel 58309
MASPERO, Martina 57120
MESTANZA, Joaquin 58288
REGUEIRA, Marcelo 58300
October 16, 2018

# Task 3

Given a truth table, it was requested to see what happens when the truth table is implemented with the smallest quantity of logic gates. To perform this task, ICs that either have NOR or NAND logic gates will be used.

## Low cost approach

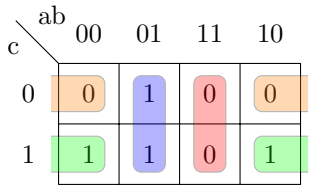


Figure 1: Karnaugh Map of the given truth table

We can get the minimal cost output expressed either in groups of maxterms (orange and red groups) or minterms (blue and green groups).
Let Z be the output label, its reduced minterms expression is:

$$Z = (\bar{A}.B) + (C.\bar{B}) \tag{1}$$

Also, its reduced maxterms expression is:

$$Z = (\bar{A} + \bar{B}).(C + B) \tag{2}$$

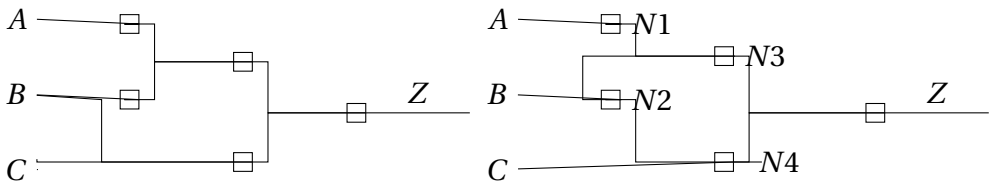


Figure 2: Implementation with NOR gates and NAND gates respectively

Note that the quantity of logic gates in both cases are the same.

# Glitch analysis

Sometimes propagation delays cause unexpected and unwanted transition in the ouput. This issue is called 'glitch' and it's related to how the logical circuit is synthesized. In this section, the NAND implementation will be taken into account. If we take into accout the propagation time of the NAND logic gate (here we assume all propagation times are equal) and then observe transition of the states A,B,C from 011 to 001 respectively.

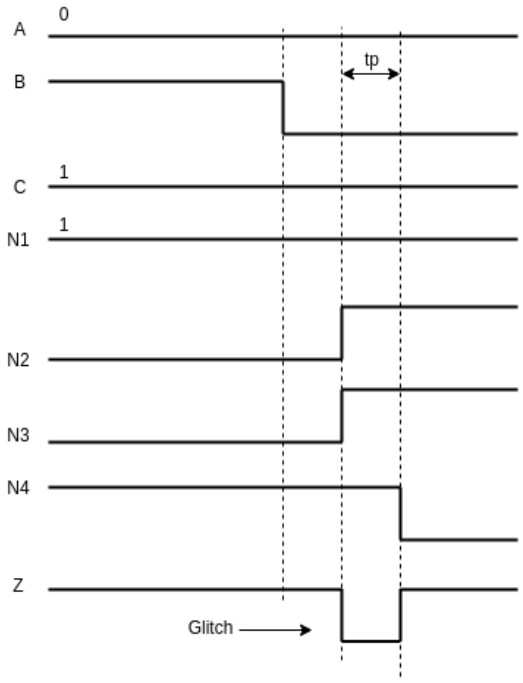


Figure 3: Propagation time analysis

Note that the interval of time tp in Figure 3 denotes the propagation time. The result might be at first surprising but in fact this effect was in someway expected. Take a look at the Karnaugh map in Figure 1. An input that follows multiple paths to an output can create a glitch if one path has an inverter (in this case implemented with a NAND logic gate) and one does not. This issue is called "asymmetric path delay" and it can be seen in Figure 1 with the input B.

# Conclusion

In order to remove glitches it is needed to add redundant groups in the Karnaugh Map so that disjoint groups overlap.

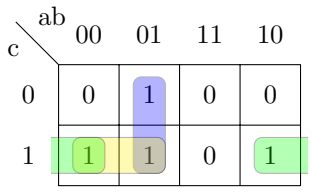


Figure 4: Improved karnaugh

In this case the output is given by:

$$Z = (\bar{A} + \bar{B}).(C + B) + (\bar{A}.C)$$

Note that the new term added to the expression is not dependent of the value of B so regardless of the variation of B there won't be glitches in the output, because it will be fixed by this new

term of sum of products.
Finally to give some sense to the adding some redundant logic, when we make Karnaugh groups we take the non-changing states and the only variable that changes of state in the new group is B which is reasonable due to the fact that that's the variable causing the issue. Depending on the application it may be important or not to remove glitches. There might be more glitches, remember that this is just one particular case, where glitches are more likely to happen.

# Task 4

In this section, there are measured the propagation delay, rise time and fall time of a 74HC02 logic gate (CMOS tecnology) in several configurations: without load, and with a circuit load as shown below.

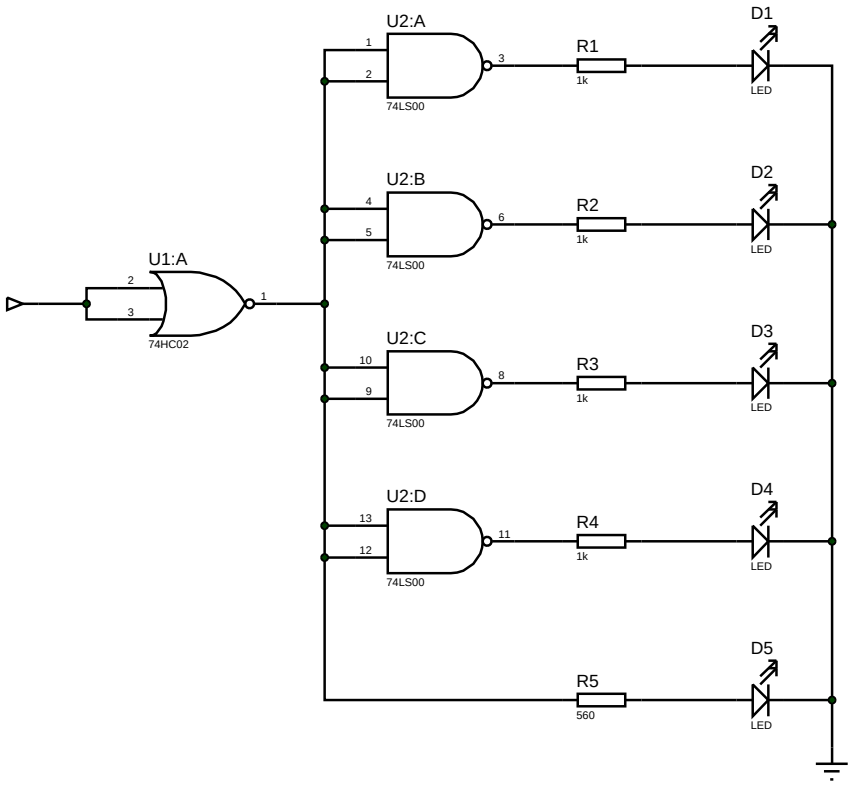


Figure 5: Circuit load schematic - Made in Proteus 7.8

In the following table are the measured times for the diferent configurations.

| CASE | $tpd_{L-H}$ | $tpd_{H-L}$ | $t_r$ | $t_f$ |
|---|---|---|---|---|
| Without LOAD | | | | |
| Circuit LOAD | | | | |
| Circuit LOAD (100KHz) | | | | |
| Circuit LOAD (100KHz with Capacitors) | | | | |

Table 1: Measured times

CONCLUSIONES DE TEMPERATURA, TIEMPOS Y EL PORQUE DE USAR 100nF.

# Task 6

In this section, a D Flip-Flop and a SR Latch are implemented, based on logic gates.

## SR Latch

For the SR Latch, it is implemented using NOR gates of 74HC02 integrated circuit. The resulting schematic is shown below.

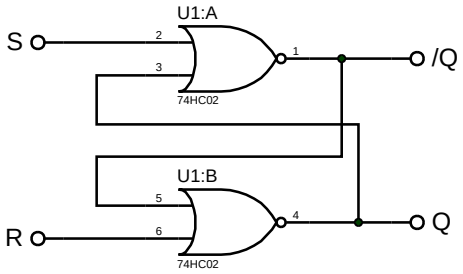


Figure 6: SR Latch circuit - Made in Proteus 7.8

PARAMETROS IMPORTANTES: PROPAGACION DE SET O RESET A LAS SALIDAS. MEDIR. COMPARAR CON UNO COMERCIAL CUALQUIERA

| PARAMETER | | FROM | TO | VALUE |
|---|---|---|---|---|
| $t_{pd}$ | Circuit | S | Q | |
| | 74HC279 | S | Q | 30ns o 26ns |
| $t_{pd}$ | Circuit | R | Q | |
| | 74HC279 | R | Q | 31ns o 36ns |

Table 2: Measured values

## D Flip-Flop

For the D flip-flop, it is implemented using the SR Latch designed before, adding the remaining parts as shown below.

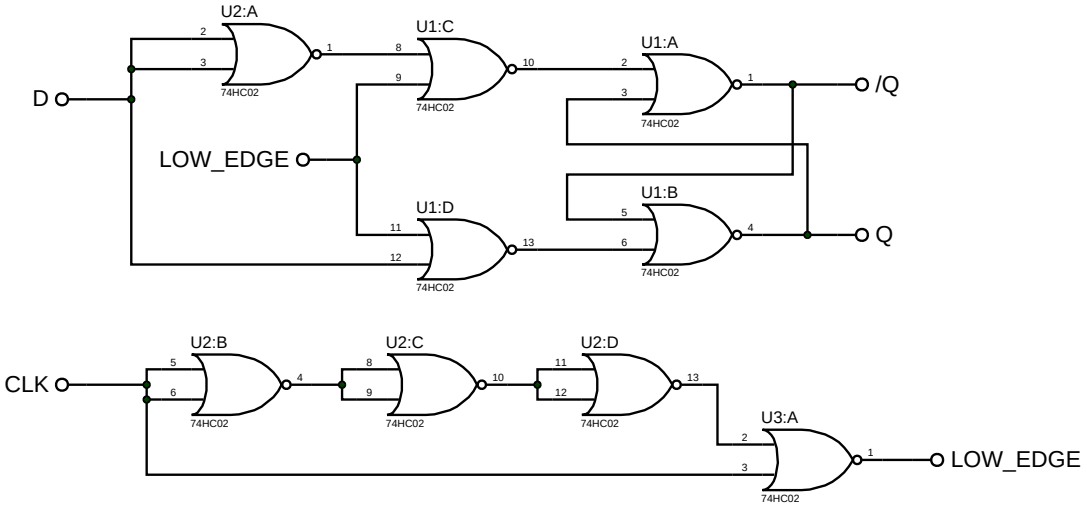


Figure 7: D Flip-Flop circuit - Made in Proteus 7.8

PARAMETROS IMPORTANTES: PROPAGACION DEL D A LA SALIDA LUEGO DEL CLK, TANTO PARA 0 COMO PARA 1. MEDIR. COMPARAR CON UNO COMERCIAL CUALQUIERA. The designed circuit will be compared with the 74HC74 D Flip-Flop, using the datasheet of Texas Instruments.

| PARAMETER | | FROM | TO | VALUE |
|---|---|---|---|---|
| $t_{pd}$ | Circuit | CLK | Q or /Q | |
| | 74HC74 | CLK | Q or /Q | 44ns o 37ns |
| $t_t$ | Circuit | | Q or /Q | |
| | 74HC74 | | Q or /Q | 19ns o 16ns |

Table 3: Measured values

# Task 7

In this task we're asked to implement one syncronomous, and one asynchronous 3-bit counter, and contrast them.

## Async. counter

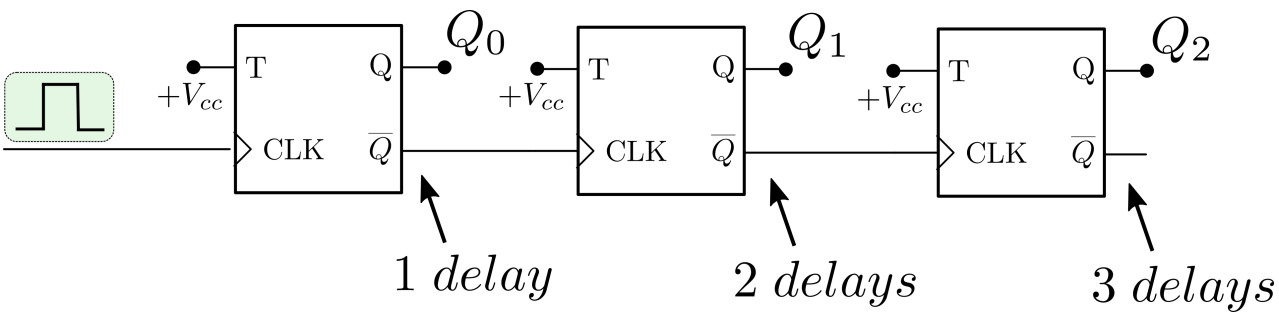


Figure 8: Async. counter - delays

This counter work with a casquade triggering flip flops clocks done by the not Q output.

The idea is that, everytime the left flip flop is clocked two times, not Q goes up one time, thus, right flip flop is clocked one time (the half) Propagating this logic to the general case, n-th flip flop has a period $2^n$ thus we made a binary counter.

The theorical problem of this system is the delay propagation. As there is a delay between clock rising edge and Q update, the total delay will be multiplied by the number of flip flops.

## Sync. counter

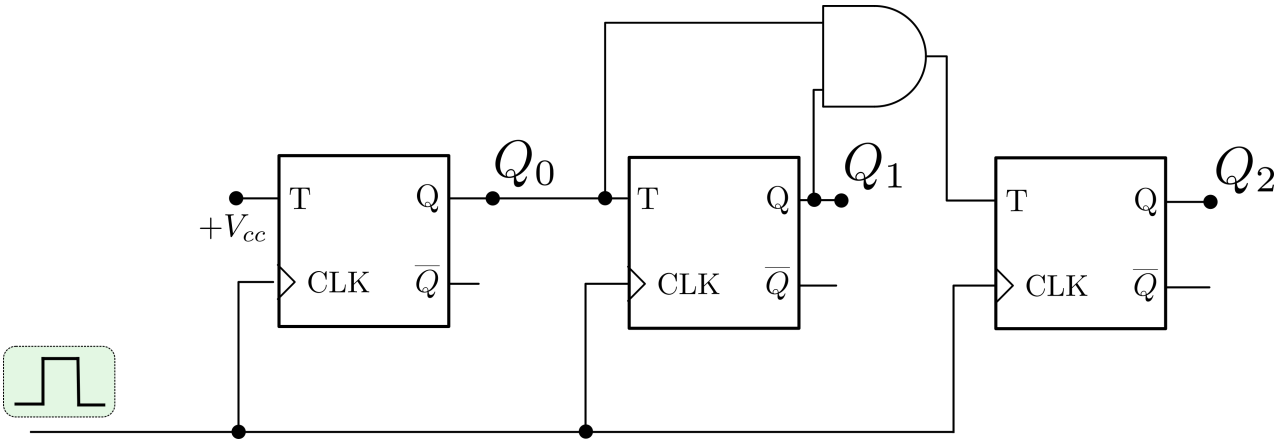


Figure 9: Sync. counter - delays

This counter is made to fix the problem of the previous counter. In this logic, all flip flops clocks are connected to the same trigger, thus, all of them are updated in the same instant, not, one after the other

# Task 8

In this section, a distance measurement system is implemented using discrete logic and the ultrasonic sensor HC - SR04. With it, distances between 1.7cm and 4.25m can be measured. The design is shown in the following block diagram.

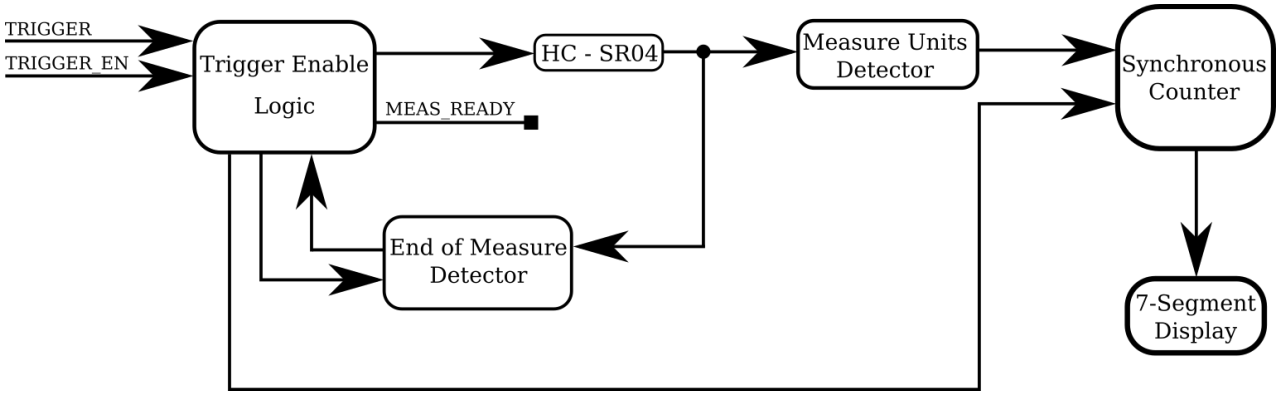


Figure 10: Distance measurement system - Block diagram

## Trigger Enable Logic

For this part, a T flip-flop is used to define two states: (1) Measure enabled and (2) Measuring.

In state (1), with TRIGGER_EN on HIGH level, a measure can be made by sending a positive edge to the TRIGGER input, then the flip-flop changes to state (2), preventing a new retrigger while measuring if a new positive edge is detected in TRIGGER input. This two states and the MEASURE_READY bit are referenced from the $\overline{Q}$ output of the flip-flop. When the measure ends, an edge is sended to the CLR pin of the flip-flop, for returning to state (1), allowing to make a new measure. The previus performance is shown in the following time diagram.

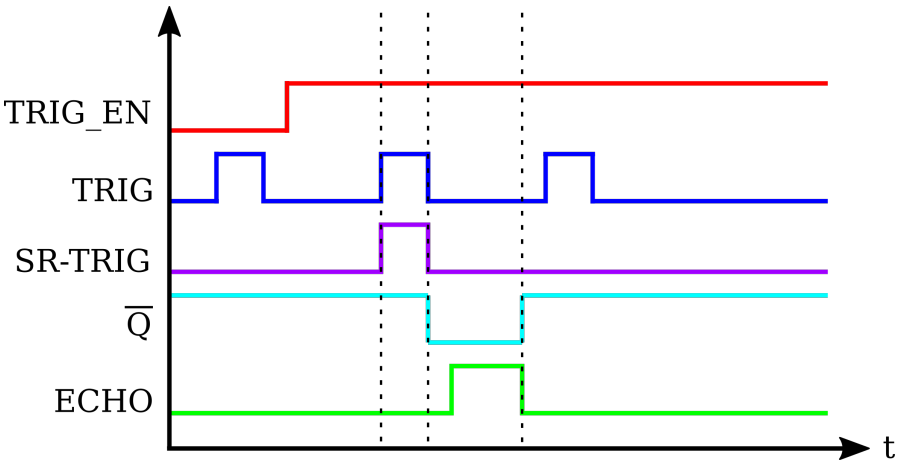


Figure 11: Trigger Enable Logic - Time diagram

The ultrasonic sensor needs a pulse of a period T > 10$\mu$Seg, so the edge of TRIGGER input is sended to a monostable circuit (implemented with a LM555 timer) to ensure that the condition is met. The final schematic of this section is shown below.

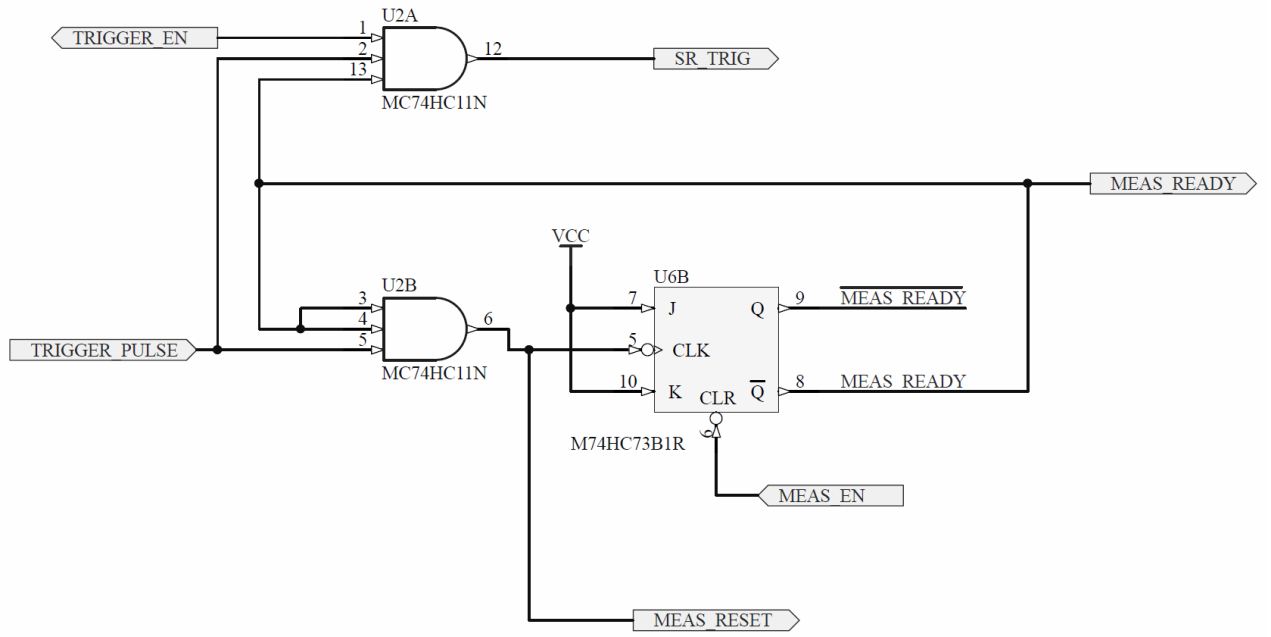


Figure 12: Trigger Enable Logic - Schematic circuit made in Altium 15

## HC - SR04 Sensor

To make the measures in the mentioned interval, the HC - SR04 after receiving the TRIG input pulse, it answers at the ECHO out with a pulse of a period T > 100$\mu$Seg (1.7cm), up to 25000$\mu$Seg (4.25m). It will be fractioned in units of 100$\mu$Seg (to be counted).

To know the measure value, the obtained number $N$ is processed in the following calculation:

$$Distance[m] = 170 \cdot 100 \cdot 10^{-6} \cdot N$$

By fractioning the ECHO pulse in discrete units, the resulting resolution is equivalent to 1.7cm, which is the smallest unit that can be counted (100$\mu$Seg).

## Measure units detector

For fractioning the ECHO response into units of 100$\mu$Seg, it is connected to an AND gate with a clock source (CLK) whose period is 100$\mu$Seg.
In this way, while the ECHO out stays at HIGH level, the signal at the AND gate output is equal to the clock source. While not measuring, the output stays at LOW level. The previous performance is shown in the time diagram below.

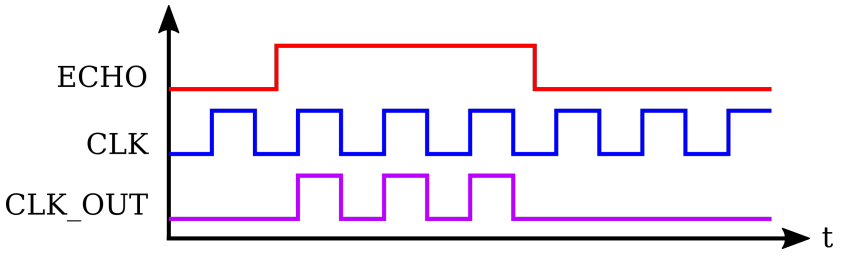


Figure 13: Measure units detector - Time diagram

The CLK signal was implemented with a LM555 timer in astable configuration. The schematic for this section is shown below.

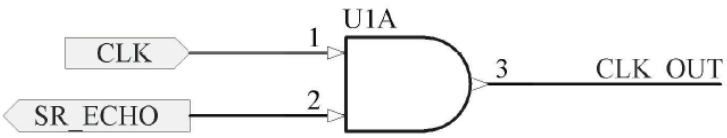


Figure 14: Measure units detector - Schematic circuit made in Altium 15

## Synchronous counter

For counting the time units, an integrated synchronous counter is implemented (CD4040), whose CLK input is conected to the out signal of the measure units detector. It detects the positive edges of the CLK signal, in this case having one per 100$\mu$Seg. From the 12-bit output, only 8-bit are used, because for measuring the maximum distance, only 250 untis are needed (25000$\mu$Seg over 100$\mu$Seg results in 250 units), and if the binary out is converted to decimal, the maximun number is $2^8 = 256$ (which covers the 250 maximum).
The integrated circuit implementation is shown below.

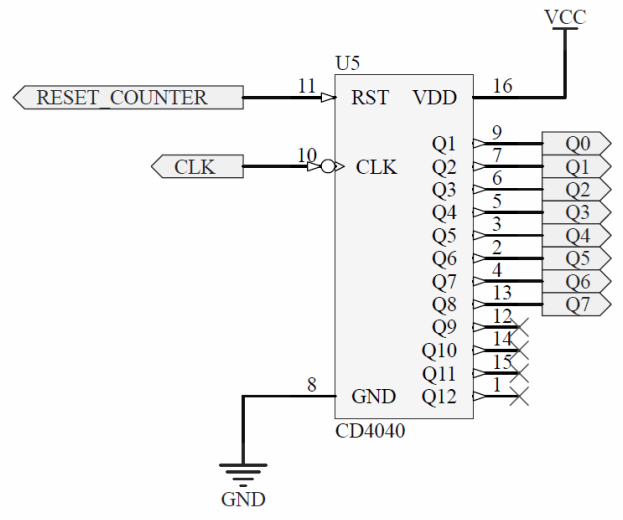


Figure 15: Synchronous counter - Schematic circuit made in Altium 15

## End of measure detector

When the measure ends, the negative edge produced by the ECHO is used in a discrete edge detector to reset the flip-flop in the trigger enable logic, so a new measure can be started at any time. The circuit diagram of the implementation is shown below.

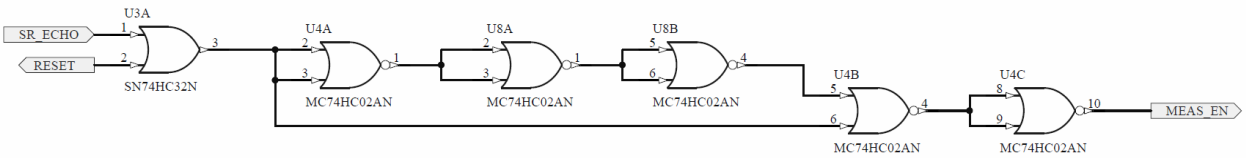


Figure 16: End of measure detector - Schematic circuit made in Altium 15

The binary output resulting from the measure stays on until a new positive edge from the trigger is received. Then, the pulse used to change the flip-flop state in a new measure is also used to reset the previous binary output from the synchronous counter.

## Manual reset

If when connecting the power supply to the circuit, the counter is not in cero, or the flip-flop starts switched into the state (2), a manual switch for reset is added with a pull-down resistor, as shown below.

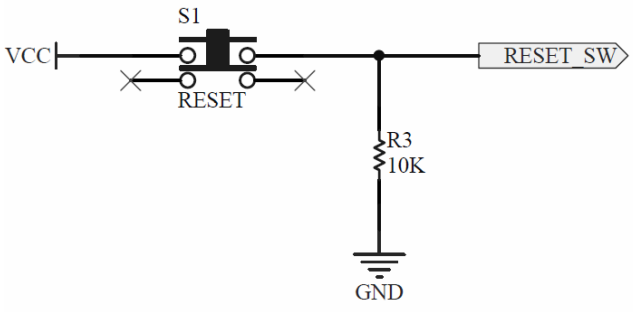


Figure 17: Manual reset with pull-down resistor - Made in Altium 15

## Aditional settings

### MEASURE_READY indicator

To know if the flip-flop starts in the correct state when connecting the power supply or if the last measure has finished, a bicolor LED was used through two transistors, using the $Q$ and $\overline{Q}$ outputs of the flip-flop. When a measure ends (or when at the power supply connection the flip-flop starts at the correct state), the LED turns green as an OK indication. When measuring, the LED turns red until it ends. If when connecting the power supply (or in any moment) the flip-flop starts in the wrong state, the LED will remain in red. This can be fixed by pressing the reset switch.
The schematic circuit is shown below.

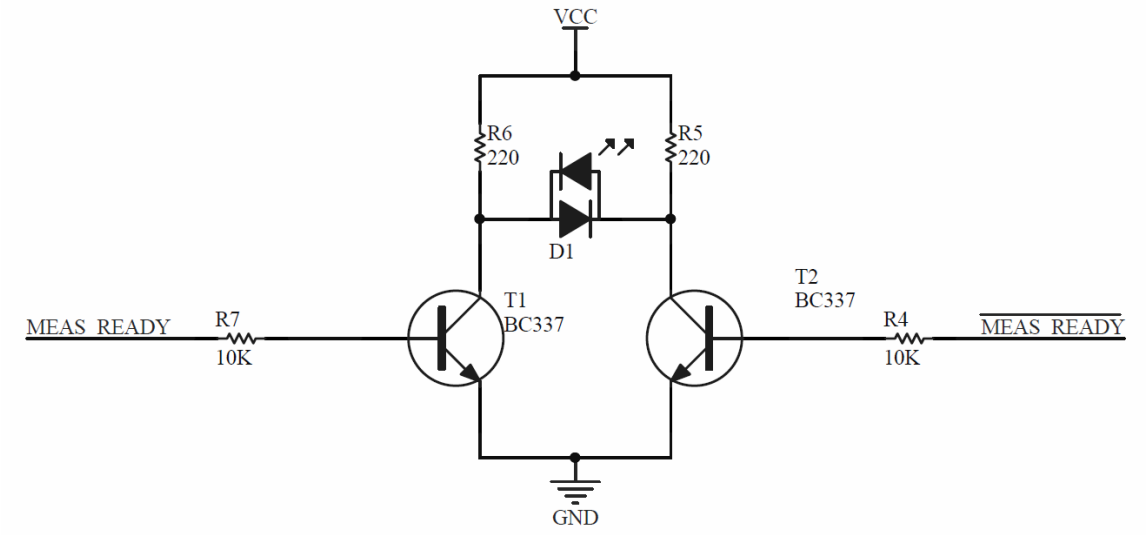


Figure 18: MEASURE_END indicator - Transistor logic made in Altium 15

The calculation of the resistors can be read from the Annex.

**Measure counted units indicator**

To obtain the amount of time units measured more easily, the CLK out of the measure units detector is used in decade counters implemented with CD4033, because their out pins are decoded for use in 7-segment displays. Since the maximum measure has 3 digits, there are 3 counters and displays implemented (their reset pin is connected to the reset signal of the end of measure detector, and to the reset switch through an OR gate). The implemented circuit schematic is shown below. It has been done in a separete PCB.

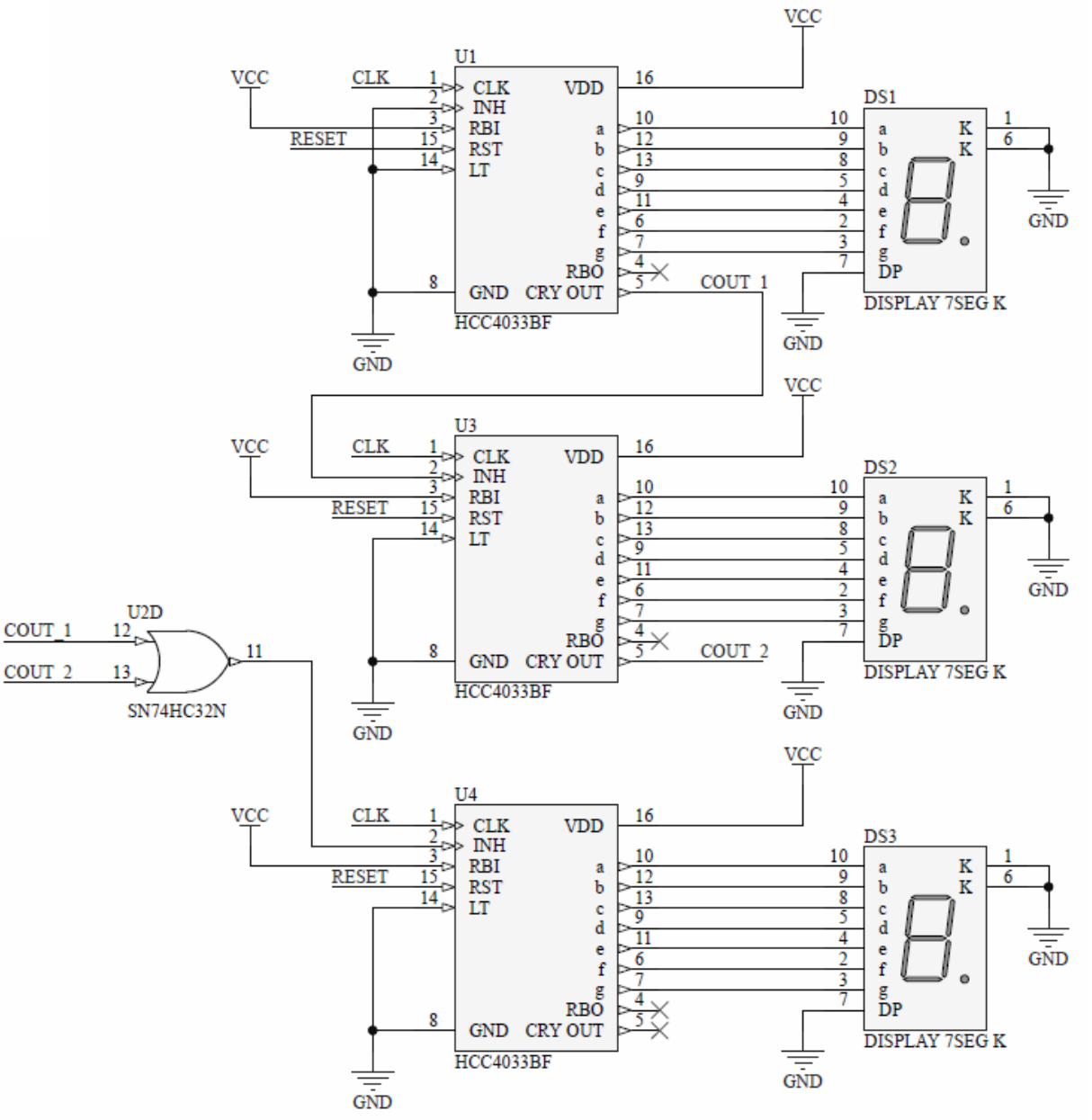


Figure 19: Measure display - Schematic circuit made in Altium 15

# Appendix

## MEASURE_READY indicator - Resistors calculation

For the circuit design, the transistors are used in saturation-cutt off mode. Considering a $15mA$ current for the LED, it will be the $IC_{SAT}$. Going through the out loop we have:

$$V_{CC} - IC_{SAT}R_C - V_{LED} - VCE_{SAT} = 0$$

Considering as $VCE_{SAT} = 0.2V$ and a $V_{LED} = 1.8V$, the value of $R_C$ (collector resistor) can be obtained as follows:

$$\frac{V_{CC} - V_{LED} - VCE_{SAT}}{IC_{SAT}} = RC = 200\Omega$$

Normalizing the value, it remains as $R_C = 220\Omega$. Recalculating the new $IC_{SAT}$:

$$\frac{V_{CC} - V_{LED} - VCE_{SAT}}{R_C} = IC_{SAT} = 13.6mA$$

Using the BC337 transistors, according to ON Semiconductor datasheet, the minimum HFE is 100, which will be used to calculate the minimum $IB_{SAT}$ (because its the worst case, to guarantee the saturation):

$$\frac{IC_{SAT}}{HFE_{MIN}} = IB_{SAT-MIN} = 136\mu A$$

With that value, the base resistors can be calculated, and then normalized down to get an $IB_{SAT}$ over the previous limit. Going through the entry loop:

$$V_{Q-OUT} - VBE_{ON} - IB_{SAT}R_B = 0$$

Considerating that $VBE_{ON} = 0.7V$ and the output voltage of the flip-flop as $5V$ on HIGH level:

$$\frac{V_{Q-OUT} - VBE_{ON}}{IB_{SAT}} = RB_{MAX} = 31.6K\Omega$$

Normalizing the value, it results in $R_B = 10K\Omega$. Checking the resulting $IB_{SAT}$:

$$\frac{V_{Q-OUT} - VBE_{ON}}{R_B} = IB_{SAT} = 436\mu A$$

According to maximun ratings of Texas Instruments datasheet, the $Q$ and $\overline{Q}$ outs can manage a current up to $25mA$, so the resulting $IB_{SAT}$ is in range.