Digital Electronics - Laboratory 1
Combinatorial and sequential logic

Academic year 2023-2024 / Teaching assistant: Arthur Fyon

Preface

These laboratory notes are an indispensable complement to the course and to the theory rehearsals. It is essential to extend the teaching by a concrete experimentation which illustrates the theory, justifies the simplifying models presented and dissipates the apprehensions in front of the "hardware" by showing how easy it is to assemble logic circuits, i.e. elements constituting a computer.

The picture painted here is not deliberately idyllic. There is obviously a world of difference between the realization of a combinatorial adder and that of a programmable card which concentrates the ingenuity of several man-years. However, both use the same basic components. The respect of a specification by the efficient marriage of software and hardware can only be obtained by a long experience which starts in the didactic laboratory, or better still, if one is a fanatic, in one's personal lab.

As for mathematics, and more modestly, there is no royal road for electronics! (Allusion to the answer of the brilliant mathematician Euler to the Tsar of Russia, whose guest he was, and who, used to the privileges that his title implied, asked for a shortcut to mathematical knowledge). The path to mastery begins with the elementary manipulations of the didactic laboratory.

Chapter 1

Introduction (to read before the lab)

The goal of this lab is to have you familiarized with the combinatorial and sequential logic by analyzing a homemade binary clock and by constructing yourself several elements of this clock. More precisely, you will play with buttons, 4-bit counters and 555-timers.

Before starting this first laboratory, this introduction will also be used to remind the basic knowledge you need to realize the laboratory comfortably.

1.1 Passive components

A resistor is a passive two-terminal electrical component that implements electrical resistance (in Ω) as a circuit element. The behavior of an ideal resistor is described by Ohm's law, indicating that the voltage across it is directly proportional to the current flowing through it:

$$V = R \cdot I$$

In electronic circuits, resistors are used to reduce current flow, adjust signal levels, to divide voltages, bias active elements, and terminate transmission lines, among other uses.

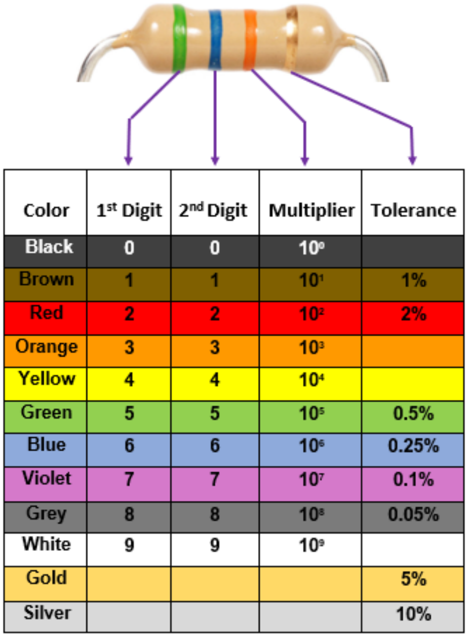
In the laboratory, you will use through hole axial resistors. The resistance of such resistors can be read by decoding the colored rings drawn on it, following the Table in Figure 1.1.

Thus, to know the value of the resistance of such resistors, orient the silver/gold ring to the right, and read the color from left to right to decode the value of the resistance. In the example of Figure 1.1, we can observe green, blue, orange and gold. Meaning that

$$R = 56 \cdot 10^3\,\Omega \pm 5\% = 56\,\mathrm{k}\Omega$$

In addition to that, these resistors have a maximum value of power that they can dissipate. Thus, one should always check that the power of dissipated by the resistor is always below this maximum value. Note that these maximum values are related to the size of the resistor: the larger the resistor, the higher the maximum value of power they can dissipate.

An often faster method to measure the resistance of a resistor is to measure it with a multimeter (note: do not measure the resistance value of a resistor in a powered circuit).



Color	1st Digit	2nd Digit	Multiplier	Tolerance
Black	0	0	10^0	
Brown	1	1	10^1	1%
Red	2	2	10^2	2%
Orange	3	3	10^3	
Yellow	4	4	10^4	
Green	5	5	10^5	0.5%
Blue	6	6	10^6	0.25%
Violet	7	7	10^7	0.1%
Grey	8	8	10^8	0.05%
White	9	9	10^9	
Gold				5%
Silver				10%

Figure 1.1: Through hole axial resistor decoder

A diode is a semiconductor device that essentially acts as a one-way switch for current. It allows current to flow easily in one direction, but severely restricts current from flowing in the opposite direction. There exists many types of diodes: junction diode, Zener diode, light emitting diode (LED), etc.

Example of a through hole diode is represented in Figure 1.2. The position of the cathode is represented by a silver ring on the diode.

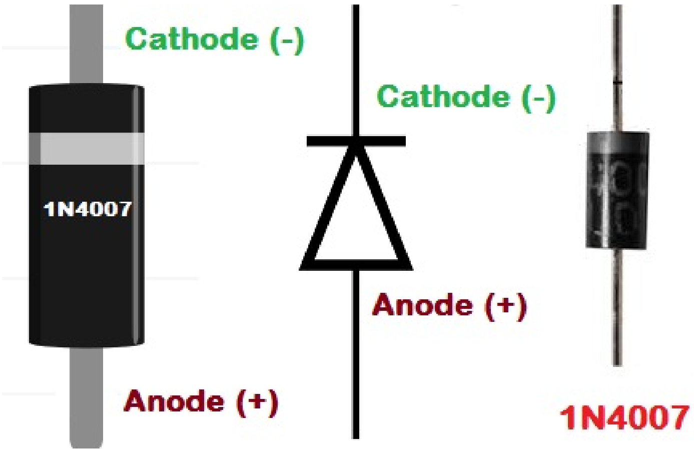


Figure 1.2: Through hole diode

The ones used in the laboratory are LEDs. Their operation is the following (in an ideal world):

- **direct** polarization (direction of the current = direction of the arrow): it lights up and the voltage at its terminals is 1.6V. The current through it must be limited to 20mA;
- **reverse** polarization: the LED remains off and behaves as an open circuit.

The LED is very useful when one wants to know with the naked eye if a signal is 1 or 0 in a digital circuit. To do so, we often place the LED in **series** with a 330 Ω resistor to limit the current in the LED. In practice, the direction of connection of an LED can be recognized by the length of the legs (and by the flat side of the LED): the smallest leg should be placed at the side of lower voltage (often ground), as indicated in Figure 1.3.

A specific electronic component using LEDs is the 7-segment display, used to display numbers. In practice, 7-segment displays have 10 pins: 2 for powering it, 1 driving each segment and 1 driving the decimal point, as represented in Figure 1.4.

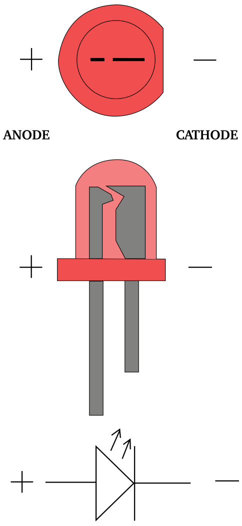


Figure 1.3: LED scheme

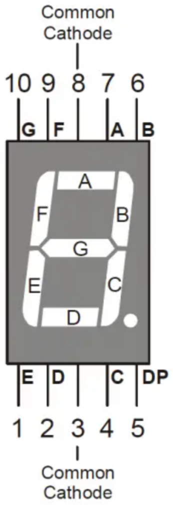


Figure 1.4: 7-segment display

Note that decoding a BCD (Binary Coded Decimal) number (4 bits) to 7-segment display might be fastidious. Hopefully, some ICs decoder exists, such as the CD4543BE from Texas Instruments used in this laboratory, represented in Figure 1.5.

A capacitor is a two-terminals passive electronic component that stores electrical energy in an electric field. The capacity of a capacitor to store electrical energy is called the capacitance, expressed in Farad (F). Capacitance values range from a few picofarads (10^{-12} F) to a few thousand microfarads (10^{-6} F). There are two main families of capacitors:

- non-polarized capacitors ($10^{-12}\,\mathrm{F} < C < 10^{-6}\,\mathrm{F}$);
- electrolytic capacitors (blue cylinders) whose capacitance can reach several thousand microfarads.

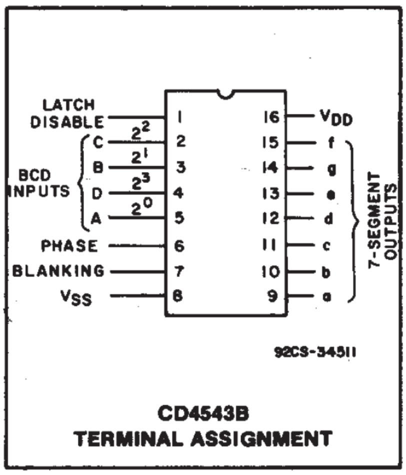


Figure 1.5: BCD to 7-segment display decoder CD4543BE

The capacitors used in the laboratory are generally not polarized. The direction of the connection is therefore not important. On the other hand, for electrolytic capacitors, the signs are clearly indicated near the legs. A connection error often causes an explosion of the component with the consequences that this implies: burns, damage to other components, etc.

Another important characteristic of a capacitor is its maximum allowable voltage. This is usually written on the cylinder.

Again, a very fast method to measure the capacitance of a capacitor is to measure it with a multimeter (note: do not measure the capacitance value of a capacitor in a powered circuit).

A button is a simple switch mechanism characterizing a short circuit or an open circuit between two electrical points, depending on the state of the button. The main inconvenient of practical buttons are the bounces. Indeed, when pressing or releasing the button, the transition between the short circuit and the open circuit is not sharp and perfect, there are high frequency voltage oscillations, as represented in Figure 1.6. One should always implement a digital or analog filter if these bounces are problematic.

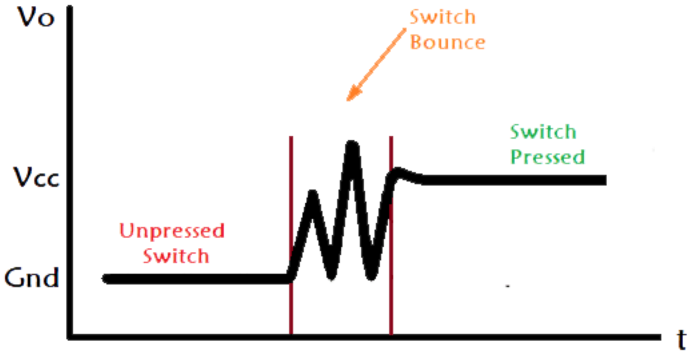
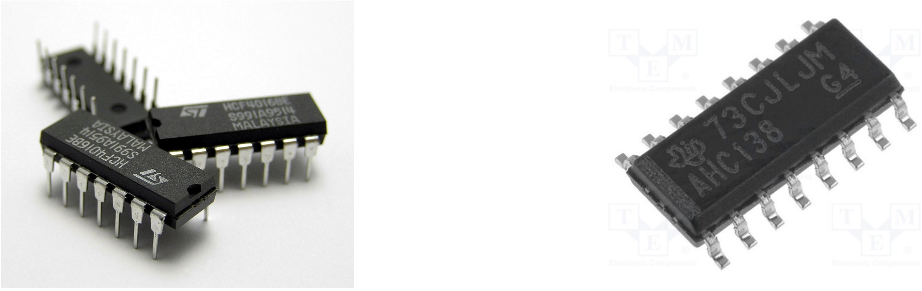


Figure 1.6: Button bounces

1.2 The integrated circuit (IC) family

1.2.1 Introduction

Depending on their size, integrated circuits can be classified as **SSI** (Small Scale Integration), **MSI** (Medium Scale Integration), **LSI** (Large Scale Integration) or **VLSI** (Very LSC). SSIs contain less than 12 logic gates (or equivalent gates); MSIs can contain up to 100 equivalent gates; and LSIs and VLSIs contain more than 10,000 gates, which allows for more complex logic functions. In practice, ICs consists in a black case with several inputs/outputs pins. The package can be DIP (Dual Inline Package) with two rows of straight pins. This kind of case is mostly used in breadboards (see later). Another large family of package is SMD (Surface Mount Device), with 2 or 4 ranges of bent pins. This kind of case is mostly used in PCB (Printed Circuit Board) (see later). An example of the two packages is represented in Figure 1.7.



(a) DIP package

(b) SMD package

Figure 1.7: Package examples

The pin numbering of all ICs follows the same convention: pointing the pins down and the notch up (or the circle), the top left pin is numbered 1. Then go down around the perimeter of the case and you find the pins of increasing number. This principle is represented in Figure 1.8.

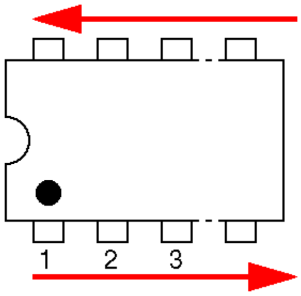


Figure 1.8: IC pin numbering convention

One IC is differentiated from another by an identification number. For example, the **74LS00** circuit is a quadruple NAND gate with 2 inputs:

- "74": most used standard used currently;
- "LS": manufacturing technology (Low power Schottky);
- "00": circuit type (00 = quad 2-inputs NAND).

1.2.2 Powering

Each IC must be powered, since they are active components. To do this, connect the VCC pin to a stable 5 V voltage (or other, depending on the IC technology) and the GND pin to the corresponding ground. In general, the ground is located at the bottom, on the left (pin 7 or 8) and the positive voltage, opposite the chip (pin 14 or 16). This rule is not always respected, so it is wise to check before connecting. A **reversal** of the power supply terminals is often **fatal**.

If you have a stabilized 5 V power supply (like the one in the laboratory), use it. Otherwise, one should use a power stabilizer circuit such as the **7805** circuit, producing a stable 5 V voltage. In addition to this, decoupling capacitors are often used to avoid that current spikes propagates through the circuit.

1.2.3 Inputs

A logic input is a voltage. The value '0' is implemented by a connection to ground. A '1' normally corresponds to a connection to VCC. However, it is advisable in "serious" circuits to introduce a 10 kΩ resistor in series to protect the input against transient overvoltages. This resistor is sufficient for 50 inputs. You should never leave floating inputs, since a floating input is an indeterminate input. If you want to enter a '1' and a '0' alternately, you can use a switch mounted as represented in Figure 1.9.

If the switch is in the open position (as shown in the figure), the circuit "sees" a logic '1' as input. If it is in the closed position (connected to ground), the circuit "sees" a logical '0'. The resistor, called pull-up resistor, usually has a value between 5 and 1 kΩ.

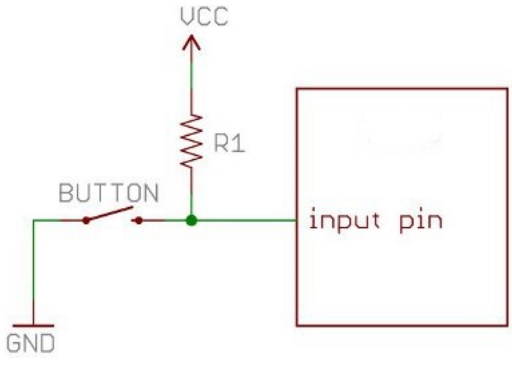


Figure 1.9: Pull-up resistor: correct input pin design

1.2.4 Outputs

A first way to check a logic value is to use the probes of a multimeter. The black probe can remain permanently connected to a ground, the red probe will be put in contact with the wire carrying the logic level to be read.

To check several logic values simultaneously or to measure transient values, this method becomes cumbersome and it becomes simpler to use a more visual method of measurement, such as LEDs.

As reminded, an LED only conducts and illuminates if it is biased in the forward direction. The voltage at its terminals is then about 1.6V. An LED cannot be connected directly to a voltage of $5\,\mathrm{V}$. It will therefore always be placed in series with a resistor of $\pm 330\,\Omega$. This resistor will see at its terminals a voltage of $5\,\mathrm{V} - 1.6\,\mathrm{V} = 3.4\,\mathrm{V}V$, it will therefore be traversed by a current of: $\frac{3.4\,\mathrm{V}}{330\,\Omega} \approx 10\,\mathrm{mA}$, like the diode. This current is enough to light the LED, but not too high to avoid damaging it. If the diode is reverse biased, it behaves like an open circuit up to a certain value of the reverse voltage (diode breakdown).

If you prefer to see the diode on for the other logic value, insert an inverter at the output of the circuit.

1.3 Terminology

Despite the large number of integrated circuit manufacturers, the terminology used is almost standardized. In order to understand a little better the description of the circuits which is given in the datasheets (easily found on Internet), here is the explanation of the most used terms.

- $\boldsymbol{V_{IH}}$ (Voltage Input High): the voltage level necessary to have a logical 1 at the input. Any voltage below this level is not considered as a HIGH state by the logic circuit.
- $\boldsymbol{V_{IL}}$ (Voltage Input Low): the voltage level necessary to have a logical 0 at the input. Any voltage above this level is not considered as LOW by the logic circuit.
- $\boldsymbol{V_{OH}}$ (Voltage Output High): the minimum voltage level of the output of a logic circuit corresponding to logic state 1.
- $\boldsymbol{V_{OL}}$ (Voltage Output Low): the maximum voltage level at the output of a logic circuit corresponding to logic state 0.

Parasitic electric and magnetic fields can induce voltages in the connecting wires of logic circuits. These signals are called **noise** and can sometimes bring the voltage below the V_{IH} value or bring it above V_{IL}. The noise immunity of a logic circuit defines the ability of the circuit to tolerate spurious voltages on its inputs. The quantitative measure of noise immunity is called the **noise sensitivity margin**. It is illustrated in Figure 1.10 by the symbols $\boldsymbol{V_{NH}}$ and $\boldsymbol{V_{NL}}$.

All voltages in the indeterminate band should never appear on the terminals of a logic circuit because they cause an unpredictable response. According to Figure 1.10, the **noise sensitivity margin to high state** V_{NH} noise is defined by:

$$V_{NH} = V_{OH} - V_{IH}$$

and the **noise sensitivity margin to low state** V_{NL} noise is defined by:

$$V_{NL} = V_{OL} - V_{IL}$$

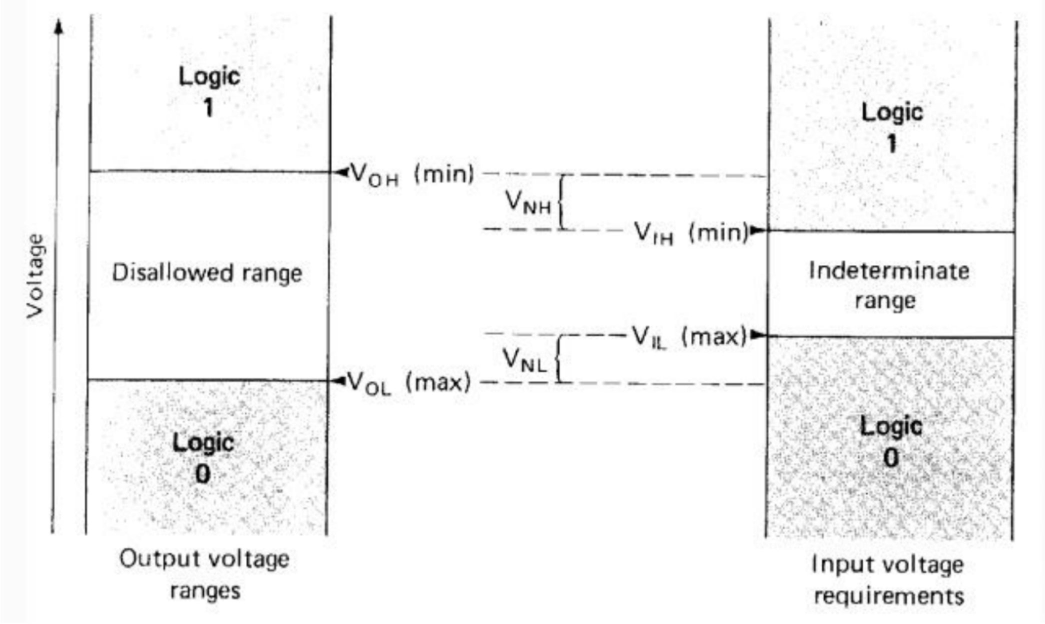


Figure 1.10: Noise sensitivity margin

Propagation delays: a signal that passes through a circuit always experiences a delay. Two propagation delays are defined:

- t_{PLH}: delay to go from logic level 0 to logic level 1;
- t_{PHL}: delay to go from logic level 1 to logic level 0;

These delays are illustrated in Figure 1.11 for an inverter.

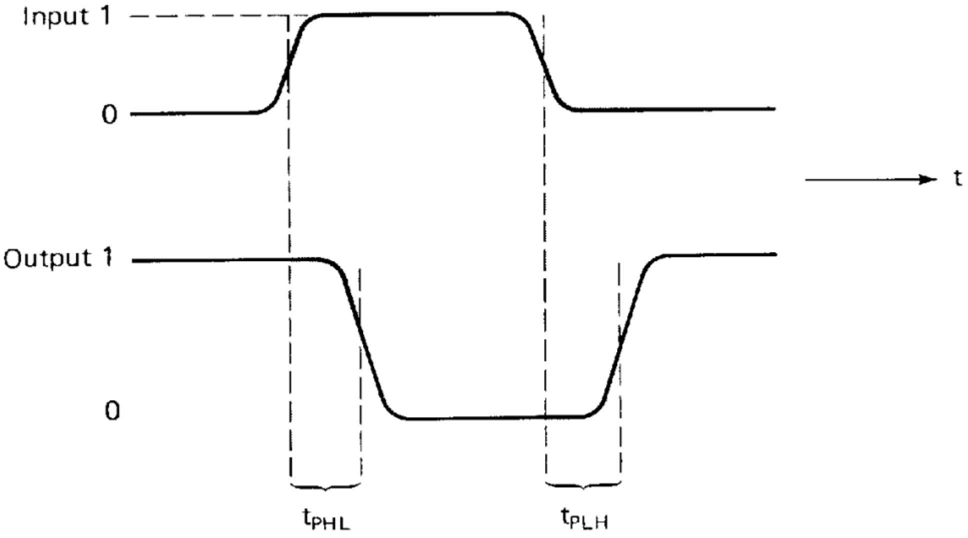


Figure 1.11: Propagation delays

The fanout is the maximum number of logic inputs that the output of a single device can drive without problems. If you exceed the number indicated, you cannot be sure that the voltages of the logic levels of the outputs will be correct. It is therefore very important to know the fanout of all the circuits used in an assembly. The gates with the highest fanout are the buffers.

Note: if you want to learn more on the different specific IC families, see the related appendix.

1.4 Specific useful ICs for the laboratory

1.4.1 555 timer

A timer is designed to deliver in output signals whose period is of a very precise duration. The 555 timer is an extremely versatile IC that can be used to build many different circuits. In this laboratory, it is used as a clock of precisely 1 Hz to build our binary clock. The one you will use in this laboratory is the NE555P from Texas Instruments, represented in Figure 1.12.

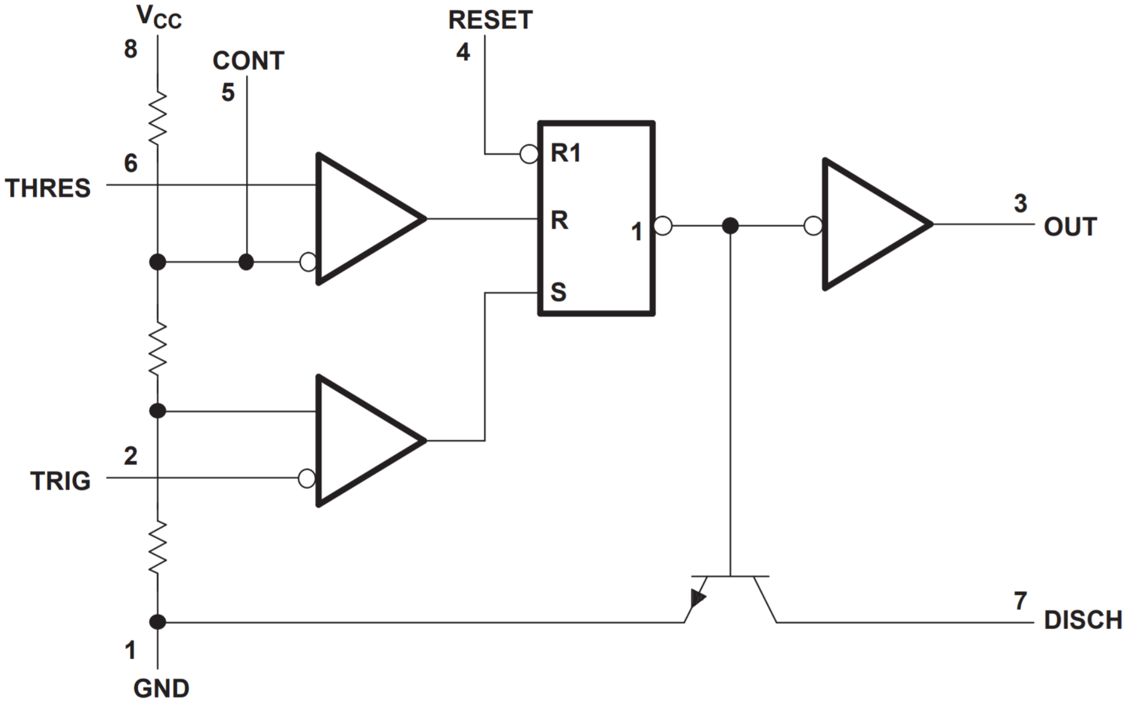


Figure 1.12: NE555P from Texas Instruments

This specific circuit can act in 2 different modes: astable and monostable.

For the astable mode, a simplified electronic schematic is represented in Figure 1.13.

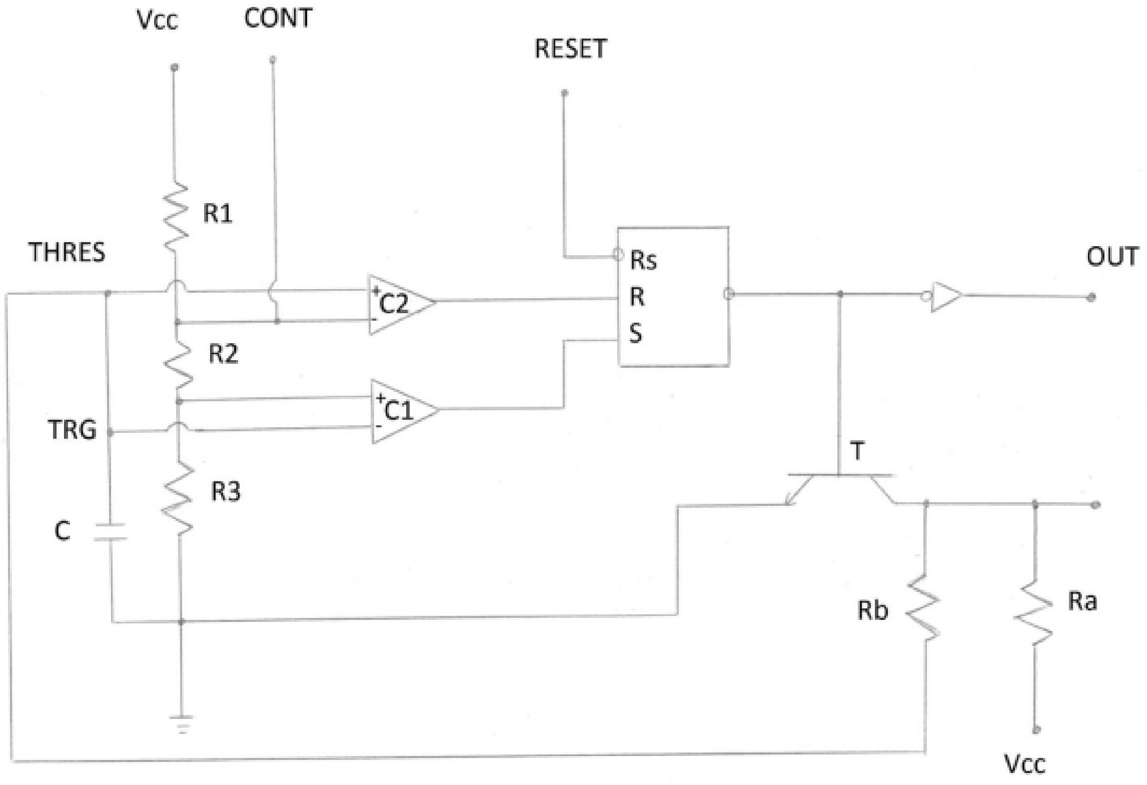


Figure 1.13: Astable mode 555 timer schematic

In which the IC has the truth table depicted in Table 1.1.

R	S	Output Q
0	0	Unchanged
0	1	0
1	0	1
1	1	NA

Table 1.1: Truth table

The basic principle of this timer is that it maintains the voltage across a capacitor between two limit values. Inside the 555 timer, three identical resistors R_1, R_2 and R_3 (R in Figure 1.12) divide the supply voltage into 3 voltages. For example, if we supply the circuit with a generator that delivers 9 volts, each of the resistors will cause a voltage drop of 3 volts.

The aim is therefore, by dividing the supply voltage into 3 parts, to provide the comparators C_1 and C_2 with two reference voltages, which will be equal to 1/3 and 2/3 of the supply voltage. Each of the 2 comparators has a leg connected to a reference point, either 1/3 or 2/3 of the supply.

The other two legs of the two comparators are linked together and connected to the terminal of the external capacitor C. The voltage on the capacitor is variable while the reference voltages are fixed. The comparator C_1 "only acts" if the voltage on the capacitor is less than 1/3 of the supply voltage. This is the case at the very beginning of the experiment since the capacitor is initially discharged. The output of comparator C_1 is therefore the logical value 1. Comparator C_2 "only acts" if the capacitor voltage is greater than 2/3 of the supply voltage, which is not the case, the output of comparator C_2 is the logical value 0.

From then on, a set is made (see truth table). A low voltage controls the transistor T which then acts as an open circuit. The external capacitor C can then be charged through the external resistors R_a and R_b . The charging continues until the voltage across the capacitor reaches 2/3 of the supply voltage. At this precise moment, we observe an inverse configuration: the output of comparator C_2 is the output of comparator C_1 is at 0.

As a result, a reset operation is performed and a high voltage controls the transistor. The transistor behaves like a closed circuit and the external capacitor C is then short-circuited and discharges through the external resistor R_b . As long as the 555 timer is powered, the charge/discharge cycle of the capacitor repeats itself endlessly. The circuit is said to be astable, it has no stable state, it oscillates continuously. In astable mode, the purpose of the circuit is to produce a continuous series of pulses at a controlled frequency.

The duration of the signal at a high logic level and then at a low logic level is determined by the values of the external resistors R_a and R_b and the capacitor. The output will be at the high logic level for a time T_h given by:

$$T_h = 0.695 \cdot (R_a + R_b) \cdot C$$

The output will be at the low logic level for a time T_l given by:

$$T_l = 0.695 \cdot R_b \cdot C$$

The total period T (duration of a cycle high logic level + low logic level) is given by:

$$T = T_h + T_l = 0.695 \cdot (R_a + 2R_b) \cdot C$$

This mode is the one in which the 555 timer of the laboratory is operating, to produce the 1 Hz required signal for the binary clock.

For the monostable mode, a simplified electronic schematic is represented in Figure 1.14.

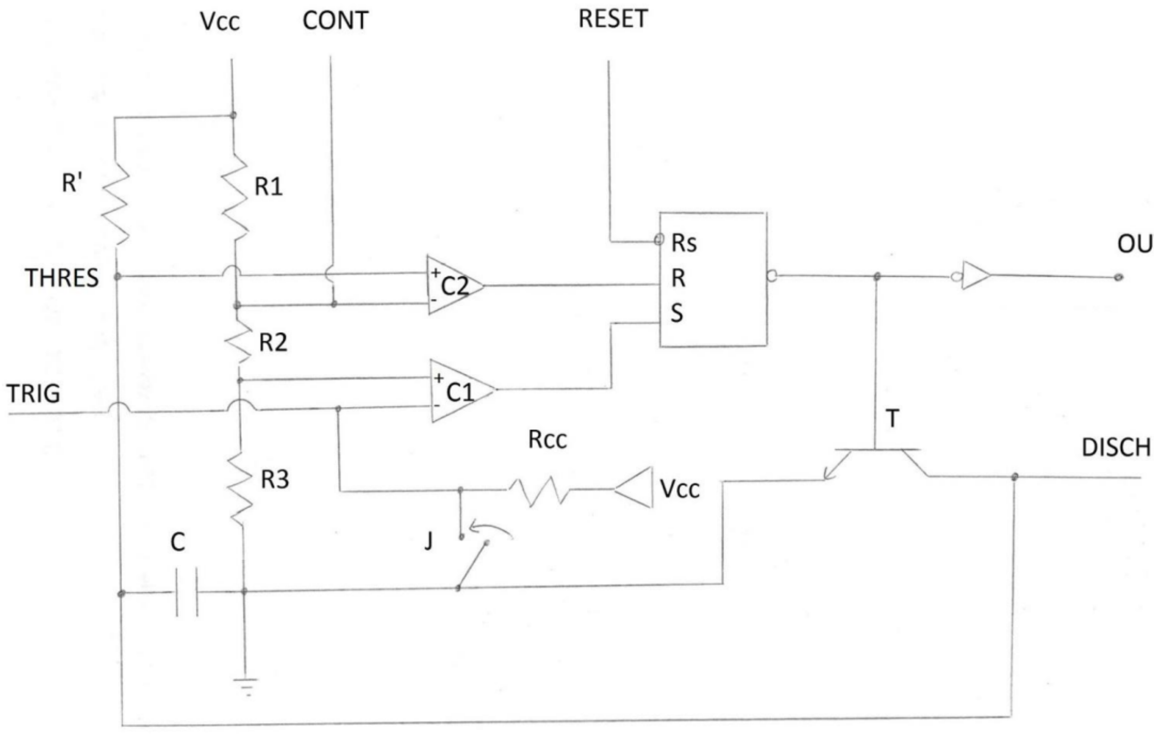


Figure 1.14: Monostable mode 555 timer schematic

Unlike the 555 timer in astable mode, which has two unstable states and continuously switches from one to the other, the 555 timer in monostable mode has one stable and one unstable state. The fundamental differences inside the circuit compared to the previous configuration are that there are not two external resistors but only one, TRIG is not connected to the capacitor, and there is a button in the circuit.

The use of the 555 timer in monostable mode allows to generate a pulse of a defined duration only with a resistor and a capacitor.

If we start from the same initial configuration as during the study of the astable mode, the capacitor is initially discharged. The circuit will remain in this stable state as long as no impulse is sent to it. When the button is pressed, the same phenomenon occurs as described for the astable mode, a charge/discharge cycle of the capacitor takes place. After the discharge, since TRIG is no longer connected to the capacitor, the output of the comparator C_1 will be equal to 0. A set will therefore not be implemented and a cycle will not start again. The circuit remains in its stable configuration, i.e. when the capacitor is discharged.

Until the button is pressed again, the circuit will not change state. In monostable mode, the purpose of the circuit is to produce a pulse of defined length following a pulse of any length. The pulse width communicated by the button is given by:

$$T_w = 1.1 \cdot R \cdot C$$

1.5 Bidirectional 4-bit counter

Bidirectional 4-bit counters, also known as Up/Down 4-bit counters, are capable of counting in either direction through any given count sequence (between 0 and 15) and they can be reversed at any point within their count sequence by using an additional up/down control input. They consist in 4 synchronously clocked D flip-flops connected in counters. In this laboratory, the bidirectional 4-bit counters are the CD4516BEE4 from Texas Instruments, depicted in Figure 1.15.

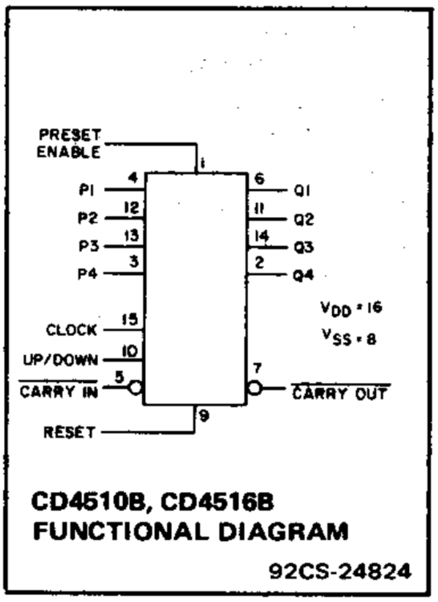


Figure 1.15: CD4516BEE4 from Texas Instruments

The pins of interest in the CD4516BEE4 are Q1-4, representing the 4 output bits of the counter (that is, the current value of the counter), CLOCK, for which a rising edge makes the counter increments and RESET, resetting the counter for a high logic value.

To explain the functioning of a synchronous 4-bit up counter, let's explain the functioning of an asynchronous 4-bit up counter. We recall the difference between asynchronous and synchronous counters: in the first case, the clock signal is applied to the input of the first stage of the counter, which leads to the successive modification of the state of the following stages. In the second case, the clock signal is applied to all the counter stages at the same time. The final state of the counter is the one fixed just after the transition of the clock signal, and derives from the one immediately preceding this transition. That is, the functioning of a synchronous counter can be understood from the functioning of an asynchronous counter. An electronic schematic for a asynchronous 4-bit counter using D flip-flops is represented in Figure 1.16.

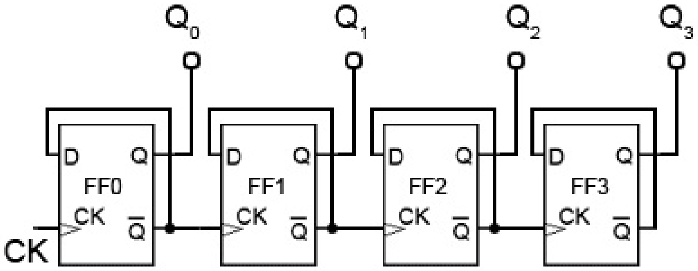


Figure 1.16: Asynchronous 4-bit up counter using D flip-flops

In this example, the counter capable of counting numbers from 0 to 15. The clock inputs of all flip flops are cascaded and the D input (DATA input) of each flip flop is connected to a state output of the flip flop.

Let us assume that the 4 Q outputs of the flip flops are initially 0000 ($Q3\,Q2\,Q1\,Q0$). When the rising edge of the clock pulse is applied to the $FF0$, then the output $Q0$ will change to logic 1 and the next clock pulse will change the $Q0$ output to logic 0. This means the output state of the clock pulse toggles (changes from 0 to 1) for one cycle.

As the $\overline{Q}$ of $FF0$ is connected to the clock input of $FF1$, then the clock input of second flip flop will become 1. This makes the output of $FF1$ to be high (i.e. $Q1 = 1$), which indicates the value 0010. In this way the next clock pulse will make the $Q0$ to become high again. So now both $Q0$ and $Q1$ are high, this results in making the 4 bit output 0011. Now if we apply the fourth clock pulse, it will make the $Q0$ and $Q1$ to low state and toggles the $FF2$. So the output $Q2$ will become 0100. As this circuit is 4 bit up counter, the output is sequence of binary values from 0, 1, 2, 3, ..., 15 i.e. 0000 to 1111. The timing diagram of such counter is depicted in Figure 1.17.

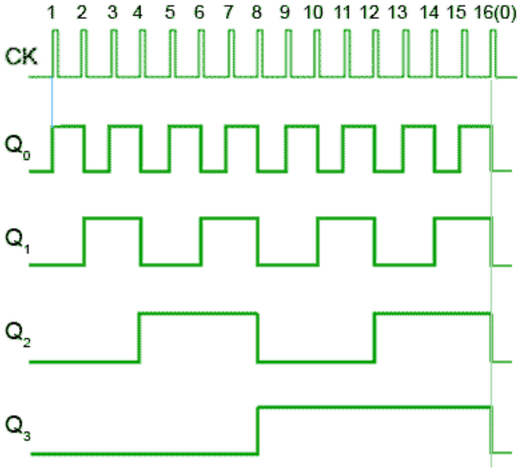


Figure 1.17: Timing diagram of an asynchronous 4-bit up counter using D flip-flops

Note that if you want to realize the same down counter is to connect the clock of cascaded D flip-flops to Q and not $\overline{Q}$.

1.5.1 Other ICs

In order to handle properly the 3 button signals (reset, start and stop) that are present on the electronic board, the SN74AC74N from Texas Instruments is used. The SN74AC74N devices contain two independent D-type positive-edge-triggered flip-flops with asynchronous preset and clear pins for each.

The different logic gates are contained in different ICs on the elctronic board.

1.6 Hardware to build your circuit

Now that you know everything concerning the components that will be used in our binary clock, one needs to know how to interconnect them. More specifically, breadboards and PCB (Printed Circuit Board) will be briefly described.

1.6.1 Breadboard

A breadboard, or protoboard, is a construction base for prototyping of electronics. Because it does not require soldering, it is reusable. This makes it easy to use for creating temporary prototypes and experimenting with circuit design. For this reason, solderless breadboards are also popular with students and in technological education. It consists of a board with many "holes" for which there is an electrical contact (a short circuit) for each horizontal line, except for the two or four long power columns on the sides of the breadboard. A breadboard and its architecture is depicted in Figure 1.18.

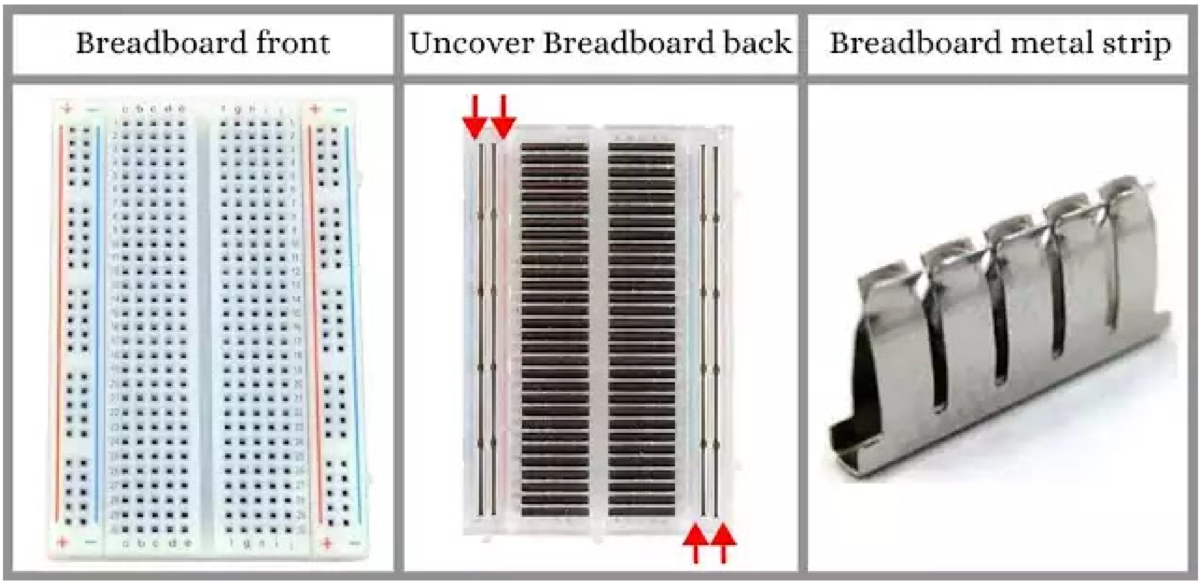


Figure 1.18: Breadboard and its architecture

1.6.2 PCB

A PCB is a laminated sandwich structure of conductive and insulating layers. PCBs have two complementary functions. The first is to affix (SMD) electronic components in designated locations on the outer layers by means of soldering. The second is to provide reliable electrical connections (and also reliable open circuits) between the component's terminals in a controlled manner often referred to as PCB design. Each of the conductive layers is designed with an artwork pattern of conductors (similar to wires on a flat surface) that provides electrical connections on that conductive layer. Another manufacturing process adds vias, plated-through holes that allow interconnections between layers. An example of a DVD player PCB is represented in Figure 1.19.

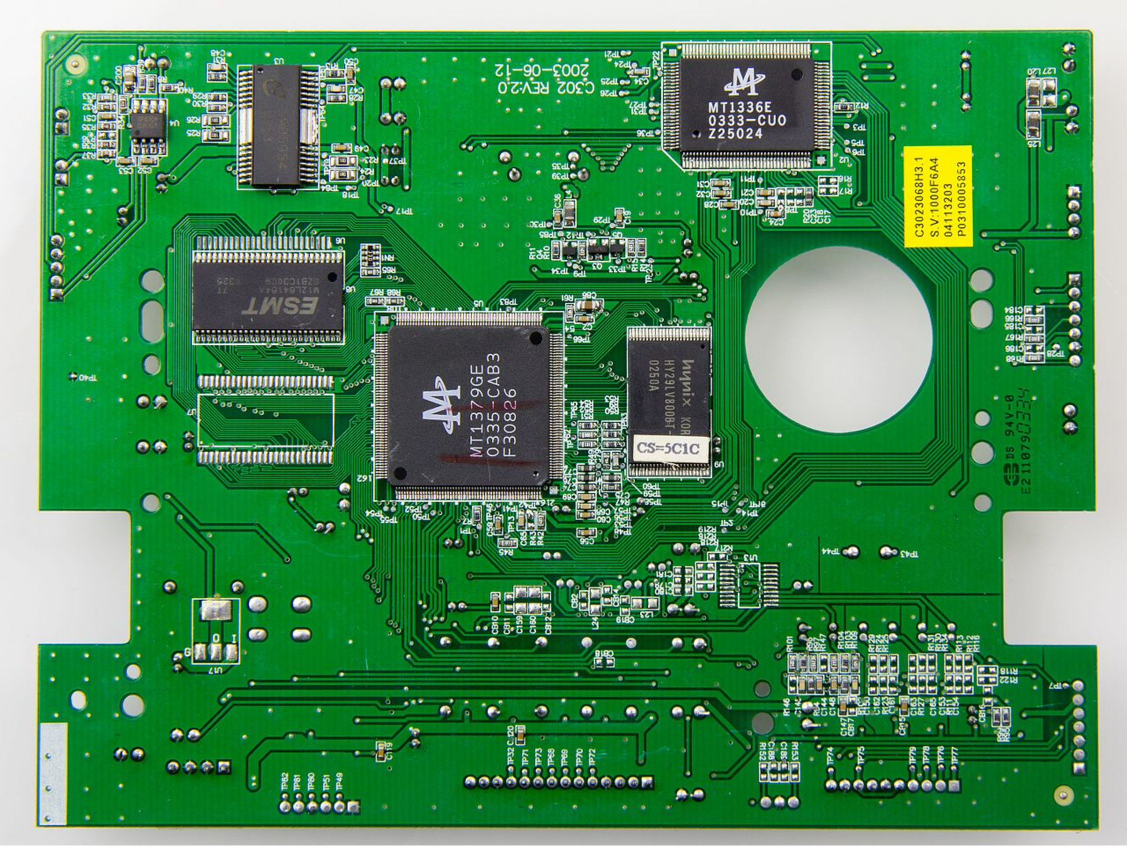


Figure 1.19: PCB of a DVD player.

1.7 ⚠ Software to download before the lab ⚠

Before the lab, you must download the *Logisim* software. *Logisim* is an educational tool for designing and simulating digital logic circuits that will help you understanding the functioning of the binary clock.

Please, download the 2.7.1 release of *Logisim* suited for your OS here. For instance, the *Windows* file is an executable one, so no installation will be needed. Moreover, download the 3 *.circ* files that can be found in myUliege and place them in the same folder as the *Logisim* executable. You might need to install a Java environment. Don't worry, you will be automatically redirected on the Java's website.

To give you an overview of what is *Logisim*, Figure 1.20 represents its GUI (Graphical User Interface).

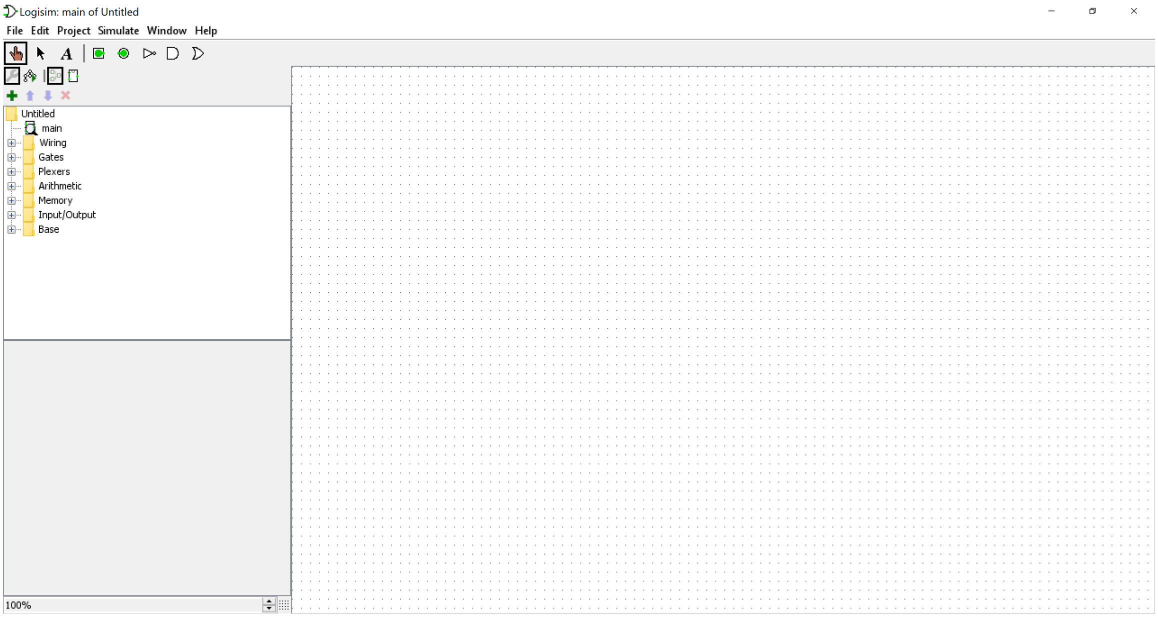


Figure 1.20: GUI of *Logisim* on Windows.

- On the top, you can several menus. The ones of interest are **file** and **simulate** in which you can handle your files and launch real time simulations respectively.
- Right below, you have several buttons. Concerning the most important ones: ☝ makes your mouse being a poke tool useful to interact with electronic elements during the simulation (such as a reset button), ↖ makes your mouse an edit and select tool useful when building your circuit and ▷○ D ▷ are quick access for placing different logic gates in your circuit.
- In the tree below, you have access to all elements that are implement in *Logisim* such as any logic gates, multiplexer, counter, clocks and so on.
- In the window in the bottom left, you have access to the parameters of the selected object. For instance, by cicking on an AND gate, you can tune the number of inputs, its orientation, etc.
- Finally, the large window on the right is your work space in which you will build your circuit.

Chapter 2

Laboratory manipulations

2.1 Materials

The manipulations of this laboratory are carried out on breadboard and on an electronic board, the latter including

- a 555 timer (NE555P) in astable mode generating the 1 Hz input signal of the binary clock;
- 4 4-bit synchronous up counters (CD4516BEE4), used to count the seconds, tens of seconds, minutes and tens of minutes;
- many LEDs used to display the time in binary numbers;
- 4 7-segment displays (HDSP-H153) with their specific BCD to 7-segment decoders (CD4543BE) used to display time in decimal numbers;
- 3 buttons (START, STOP and RESET) used to start the clock, stop it or reset it respectively, handled by D flip-flops (SN74AC74N);
- some logic gates (CD74HC08E, CD4069UBE and SN7432NE4) used for handling the different deferrals of the counters (more on this later);
- some power elements used to produce a stable 5 V voltage from an arbitrary input voltage comprised between 7 and 35 V: a power MOSFET (IRF9640) ensuring reverse polarity protection and an LDO (L7805CD2T-TR) responsible for producing a stable 5 V using any input voltage between 7 and 35 V.

From the user point of view, the only input of the board is the 7-35 V input voltage used to power the board. This input corresponds to the 2 big red and black banana connectors situated on the top left when facing the board. **NEVER** use a voltage higher than 35 V, this may cause irreversible damage to the board. Concerning the outputs, many test points are present on the board. These are mainly used to observe some useful voltages using the oscilloscope. For instance, test points can be used to observe the TRIG/THRESH signal of the 555 timer, as well as the discharge and output signals. You can let the board powered while switching from one test point to another, but you **MUST** take care of any short circuit between the test points.

When playing with circuits on breadboard, **ALWAYS** power up your circuit when the circuit is finished. **ALWAYS** use the DC power supply at 5 V when powering up circuit on breadboard! If you have any doubt, **ASK!**

In the following, the next page represents the full schematic of the electronic board (which can also be found in myUliege, *Final_ sheet.pdf*), and Figure 2.1 represents a 3D model of the full board. **Take some time to look at the connections made and to deduce the roles of each component (integrated circuits, buttons, etc.) of the board in order to familiarize yourself with it.**

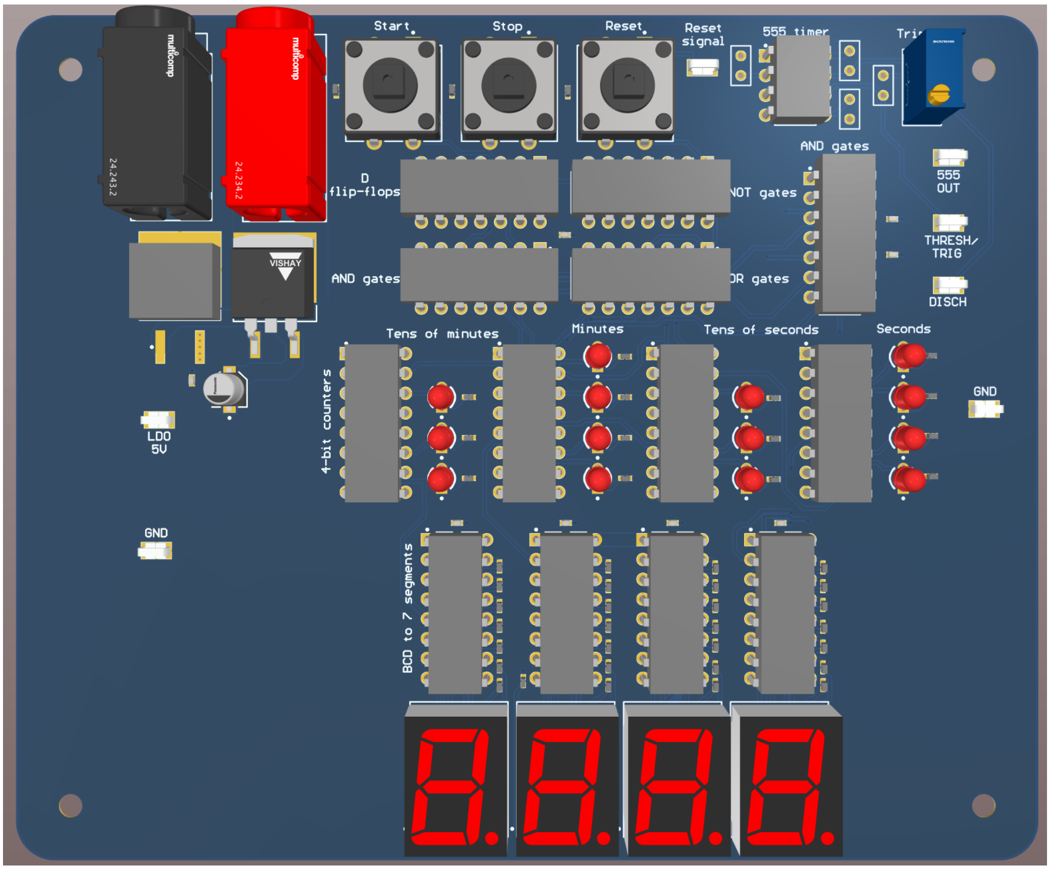
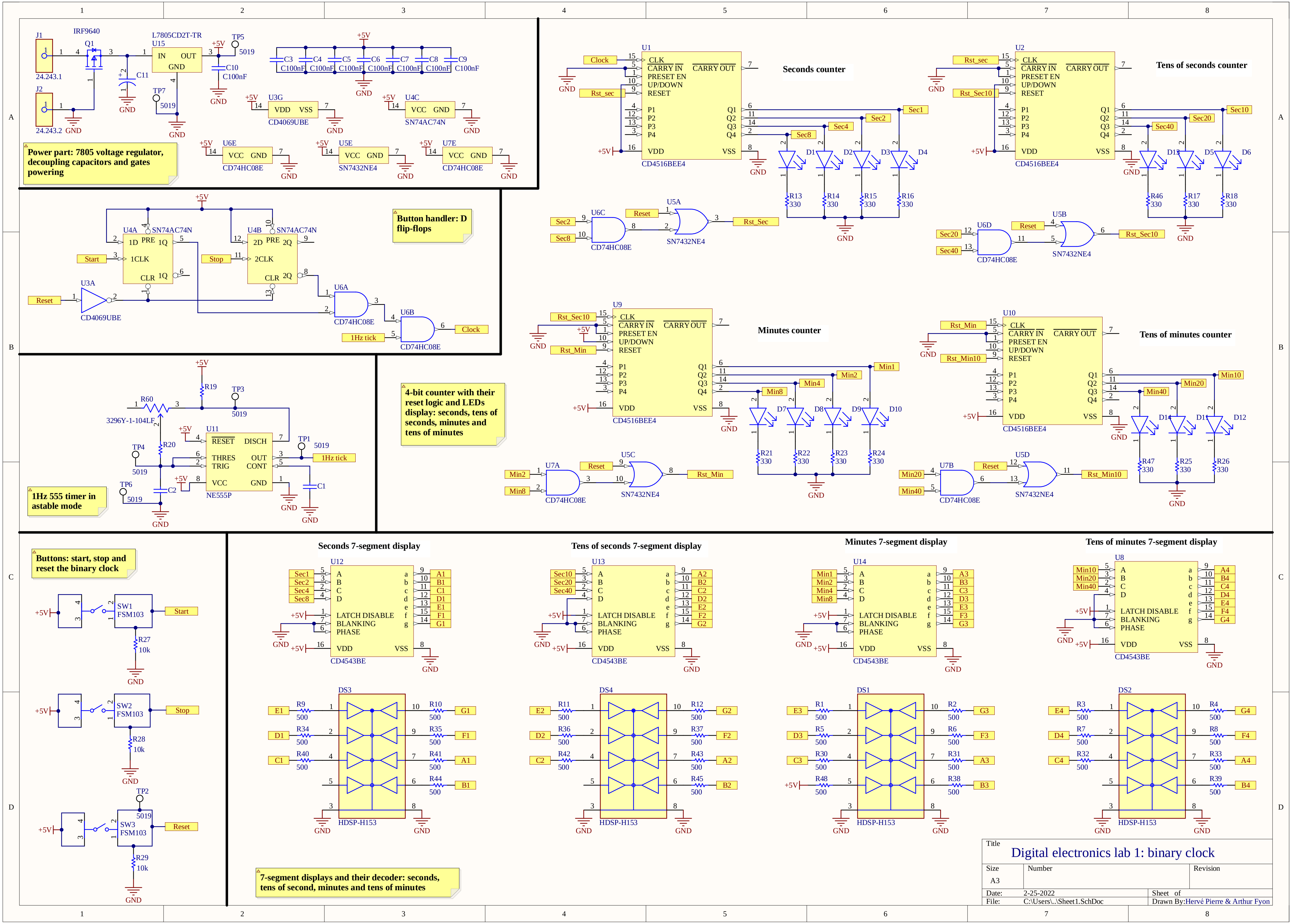


Figure 2.1: 3D model of the board.


Seconds counter
Tens of seconds counter
Minutes counter
Tens of minutes counter
Seconds 7-segment display
Tens of seconds 7-segment display
Minutes 7-segment display
Tens of minutes 7-segment display
J1 24.243.1
J2 24.243.2
Q1 IRF9640
U15 L7805CD2T-TR
IN OUT GND
C3 C4 C5 C6 C7 C8 C9 C10 C100nF
C11
TP5 TP7 5019
U3G CD4069UBE VDD VSS
U4C SN74AC74N VCC GND
U6E CD74HC08E
U5E SN7432NE4
U7E CD74HC08E
U1 U2 U9 U10 CD4516BEE4
CLK CARRY IN CARRY OUT PRESET EN UP/DOWN RESET P1 P2 P3 P4 Q1 Q2 Q3 Q4 VDD VSS
Clock Rst_sec Rst_Sec10 Rst_Sec Rst_Min Rst_Min10 Reset
Sec1 Sec2 Sec4 Sec8 Sec10 Sec20 Sec40
Min1 Min2 Min4 Min8 Min10 Min20 Min40
D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14
R13 R14 R15 R16 R17 R18 R21 R22 R23 R24 R25 R26 R46 R47 330
U5A U5B U5C U5D SN7432NE4
U6A U6B U6C U6D U7A U7B CD74HC08E
U4A U4B SN74AC74N
1D PRE 1Q 1CLK CLR 1Q
2D PRE 2Q 2CLK CLR 2Q
U3A CD4069UBE
Start Stop 1Hz tick
U11 NE555P
RESET DISCH THRES TRIG OUT CONT VCC GND
R60 3296Y-1-104LF
R19 R20 C1 C2
TP1 TP3 TP4 TP6 5019
SW1 SW2 SW3 FSM103
R27 R28 R29 10k
TP2 5019
U8 U12 U13 U14 CD4543BE
A B C D LATCH DISABLE BLANKING PHASE a b c d e f g VDD VSS
A1 B1 C1 D1 E1 F1 G1 A2 B2 C2 D2 E2 F2 G2 A3 B3 C3 D3 E3 F3 G3 A4 B4 C4 D4 E4 F4 G4
DS1 DS2 DS3 DS4 HDSP-H153
R1 R2 R3 R4 R5 R6 R7 R8 R9 R10 R11 R12 R30 R31 R32 R33 R34 R35 R36 R37 R38 R39 R40 R41 R42 R43 R44 R45 R48 500
+5V
GND

Power part: 7805 voltage regulator, decoupling capacitors and gates powering

Button handler: D flip-flops

4-bit counter with their reset logic and LEDs display: seconds, tens of seconds, minutes and tens of minutes

1Hz 555 timer in astable mode

Buttons: start, stop and reset the binary clock

7-segment displays and their decoder: seconds, tens of second, minutes and tens of minutes

Title: Digital electronics lab 1: binary clock

Size: A3 | Number: | Revision:

Date: 2-25-2022 | Sheet of

File: C:\Users\..\Sheet1.SchDoc | Drawn By: Hervé Pierre & Arthur Fyon

2.2 First manipulation: playing with the board

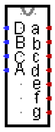
Now it's time to power the board. Configure the DC power supply of the lab to produce 9 V and connect it to the board using 2 banana cables.

1. Vary (slightly) the voltage of the DC power supply and monitor the test point voltage *LDO 5V* using the oscilloscope, what do you observe?
2. Play with the buttons, what are their roles?
3. What are the LEDs used for?
4. What are the 7-segment displays used for?
5. How high can each digit go? How many bits would be needed to count the same number of seconds with a single counter?

2.3 Second manipulation: understanding the clock

For now, let's suppose that we have access to a nice clock of frequency 1 Hz. We will understand how the 4-bit counters are connected between each other in order to produce a sexagesimal clock using first *Logisim*.

1. Launch *Logisim* and open the file *simple_ counter.circ*.
2. Observe the circuit. The building blocks of this circuit are the following:
   - is the clock block producing a periodic signal[1];
   -

 is the counter block counting up on a specified number of bits (here 4) on the *Q* pin at each rising edge of the *cl* pin, it can also be reset using a high signal on the *0* pin;
   - is a splitter used to split the different bits of a signal when these ones are grouped in one bold black wire (as with the output of the counter);
   - is an LED;
   - 

 is a custom BCD to 7-segment displays decoder that have been implemented in the file *7-segment-display-driver.circ*.
3. Start the simulation. To do so, check **Simulation Enabled** and **Ticks Enabled** below the **Simulate** menu. Moreover, tune the Tick Frequency to 2 Hz, so that the clock frequency is 1 Hz.

[1] Note that the period of the clock is specified in number of ticks. The tick frequency can be tuned under the **simulate** menu.

4. Observe what is happening. To which value the counter counts before dropping again to 0?

5. Find a way to build a reset signal (pin *0* of the counter) so that the counter counts from 0 to 9 only. To do so, use logic gates and bits of Q.

6. Now that your counter counts from 0 second to 9 seconds, implement a manual reset button (button are found below the *Input/Output* subtree).

7. To go further, we want to count from 0 second to 59 seconds. To do so, let's imagine we would use another counter. What signal of the circuit you have built could be used as the clock of the "tens of seconds" counter? There is no need of creating such circuit.

8. Once you fully understood the second part, open the *binary_ clock.circ* file. This file contains a logic circuit implementing a binary clock that can count up to 9m59s. Observe the circuit and try to understand what is happening in addition to *simple_ counter.circ*.

9. Finally, try to find similarities between the circuit contained in *binary_ clock.circ* and the electronic board you are facing.

2.4 Third manipulation: the 555 timer

In the second manipulation, we have supposed that we can have access to a nice clock of frequency 1 Hz. We will understand how to build such clock with a 555 timer in astable mode.

1. By checking the electronic scheme of the board and ignoring for now the variable resistor (called a potentiometer or a trimmer), compute the period of the 555 timer given the formula above. In the board, $C2 = 2.2\,\mu\text{F}$, $R20 = 100\,\text{k}\Omega$ and $R19 = 330\,\text{k}\Omega$.

2. Why didn't we use appropriate values of capacitor and resistor to achieve the exact desired frequency? 2 answers are correct. You may find useful the following references of C2, R20 and R19.

3. Now you should have understood the role of the trimmer. Indeed, it will increase the value of R_b by a serial association of $R20$ and the right part of $R60$, that can vary from 0 to $100\,\text{k}\Omega$. Tune the trimmer resistance by turning the screw present on the top face of it to obtain a 1 Hz 555 timer (the best you can). To do so, monitor the output signal of the 555 timer using the test point and the oscilloscope.

4. Once the 555 timer is tuned, observe the different signals of it (OUT, THRESH/TRIG and DISCH). What do they represent?

5. **BONUS if you have time** Now that you know how a 555 timer works, build your own on breadboard using the same components as the ones present on the board (except the trimmer, you should not be precise). Don't forget to supply your IC with a DC 5V with the DC power supply!

2.5 Fourth manipulation: the logic gate delays

Another interesting phenomenon to observe is the delay of an inverting gate. To do so, we will use the the AC power supply.

1. Using the AC power supply, generate a square signal of 1 MHz and 5 V peak to peak. Moreover, add a 2.5 V offset such that the minimum value of the square signal is 0 V. Use the oscilloscope to verify your signal.

2. Feed an inverting gate (CD4069UBE) with this square signal. Don't forget to supply your IC with a DC 5V with the DC power supply!

3. Monitor both the voltage before and after the inverting gate using the 2 channels of the oscilloscope.

4. What do you observe? Measure the delay between the input and the output of the inverting gate and compare it to what can be found in the CD4069UBE datasheet (transition time).

2.6 Fifth manipulation: the button bounces

Great, you know how to build a binary clock now! Another thing that is interesting to observe (especially for your project) are the button bounces. We will see how to counteract them using a hardware filter.

1. Observe the button bounces using the reset button test point voltage and the oscilloscope. Why could they be problematic? Why aren't they problematic in the case of the reset button? To do so, use the oscilloscope with an horizontal division of 500 μs or 1 ms. The mode *single* of the oscilloscope might be useful.

2. Reproduce a simple button circuit on breadboard. The electronic schematic is represented in Figure 2.2. Observe the bounces while releasing the button.

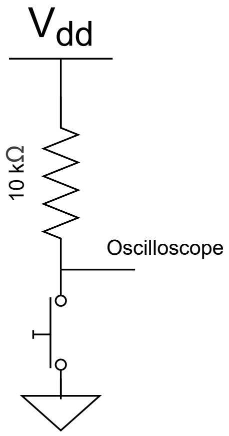


Figure 2.2: Button electronic schematic.

3. Now, we will try to filter out the bounces. Add the filter depicted in Figure 2.3.

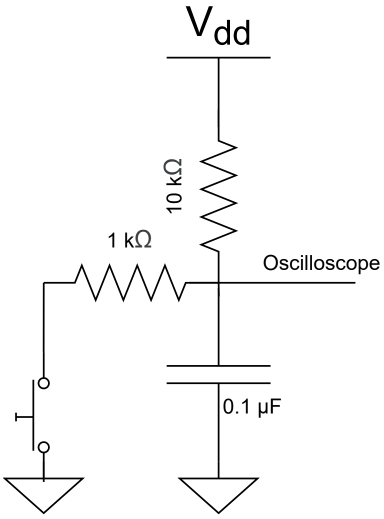


Figure 2.3: Button filter electronic schematic.

4. Do you still observe the bounces when releasing the button? Why?

Appendix A

TTL families

TTL families have been commercially available since the late 1960s. Its advantages are very low cost, high speed, drive capability, acceptable noise immunity, and the availability of hundreds of compatible components. This gives a wide selection of simple or elaborate logic blocks that can be directly interconnected to produce more complicated logic functions.

TTL components can be connected directly to each other, without concern for electrical characteristics. This level of abstraction can be achieved if certain simple connection rules are followed which have been set at design time by studies of voltage levels and current exchanges at the input or output transistors.

Of all the TTL families, the **54/74** series are the most used. The difference between these 2 families is the temperature and voltage ranges (wider for the 54 series). Many semiconductor manufacturers produce TTL circuits. Fortunately, they all accept the same numbering and thus, the same IC number always corresponds to the same element. Each manufacturer just adds a prefix. For example, Texas Instrument uses the prefix SN, Motorola MC, etc.

The nominal supply voltage for TTL is 5 volts. As far as temperature range is concerned, the 74 series operates correctly between 0 and $79°C$ and the 54 series admits temperatures from -55 to $125°C$.

An input not connected in TTL behaves as if the value applied to it were a logic 1 since the transmitter-base junction is not polarized.

After the advent of the standard 74 series, other TTL series were developed. These include:

- **74L** series, low power TTL family;
- **74H** series, fast TTL family;
- **74S** series, TTL family built with Schottky transistors;
- **74LS** series, low power TTL Schottky family;
- **74AS** series, advanced TTL Schottky family;
- **74ALS** series, advanced low power TTL Schottky family.

Table A.1 allows to compare the parameters that characterize each TTL family.

	74	74L	74H	74S	74LS	74AS	74ALS
Performance criterion							
Propagation delay (ns)	9	33	6	3	9.5	1.7	4
Consumption (mW)	10	1	23	20	2	8	1.2
Maximum clock frequency (MHz)	35	3	50	125	45	200	70
Fanout	10	20	10	20	20	40	20
Voltage parameters							
$V_{OH}(V)$	2.4	2.4	2.4	2.7	2.7	2.5	2.5
$V_{OL}(V)$	0.4	0.4	0.4	0.5	0.5	0.5	0.4
$V_{IH}(V)$	2	2	2	2	2	2	2
$V_{IL}(V)$	0.8	0.7	0.8	0.8	0.8	0.8	0.8

Table A.1: TTL families parameters

Appendix B

CMOS families

Currently, the CMOS (Complementary Metal Oxyde Semiconductor) family is becoming more and more important in MSI devices, mainly at the expense of the TTL family. The manufacturing process of CMOS is simpler than TTL and its integration density is higher. CMOS consumes only a fraction of the power dissipated by the low power TTL series (74L). Normally, the operating speed of CMOS is lower than TTL, but the new fast CMOS series now competes with the 74 and 74LS series.

The simplest circuit in CMOS is the inverter. It consists of two MOS transistors in series: the P channel device has its source connected to V_{dd} and the N channel device has its source connected to V_{ss}. The gates of the two devices are joined to form a common input. The drains are also joined to form a common output. This is represented in Figure B.1

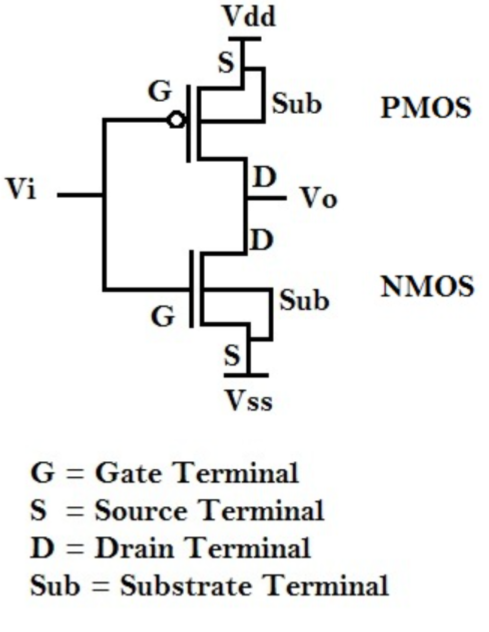


Figure B.1: CMOS inverter schematic

There are a lot of different series in CMOS. The first series was the **4000** series. The **4000B** series is an improved variant providing a higher output current. In addition to these basic families, new series have been added.

- **74C** series, pin-to-pin compatible with the TTL series of the same number and performs the same functions. The performance of this series is about the same as the 4000 series.
- **74HC** series, it is an improvement of the 74C series. The switching speed has been improved by a factor of ten. The speed of these devices is comparable to the 74LS series.
- **74HCT** series, fast CMOS series. The difference with the previous series is that this one has been studied to be compatible in voltage with the TTL series, i.e. it can be connected directly to TTL devices.

Series 4000 and 74C operate at V_{dd} values between 3 and 15 V. The 74HC and 74HCT series operate at voltages ranging from 2 to 6 volts. Thus, when in the same application, we want to use TTL and CMOS circuits, we will use a supply voltage of 5 volts (since it is compatible CMOS and TTL).

In CMOS, the values of V_{OL} and V_{OH} are close to 0 volts and 5 volts. As for V_{IL} and V_{IH}, their value is expressed as a percentage: $V_{IL} = 30\%$ of V_{dd} and $V_{IH} = 70\%$ of V_{dd}.

CMOS inputs must never be left unconnected. All inputs must be connected to a fixed voltage or to another input.

Table B.1 compares the characteristics of the main series of TTL and CMOS integrated circuits.

	74HC	4000B	74	74S	74LS	74AS	74ALS
Consumption per static gate (mW)	0.0025	0.001	10	20	2	8	1.2
Propagation delay (ns)	8	50	9	3	9.5	1.7	4
Speed-Consumption (mW)	10	1	23	20	2	8	1.2
Maximum clock frequency (MHz)	40	12	35	125	45	200	70
Noise margins (worst case) (V)	0.9	1.5	0.4	0.3	0.3	0.3	0.4

Table B.1: Comparison between TTL and CMOS families