ENTIC-lab

Answers to WP questions

Before doing any certain task, the related questions must be answered in this file.

You must update this file as many times as necessary with the format WPAnswers_g_t.pdf

(Keep the Atenea task in draft status until you finish the Lab-Project part A)

.

Question ENTIC_guide.1: *How often must you deliver a Gantt Diagram updated version?*

☐ *at the beginning of each WP*

X☐ *each lab session*

☐ *at the beginning and at the end of Parts A and B*

Question ENTIC_guide.2: *If you are the team 7 of subgroup 63, which is the right filename for deliver the Final Report in Atenea at the end of the semester? What is the deadline day for deliver this document?*

The right filename is “LABFinalReport_63_7.pdf” and the deadline is the 12th of January 2024.

Question ENTIC_guide.3: *What penalization will have you if you deliver a document one week later?*

☐ *15%*

X☐ *30 %*

☐ *50 %*

Question ENTIC_guide.4: *When you must deliver the Answers to the WP questions?*

☐ *at the beginning of each WP*

☐ *at the end of each Part*

X☐ *continuously during the semester, when you need the answer of each question*

Question 1.1: *Why are the holes in the PVC elbows necessary?*

The holes are there to aid the ROUV in keeping it submerged by letting water in and therefore adding the necessary weight to keep it from floating.

Question 1.2: *Calculate how much additional weight is necessary if the holes are not made in the ROUV elbows PVC structure. Inner diameter of PVC pipe: 16 mm.*

S=$(8\cdot 10^{-3})^2\cdot\pi$ = 0,000201 m^2

V=0,0003216 m^3

m=321,69 g

- The PVC pipe total length is **L=160 cm** [8 pieces of 15cm + 2 pieces of 10 cm + 4 pieces of 5cm].

  Cylinder volume $= \pi$. r^2 . h

  V = 160.π.(0,8)^2 = 321,69 cm^3 = 0,321m^3 = 0,321Kg

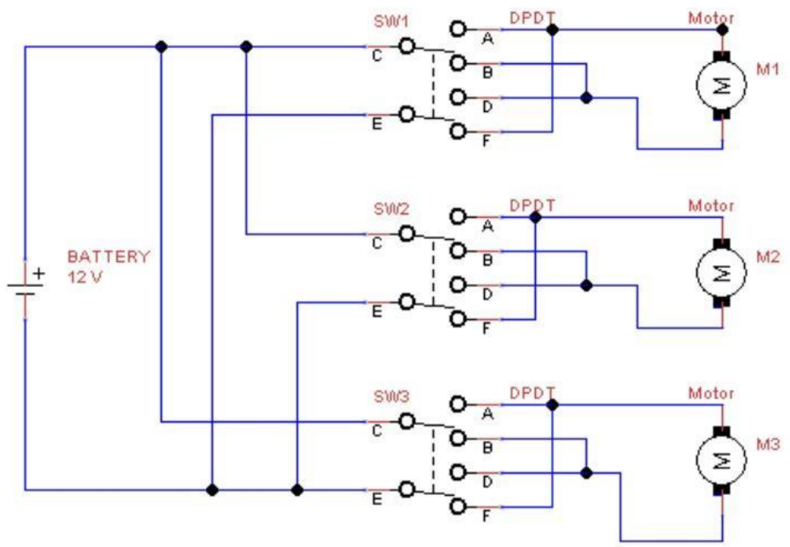
We must add approximate 321,69 g of additional weight if we don't perforate the ROUV.

Question 1.3: *Include a picture of the initial and final Gantt diagrams of WP1, making a reflexion about the progress and explaining any delay in the development of the tasks.*

Question 2.1: *In the market there are different types of switches. Which type among the switch models shown in Fig. 2.2 are we using in this project?*

Since 6 pins are necessary to operate each switch, the model that best fits is the DPDT switch. The 6 pin model is required to move the motor clockwise and anticlockwise (up, down) and no movement (standby).

Question 2.2: *Complete the Fig2.3 sketch of the electrical circuit that controls the motors using the CAD schematic software* ***https://www.tinycad.net/***.



[Question 2.3:](#) *In the soldering in figure 2.5 indicate which ones are correct and which are not and explain the reasons of that.*

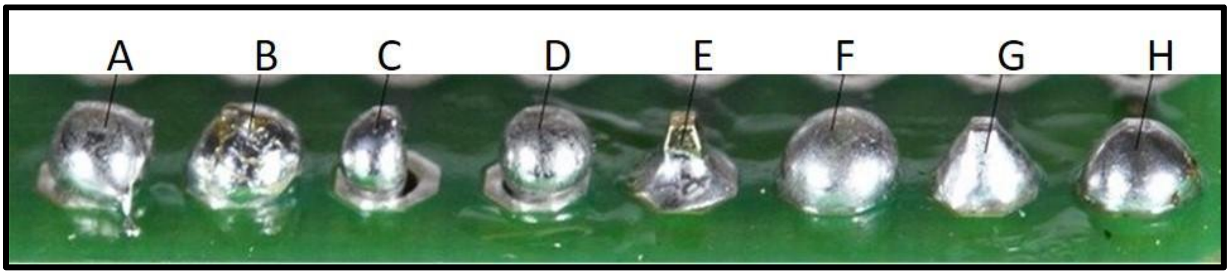


Fig 2.5 Correct and wrong soldering

The perfect solder is E because its shape resembles a volcano. The second to last solder, G, is correct, because of its cone shape, but not perfect. H and F could be considered to pass just barely, but there is too much solder on them. Though the first four (A, B, C, D) are all incorrect, they barely cover the surface of the connection, and all the shapes are barely holding themselves together.

Question 2.4: *The ROUV motors are the core of nautical bilge pumps* *Johnson L450* *model (specifications are available in ATENEA Support Documents section).*

Which is the electrical power required for one ROUV motor?

Analysing the specifications of the Johnson L450, we realise that the engine requires a voltage of 12V. The document also shows the current consumption, which is 2.5 Amps.

The power required for just one engine is:

$P = V \cdot I$ [W] (Ohm's Law)

$$P = 12\,V \cdot 2.5\,A = 30\,W$$

Calculate the total electrical power the ROUV battery must supply.

We have an amount of 3 engines, so the power required is three times the power of one engine.

$$P_{total} = 3 \cdot 30\ W = 90\ W$$

The battery must supply 90 W of power.

Which is the value of the supplied fuse?

The supplied fuse's value is 5 Amps.

Question 2.5: *Include a picture of the initial and final Gantt diagrams of WP2, making a reflexion about the progress and explaining any delay in the development of the tasks.*

Question 3.1: *What pressure increase will be observed at 1 m depth? And at 3 m depth?*

When the depth is 1m, the pressure will be 0.1 bar.

When the depth is 3m, the pressure will be 0.3 bar.

To calculate these values, you just have to use the definition of Atmosphere, where each Atm corresponds to a 10 m column of water.

Question 3.2: *The datasheet of the sensor is available in Atenea. Identify there the full scale output, and then deduce the sensor sensitivity and its value when the supply voltage is not 10 V but 5 V. Do the same for a supply voltage of 3.3V.*

To answer this question, we already have some info like the following:

$$S = \frac{V}{P}$$

· When the supply voltage is 10 V, the sensitivity equals 0.4.

$$P = \frac{V}{S} \rightarrow P = \frac{10}{0.4} = 25000\ kPa$$

· When the supply voltage is 5 V, the sensitivity equals to:

$$S = \frac{5 \cdot 10^3\ mV}{25000\ kPa} = 0.2\ \frac{mV}{kPa}$$

When the supply voltage is 3.3 V, the sensitivity equals to:

$$S = \frac{3.3 \cdot 10^3\ mV}{25000\ kPa} = 0.132\ \frac{mV}{kPa}$$

Question 3.3: *Assuming 5V as the supply voltage, what output voltages V_0 will provide the sensor at the water surface (P=100 kPa), at 1m depth and at 3 m depth? Do the same for a supply voltage of 3.3V.*

· Assuming that the supply voltage is 5 V, we have a sensitivity of 0.2 $\frac{mV}{kPa}$ as we calculated in the question 3.2.

$$V_o = S \cdot P$$

$$\cdot \text{ At water surface: } V_o = 0.2 \cdot 100 = 20\, mV$$

$$\cdot\text{At 1m depth: } V_o = 0.2 \cdot 110 = 22\, mV$$

$$\cdot\text{At 3m depth: } V_o = 0.2 \cdot 130 = 26\, mV$$

For a voltage supply of 3.3 V we've a sensitivity of 0.132 $\frac{mV}{kPa}$. To calculate the output voltages, we use the same formula again.

$$\cdot\text{At water surface: } V_o = 0.132 \cdot 100 = 13.2\, mV$$

$$\cdot\text{At 1m depth: } V_o = 0.132 \cdot 110 = 14.52\, mV$$

$$\cdot\text{At 3m depth: } V_o = 0.132 \cdot 130 = 17.16\, mV$$

Question 3.4: *In a real case and due to changing atmospheric conditions, the pressure at the water surface can be different than 100 kPa. Discuss how this effect can be corrected to always obtain a correct depth measurement.*

We calibrate the instrument with the real measurement at the surface. That is to say, we replace the 100kPa assumed value with the measured value.

Question 3.5: *Obtain the gain* of the amplifier *for the values of R_1 to R_4 provided in the INA126 datasheet. Then, compare it with the expression provided there and identify the role of R_g as a gain trimmer.*

By consulting the INA126 datasheet, we get the values of $R_{1,2,3,4}$.

$$R_1 = R_4 = 40\ k\Omega$$

$$R_2 = R_3 = 10k\Omega$$

The gain of the amplifier is the following:

$G = \frac{V_O}{V_2 - V_1} = \left(\frac{R_1}{R_2} + \frac{R_1+R_4}{R_g} + 1\right)$, if we use our values the gain is:

$$G = 5 + \frac{80\ k\Omega}{R_g}$$

Due to R_g is in the denominator, the gain of the amplifier will decrease if R_g increases.

Question 3.6: *According to the datasheet, what is the maximum value of the output voltage that can provided by the amplifier (saturation voltage)?*

If we have a look at the datasheet, we find that the maximum value of the output voltage (saturation voltage) is the following:

$$V_{max} = V^{+} - 0.75\,V \;=\; 5 \;-\; 0.75 = \;4.25\,V$$

Question 3.7: *Which amplifier gain is necessary to obtain the maximum amplifier voltage output for 3 m depth and 5V of supply voltage? Which R_g resistor value will provide this gain? Do the same for 1 m depth and a supply voltage of 3.3V.*

As we saw in Question 3.3, the output voltage with 5 V of supply voltage at 3m depth is 26 mV. The gain is the following:

$$G = \frac{V_{in}}{V_{out}} = \frac{4.25\,V}{26\;\cdot\;10^{-3}V} = 163.46$$

R_g's resistance will be:

$$R_g = \frac{80}{163.46 - 5} = 0.50485\;k\Omega = 504.85\;\Omega$$

We can not find the exact value of resistance but we can choose the nearest to our wanted value, which is $510\;\Omega$. If we choose this resistor, the gain will change to 161.86.

When we have a supply voltage of 3.3 V at a depth of 1m, the output voltage is 14,52 mV.

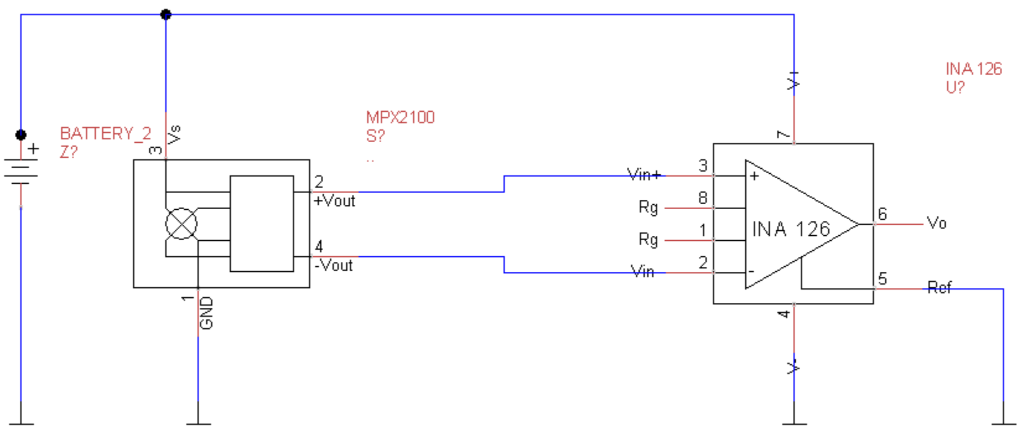
We proceed in the same way as before to calculate the gain:

$$G = \frac{2.55}{14{,}52\;\cdot\;10^{-3}} = 175.62$$

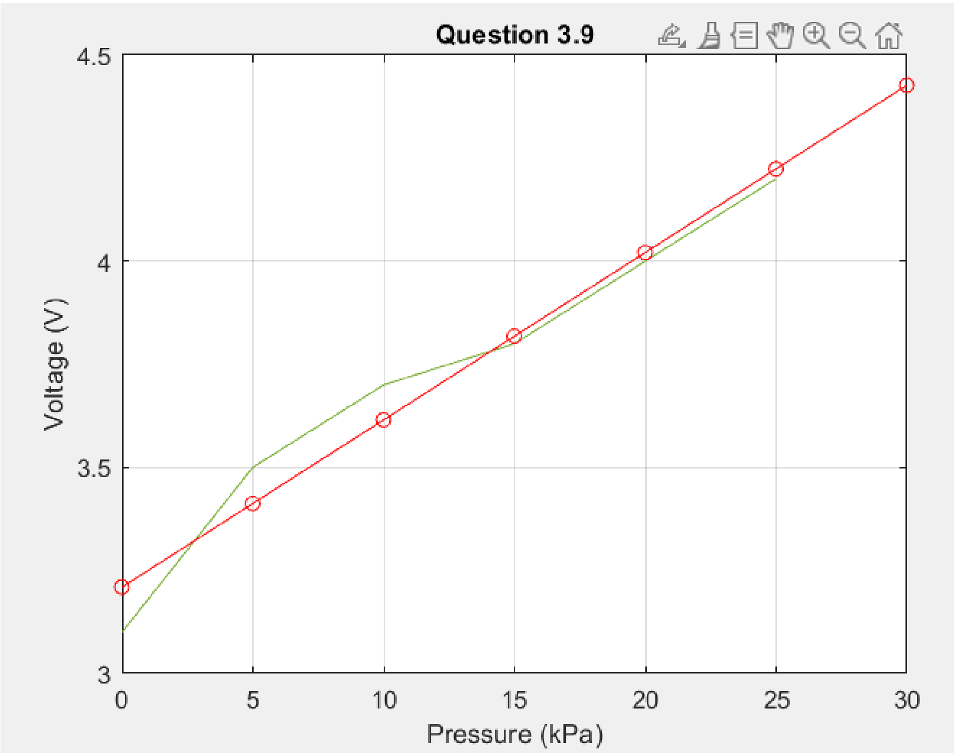
$$R_g = 0.46425\;k\Omega = 464.25\Omega$$

The closest equivalent resistance is a series connection of 470 Ω and 33 Ω resistors.

Question 3.8: *Using the ENTIC library available in Atenea, draw the sensor and amplifier schematic with the CAD schematic software available in* http://www.tinycad.net/.



[Question 3.9:](#) *Obtain in the lab 10 measurement points (P(Kpa), Vo (volts)) along the measurement range. Draw its graphical representation, calculate and plot the linear regression and the linearity error (lineal regression straight line – measured points).*



```
v=[3.1,3.5,3.7, 3.8, 4, 4.2];
p=[0, 5, 10, 15, 20, 25];
plot(p,v);
range=1:6;
a=polyfit(p(range), v(range),1);
x=0:5:30;
y=polyval(a,x);
hold on
plot(x,y,'ro-','LineWidth', 0.5);
grid on
plot(x,y,'ro-','LineWidth',0.5);
xlabel('Pressure (kPa)');
ylabel('Voltage (V)');
title('Question 3.9');
plot(x,y,'ro-', 'LineWidth', 0.5);
```

Question 3.10: *Calculate the digital data range provided by the acquisition circuit in measurements from 0m to 3m of water depth. Consider that the 10 bit A/D converter of the Arduino assigns 0 to a 0V input and 1023 (2^{10}-1) to a 5V input.*

It's a simple proportional relationship to map a range of 1023 to 5:

$$V_o(x) \cdot G \cdot \frac{1023}{5} = D(x)$$

We saw that voltage ranged from 20 mV to 26 mV and so plugging in to the formula we get

$$V_o(0) \cdot G \cdot \frac{1023}{5} = \text{20e-3} \cdot 163.46 \cdot 1023 / 5 = 668{,}88$$

$$V_o(3) \cdot G \cdot \frac{1023}{5} = \text{26e-3} \cdot 163.46 \cdot 1023 / 5 = 869{,}54$$

And so the digital data range resulting from measurements from 0 bar to 3 bar is from approximately 670 to 870.

Question 3.11: *Explain briefly what is the purpose of the following functions, used in the sketch of Fig. 3.4: **Serial.begin**(), **analogRead**(), **Serial.println**() and **delay**().*

According to the Arduino library docs:

Serial.begin() sets the data rate in bits per second (baud) for serial data transmission.

analogRead() reads the value from the specified analog pin.

Serial.println() prints data to the serial port as human-readable ASCII text followed by a carriage return character (ASCII 13, or '\r') and a newline character (ASCII 10, or '\n').

delay() pauses the program for the amount of time (in milliseconds) specified as parameter

---

Question 4.1: *For the following mesured values with arduino: 663, 728, 1012.*

a) Determine the pressure and the depth corresponding to each value.

- **663**

V=$\frac{663.5}{1023}$ = 3,24

V_{in}=$\frac{V_O}{G} = \frac{3,34}{163,46}$= 20,43 mV

20,43 mV/0,2=102,15 kPa

pressure=102,15 kPa we conclude that the depth is **0,21 m**.

-**728**

V=$\frac{728.5}{1023}$ = 3,55

V_{in}=$\frac{V_O}{G} = \frac{3,55}{163,46}$= 21,76 mV

21,76 mV mV/0,2=108,83 kPa

pressure=108,83 kPa we conclude that the depth is **0,88 m**.

- **1012**

V=$\frac{1012.5}{1023}$ = 4,94

V_{in}=$\frac{V_O}{G} = \frac{4,94}{163,46}$= 30,22 mV

30,22 mV/0,2=151,1 kPa

pressure=151,1kPa we conclude that the depth is **5,11 m**.

b) *Encode each value in plain binary (2 bytes)*

$\mathbf{(663)_{10}}$ = $(10\ 10010111)_2$

$\mathbf{(728)_{10}}$ = $(1011011000)_2$

$\mathbf{(1012)_{10}}$ = $(1111110100)_2$

c) *Encode each value in ASCII.*

663: *54 54 51*

728: *55 50 56*

1012: *49 48 49 50*

Question 4.2: *Go to the Arduino language reference and identify which types are suitable to work with the numeric data provided by the A/D converters. What code length (in bytes) corresponds to each type?*

Type of Data	***length (in bytes)***
Int	*2 bytes*
Double	*4 bytes*

Float	*4 bytes*
Long	*4 bytes*
ASCII	*3 bytes*
Short	*2 bytes*
Byte	*1 byte*

Question 4.3: *Compare the screen captures obtained and explain the differences found. What sequence of bytes is really sent in each case?*

The difference between both comes from the fact that /r/n is not placed at the end of the string in the case of Serial.print() so it appends to the last string and appears all in the same line.

Question 4.4: *Open the Windows Device Manager and find the serial COM ports defined on your computer. Plug the USB to RS232 adapter into a USB port on your computer. What is the new COMx port associated with this adapter?*

The COM port of the RS232 adapter in our case is 6, so COM6.

Question 4.5: *Identify the cables color connected at the RS-232 send and receive pins.*

GND: Pink (Ground)

TXD: Yellow (Transmit Data)

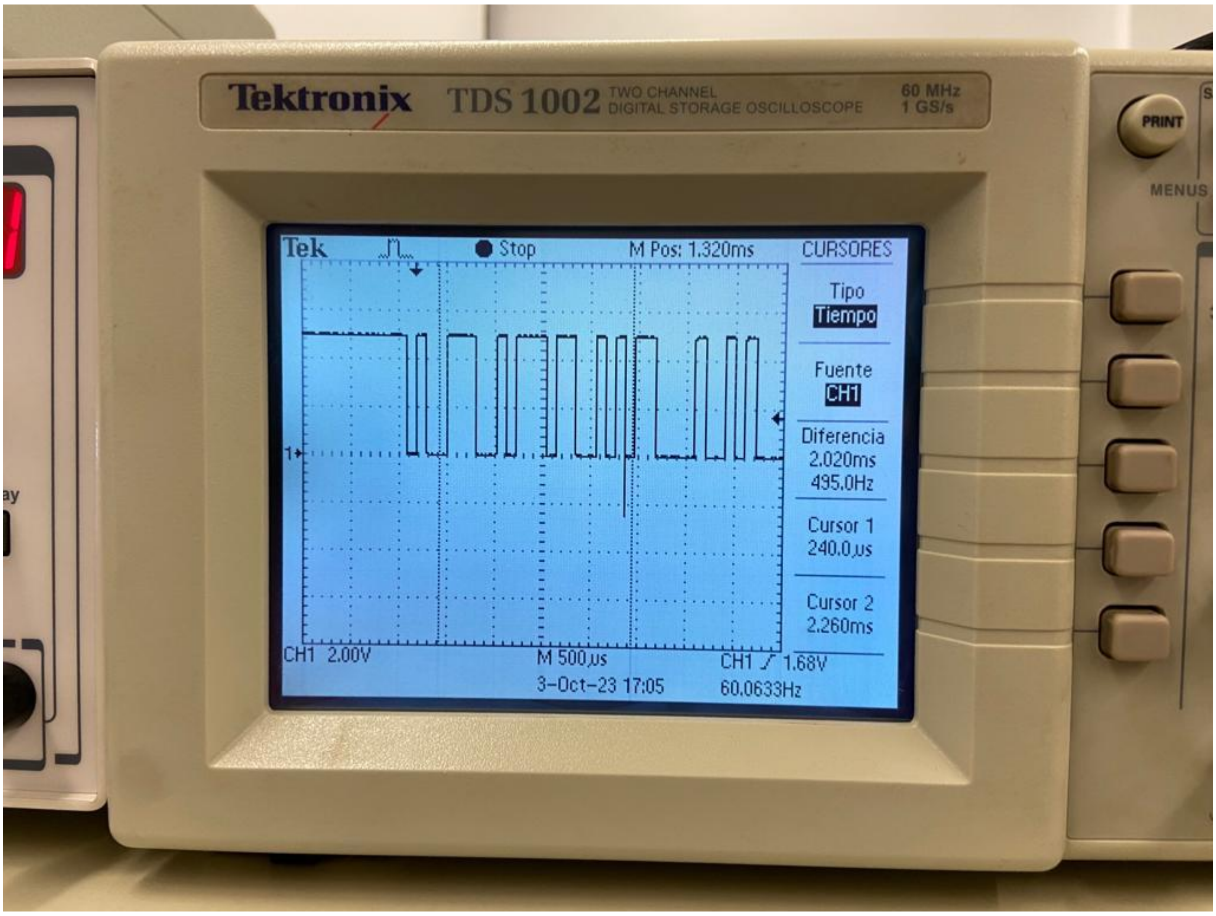
RXD: Blue (Receive Data)

Question 4.6: *Calculate which time-base must be selected in the oscilloscope to see 10 bits on the screen at the transmission rate selected by default in **Terminal**.*

10 bit · (1s/9600bits) = 1, 04 ms → 1,04ms/10≃100us.

The correct time-base is 100us.

Question 4.7: *Capture the waveform displayed on the oscilloscope. Determine the voltage levels, bit period and the number of bits per character. Which level indicates the idle state? Using an ASCII table explain the waveforms displayed.*



Voltage levels are 5.52Vpp (peak-to-peak) and bit period (duration of each bit) is 104us.

Each character is encoded in 20 bits, the first 10 bit hex ASCII reverse data stream represents the tens place of the decimal ASCII real character and the second represents the ones place of the real ASCII character, the 'a' in this case.

Idle state is ~5V.

We sent the character 'a', which in ASCII is 97d (decimal). Asynchronous serial transmission encodes 8 data bits after a start bit and finally a stop bit. So if you read what the oscilloscope shows (the last two bits are missing), you can represent the following:

Data stream 1 (8b)	Data stream 2 (8b)	Data stream 3 (8b)	Data stream 4 (8b)
10011100	11101100	10110000	01010000
Reverse	Reverse	Reverse	Reverse
0b00111001	0b00110111	0b00001101	0b00001010
0x39	0x37	0x13	0x10
ASCII of 0x39	ASCII of 0x37	ASCII of 0x15	ASCII of 0x10

'9'	'7'	'\r' or CR or ^M	'\n' or LF or ^J

Notes:

- Before and after each data stream, there is a start and stop bit.
- The last two bits are not shown in the oscilloscope display but are 01, the 0 is the last data bit and the 1 is the last stop bit.

And '97' parsed as a decimal integer is 0d97, which in ASCII translates to 'a'.

Question 4.8: *Taken into account the ROUV specifications, what is the reason to use RS-232 to communicate with the ROUV, instead of using USB (which is available in both the Arduino and the computer)?*

The RS-232 standard results in cable lengths: 10 Kb/s can be sent reliably through a 1 km cable while with USB you can at most use a 5-meter cable.

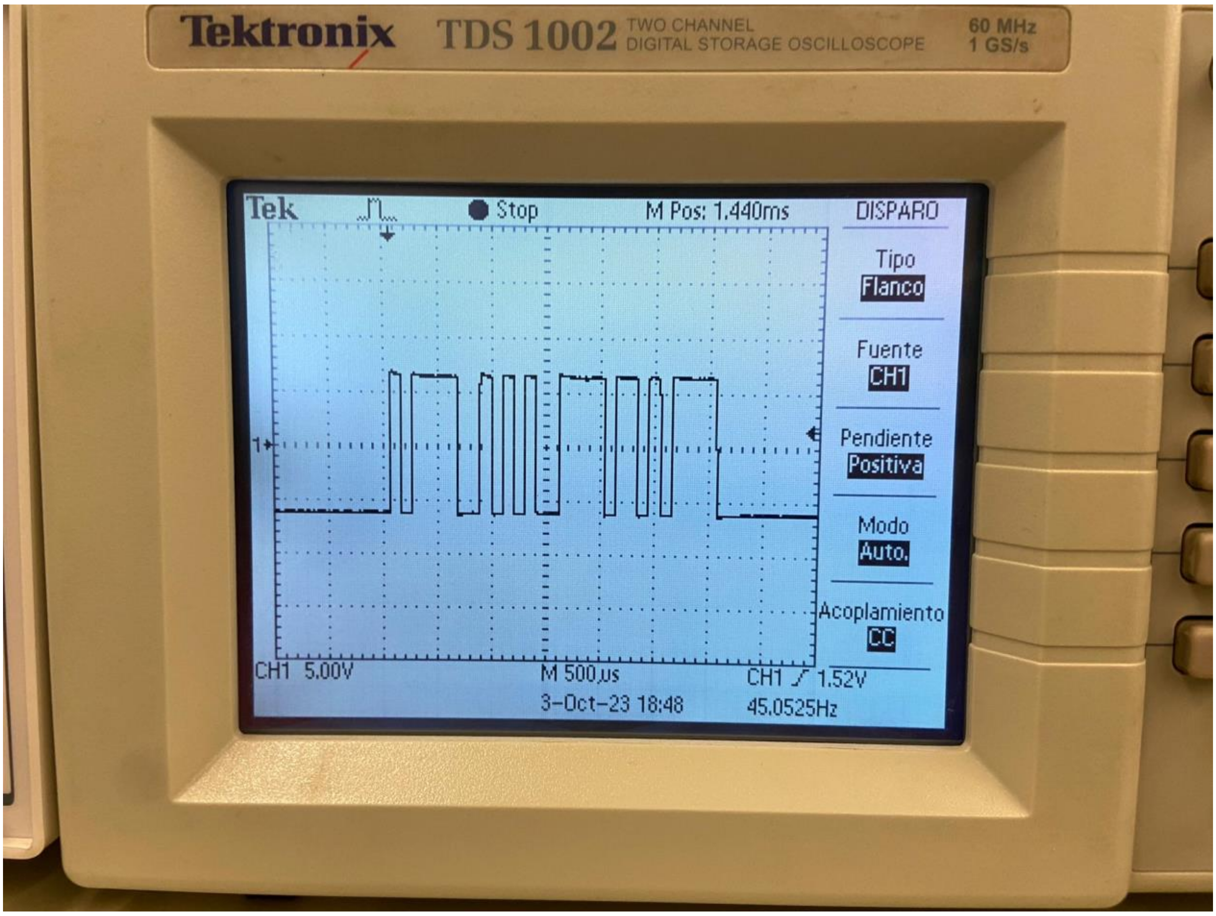
Question 4.9: *Capture the waveform displayed on the oscilloscope. Obtain the voltage levels of the signal. Which level corresponds to the idle state?*

Voltage levels are ~10Vpp (peak-to-peak) and bit period (duration of each bit) is 104us.

Each character is 10 bits.

Idle state is ~5V.

We sent the character 'a', which in ASCII is 97d (decimal). Asynchronous serial transmission encodes 8 data bits after a start bit and finally a stop bit. So if you read what the oscilloscope shows (the last two bits are missing but assumed to be 01), you can represent the following:



Data stream 1 (8b)	Data stream 2 (8b)	Data stream 3 (8b)
10000110	10110000	01010000
Reverse	Reverse	Reverse
0b11000001	0b0001101	0b00001010
0d97	0d13	0d10
ASCII of 0d97	ASCII of 0d13	ASCII of 0d10
‘a’	‘\r’ or CR or ^M	‘\n’ or LF or ^J

Note: Before and after each data stream, there is a start and stop bit.

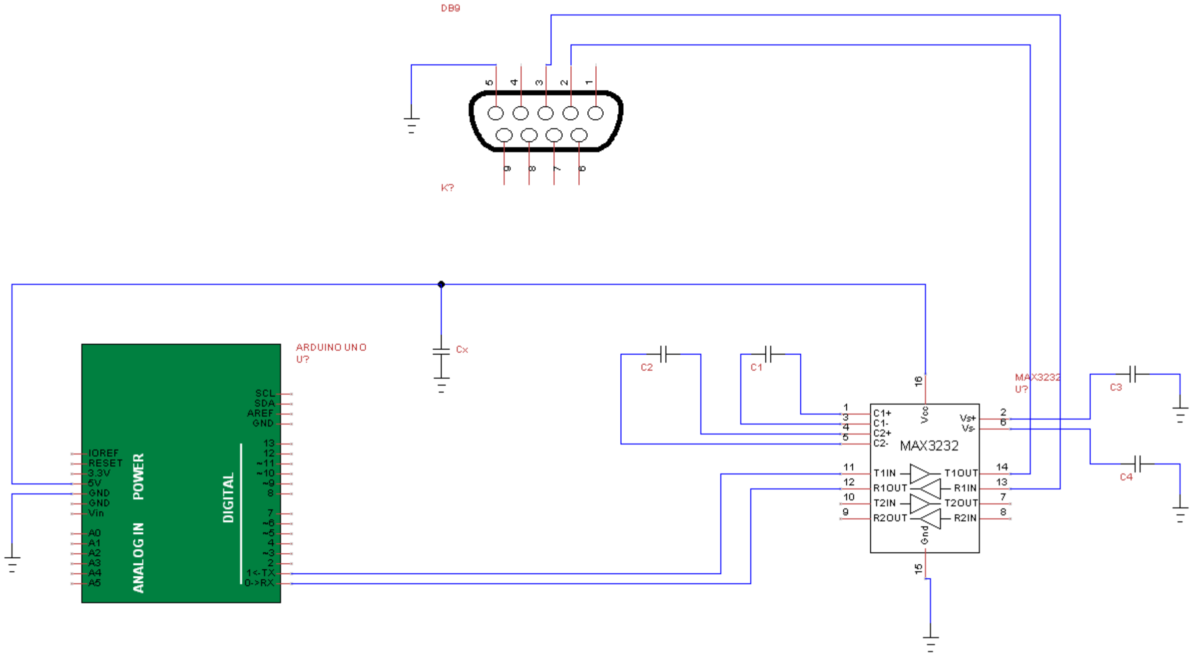
Question 4.10: *Draw the schematic (using the CAD schematic software https://www.tinycad.net/) of the level shifter circuit, including only the transmission path (from the Arduino Tx pin to the RS-232 DB9-female connector). What are the appropriate values of capacitors C1, C2, C3 and C4?*

V_{CC} vs CAPACITOR VALUES

V_{CC}	C1	C2, C3, C4
3.3 V ± 0.3 V	0.1 µF	0.1 µF
5 V ± 0.5 V	0.047 µF	0.33 µF
3 V to 5.5 V	0.1 µF	0.47 µF

$$C_1 = 0{,}047\mu F$$

$$C_2 = C_3 = C_4 = C_x = 0{,}33\mu F$$

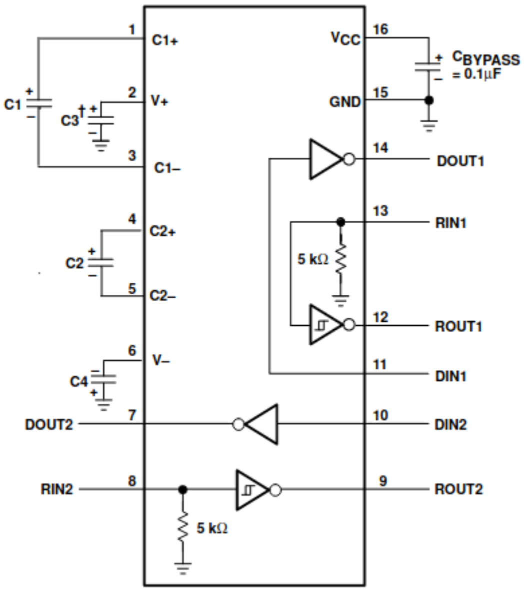


Question 4.11: *Describe the difference(s) between the sketches of Figs. 4.1 and 4.5 seen from the user.*

The differences between Figures 4.1 and 4.5 are quite straightforward. In Figure 4.5, the code prompts the user for instructions and allows them to input a character, which then triggers a loop to print numbers. In comparison, Figure 4.1 doesn't offer the option for user input; it simply prints the character as directed by the code.

Question 4.12: *Explain why the Rx input pin of the Arduino must be in high impedance state (i.e. not connected) when a compiled sketch is being uploaded.*

APPLICATION INFORMATION



In the Arduino schematic, you can see the lines M8TXD and M8RXD that leave from the ATmega16U2 and connect to the PD0 (Rx) and PD1 (Tx) pins of the ATmega328. These lines transmit the converted TTL pulses from USB via the ATmega16U2, which connects to the ATmega328 to establish communication between the main microcontroller (ATmega328) and the PC. These lines are connected to Rx and Tx as they are the pins of transmission and reception that connect to the USART of the microcontroller. So in the process of uploading the sketch, the Rx and Tx pins of the Arduino are occupied establishing USART communication between the PC and ATmega328 (through the ATmega16U2's usb-to-serial installed firmware).

The Rx pin is connected to the ROUT1 pin of the MAX232 which while uploading the sketch will create interference because the ROUT1 pin has a 5K resistance connected to GND and a significant amount of current will be pulled to GND and direct enough power away from the ATmega16U2 to interfere with the serialized data and corrupt the voltage values. The error detection then kicks in and exits the upload process

Question 4.13: *Modify the sketch* ***ENTIC-Analog2Serial-bidir*** *to stop data sampling & transmission when any of the following conditions is true:*

1) *timeout, i.e. data values must be sent for only 2 minutes;*
2) *end command from user, i.e. the Arduino receives the character 'E'.*

```
int outValue = 0;
long timeout = 120000; // 120000 milliseconds is 120 seconds (2 minutes)

void setup() {
 Serial.begin(9600);
 while (Serial.available() <= 0 || Serial.read() != 'S') {
        Serial.println("Send 'S' to start");
        delay(200);
 }
 Serial.println("Starting measurement");
```

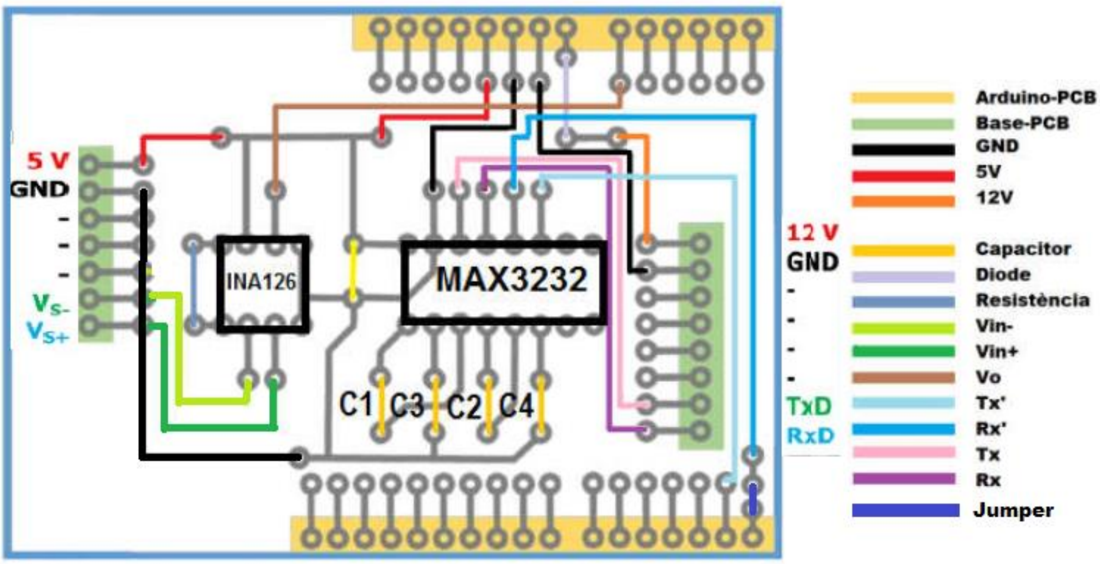
```
}

void loop() {
        time = millis();
        if (time >= timeout) exit(0);
        if (Serial.read() == 'E') exit(0);
        outValue = analogRead(A0);
        Serial.println(outValue);
        delay(200);
}
```

Question 4.14: *Explain and justify where these three components should be placed in Fig. 4.8:*

a) *The resistor setting the amplifier gain (see WP3).*
   Connected to pin 1 and pin 8 of the INA126 as it says in the support documentation.
b) *The jumper for the serial reception path (see Fig. 4.4).*
   The jumper connects pin 12 of the MAX3232 to the Arduino Rx pin.
c) *The protection diode for the Arduino power input (see Fig. 4.9).*

   The diode must be connected between the Arduino power pins Vin and the battery, to protect the Arduino board against sudden changes of voltage polarity and amperage peaks.

Question 4.15: *Using Fig. 4.8 as floorplan, draw a schematic describing how the circuits must be implemented on the interface-PCB. Identify and label all interconnection signals with the Arduino and the base PCBs (strip pins & sockets), describe the placement of all electronic components and add rigid cables whenever additional connections were necessary. Use a colour code to improve the readability and usefulness of the schematic (i.e. red for power supply tracks & cables, black for GND, etc.).*



[Question 4.16:](#) *Include a picture of the initial and final Gantt diagrams of WP4, making a reflexion about the progress and explaining any delay in the development of the tasks.*

[Question 5.1:](#) *Write the instructions to generate with Matlab a vector of 1000 measured pressure points according to:*

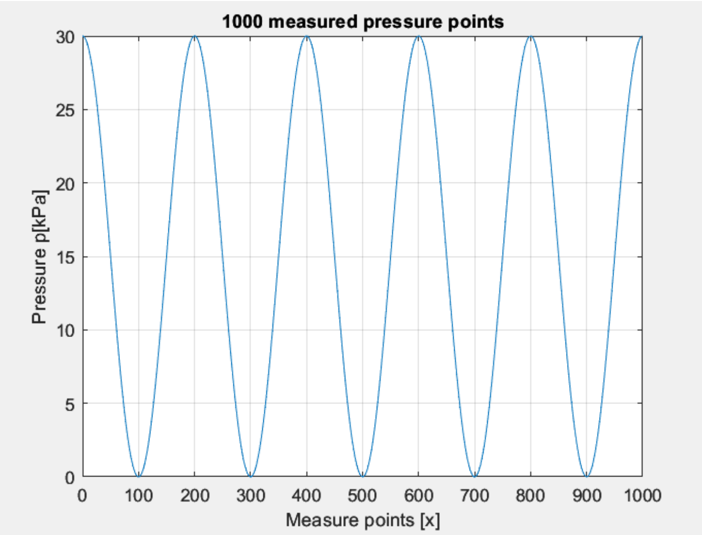
- *the pressure variation must follow a sinusoidal shape. The vector must contain 5 sinusoidal cycles.*
- the values of the simulated pressure signal must match the values obtained by the Arduino's A/D converter when it acquires the information from the pressure sensor and the amplifier.
- consider the maximum and minimum depth of the submarine specifications.

```
x=0:1:999;
frequency=5/1000;    %1/200
MaxPressure=130;     %Max Pressure
MinPressure=100;     %Min Pressure
offset=(MaxPressure+MinPressure)/2;  %Offset
Amplitude=offset-MinPressure;
Pressure=Amplitude*cos(2*pi*frequency*x)+offset;   %Pressure function
```

Question 5.2: *Write the Matlab instructions to plot the complete graph of 1000 points. The horizontal axis must be an index vector: 1,2,3,4, …*

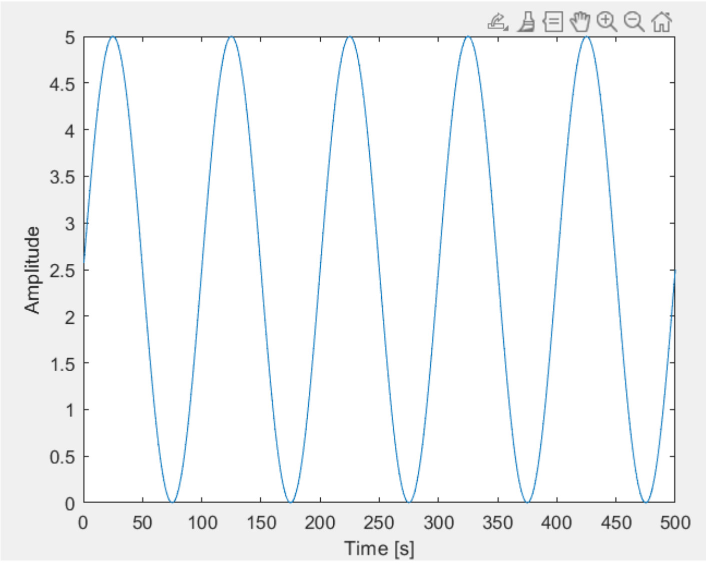
The code we have to run to get the graph is:

```
x=0:1:999;
frequency=5/1000;    %1/200
MaxPressure=130;     %Max Pressure
MinPressure=100;     %Min Pressure
offset=(MaxPressure+MinPressure)/2;  %Offset
Amplitude=offset-MinPressure;
Pressure=Amplitude*cos(2*pi*frequency*x)+offset;   %Pressure function

%PRESSURE FUNCTION PLOT (5.2)
plot(x,Pressure);
grid on;
title('1000 measured pressure points');
xlabel('Measure points [x]');
ylabel('Pressure p[kPa]');
```



Question 5.3: *Given that the 1000 points have been acquired at the rate of 2 per second, write the instructions to generate a pressure-time matrix. Each row must be a (pressure, time) sample. Draw a new plot, now with the horizontal axis in seconds.*

```
i=1:1000;
v=2.5*sin(pi*i/100)+2.5;
t=0.5:0.5:500;
Matrix=[t,v];
plot(t,v);
xlabel("Time[s]");
ylabel("Amplitude");
```



Question 5.4: *Write a new Matlab script to save the simulated pressure measurements in a .txt file. Check that the data in the generated file is correct with a text file editor. Create other files with depth shapes different from the sinusoidal one (triangle, square, …).*

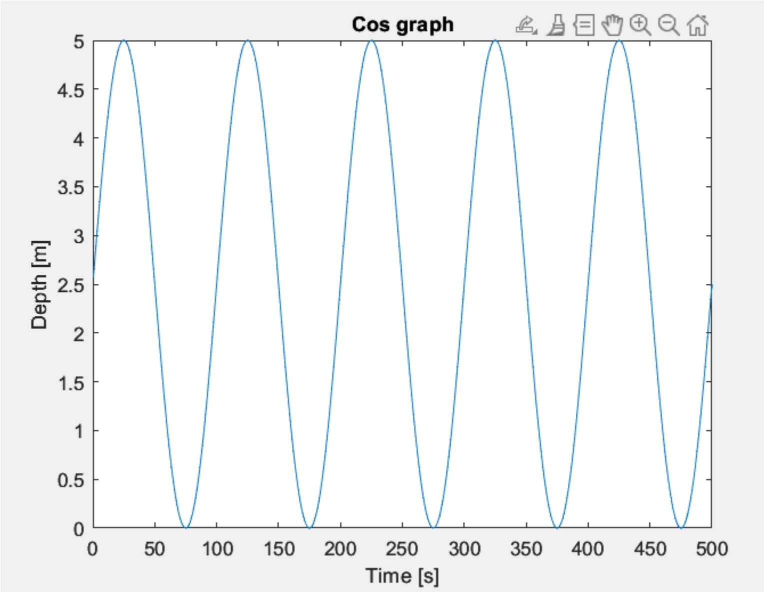
The script could be like this (We use float for the measurements) :

```
clear all;
i=1:1000;
sinusoidal=2.5*sin(pi*i/100)+2.5;
t=0.5:0.5:500;
%Matrix=[t,v];
file1=fopen("pressureMeasurmentsSinusoidal.txt", "w");
formatSpec='%f';
for j=1:1000
    fprintf(file1, '%f %f\n', t(j), sinusoidal(j));
end
fclose(file1);
%%
i=1:1000;
triangularWave=sawtooth(pi*i/100);
t=0.5:0.5:500;
file2=fopen("pressureMeasurmentsTriangularWave.txt","w");
formatSpec='%f';
for j=1:1000
    fprintf(file2, '%f %f\n', t(j),triangularWave(j));
end
fclose(file2);

%%
i=1:1000;
x=square(i);
t=0.5:0.5:500;
file3=fopen("pressureMeasurmentsSquareWave.txt","w");
formatSpec='%f';
for j=1:1000
    fprintf(file3, '%f %f\n', t(j),x(j));
end
fclose(file2);
%%
```

Question 5.5: *Write a new Matlab script to plot the data saved in the files. Units: meters & seconds.*

Matlab script to “read” and load the data from the text file (“pressureMeasurments.txt”).

```
data= load('pressureMeasurments.txt');
T=data(:,1);
T2=data(:,2);
plot(T,T2);
xlabel('Time [s]');
ylabel('Depth [m]');
title('Cos graph');
```



Question 5.6: *Adapt the script developed in 5.5 to read the data Arduino sends on the serial port and to draw a plot of the ROUV depth. In your Matlab code, include the necessary instructions to open and close the serial port (which will be used once the measurements and collection of data finishes).*

Question 5.7: *Write a Matlab script to start&stop the measures on the ROUV (you will be sending data from the computer through the control box and finally to the Arduino). Synchronization with the sketch running in the Arduino (**ENTIC-Analog2Serial-bidir** or alternative) will be necessary.* Finally, the collected depth data (in meters) has to be saved in a *.txt* file named **resultats.txt** for future manipulation. *Write the instructions to create, fill, and close the file* **resultats.txt** *using Matlab.*

Question 5.8: *Develop the whole Matlab code to display real-time depth data (depth vs. time). It is important that you apply some technique to avoid noisy points in the graph. Explain your solution. It is also interesting to record (and plot) the maximum and minimum depth during the ROUV journey. Show two representative Maltab plots and paste them in your written report.*

Question 5.9: *Include a picture of the initial and final Gantt diagrams of WP5, making a reflexion about the progress and explaining any delay in the development of the tasks.*