Workshop 3

Yuxuan Han, Fleming Society

November 21, 2023

In this workshop, you will learn:

- How can Arduino board use serial communication to communicate with other devices.
- What is a voltage divider and how to calculate resistance using analogRead() results.
- How to measure the temperature using a thermistor.

You are supposed to complete the following tasks during the workshop:

1 Workshop Task 1: *"Hello World"* using serial transmission

In this task, you will create an Arduino program that transmit *"Hello World"* to the computer every one second through serial communication.

To do some basic serial communications, you need to use two new Arduino functions, which are introduced below:

Serial.begin()

Description:

Setup serial communication using the specified data rate in bits per second (baud).

Syntax:

```
Serial.begin(speed);
```

Parameters:

speed : Transmission data rate in bit per second (baud). Allowed data type: long

Returns:

N/A

Serial.print()

Description:

Prints data to the serial port as human readable ASCII text.

See Documentation for more details.

Use Serial.println() if you want the function to begin a new line every time after executed. Syntax is the similar.

Syntax:

```
    Serial.print(val);
```

Parameters:

`val` : the value to print. Allow data types: any data type

Returns:

Number of bytes written.

In this task, you need to:

First:

In `void setup()` , setup serial communication using a typical baud rate, e.g. 9600.

Next:

In `void main()` , use `Serial.print("Hello World")` to print *Hello World* on the serial monitor. Use `delay()` to add a delay after printing.

Note: You can change `Serial.print()` to `Serial.println()` and compare the difference.

Finally:

Compile and upload your program.

On the top-right corner of the Arduino IDE, click the icon to open serial monitor[1]:

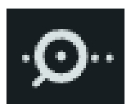


Figure 1: Serial Monitor

Select the baud rate that you used when setting up the serial communication, and watch the results appear on your serial monitor.

2 Workshop Task 2: Measure resistance of a potentiometer.

In this task, you need to measure the resistance of a potentiometer and display it on the serial monitor. Each of you will be given a potentiometer (See *Fig. 2*)[2] and a resistor with known resistance. You need to connect them in series to create a voltage divider (See *Fig. 3*).

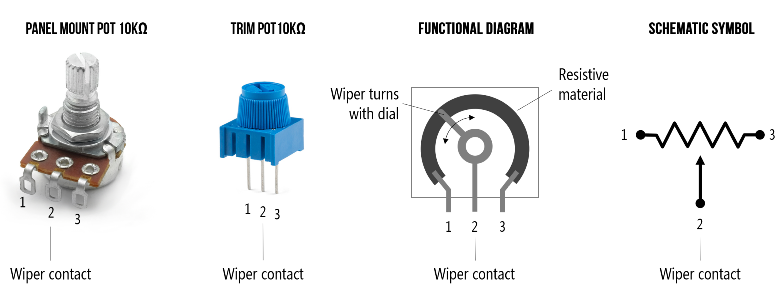
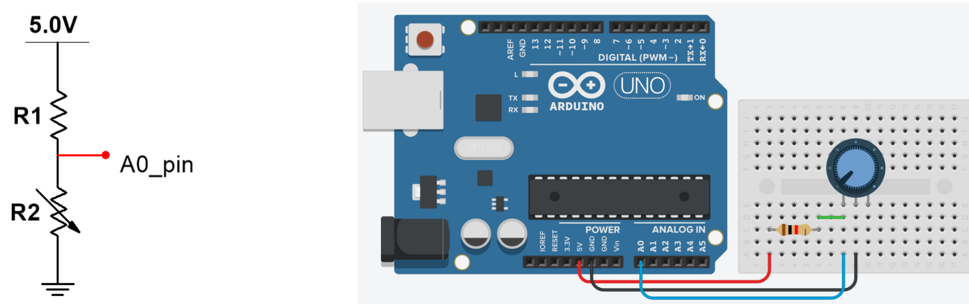


Figure 2: Examples of potentiometers.

[1] Instruction based on Arduino IDE version 2.2.1, steps may be different for older versions.
[2] Figure Source: https://makeabilitylab.github.io/physcomp/arduino/potentiometers.html



(a) Circuit diagram. (b) Circuit connection.

Figure 3: Task 2 circuit

Since analogRead() is measuring the potential difference between the input voltage of the pin and the ground, to measure the resistance of the potentiometer, you need to make sure only the potentiometer is connected between your analog pin used and the ground. (Or you may use the Krichhoff Voltage Law instead).

Applying Ohm's Law to the voltage divider shown in *Fig. 3a*, and we can derive the expression for R_2:

$$I = \frac{V}{R_{tot}} = \frac{5}{R_1 + R_2} \tag{1}$$

$$V_2 = IR_2 = \frac{5R_2}{R_1 + R_2} \tag{2}$$

$$\Rightarrow R_2 = \frac{R_1 V_2}{5 - V_2} \tag{3}$$

Note: V_2 is the voltage across the potentiometer, which can be measured by using analogRead().

If you have read the documentation for analogRead() before, you may notice that the return value of it is integers from 0 - 1023. This cannot be a voltage value in volt. In fact, the ADC (analog digital converter) in Arduino divided 5 V into 1024 units, from 0 - 1023. This means, to get the real voltage in volts, we need to do the following conversion:

$$V_{volt} = \frac{5}{1024} \times analogRead() = 0.0048828125 \times analogRead() \tag{4}$$

IMPORTANT: Do not use 5/1024, since the low precision in float calculations will give you zero!

Now, you can build the circuit and start programming. Remember the important equations that we need to use (*Eq. 3* and *Eq. 4*). After finish your program, you can verify your results by comparing with the resistance measured by using a multimeter. Ask one of our volunteers to get you the probes if you needed.

3 Workshop Task 3: Measure temperature using a thermistor.

In this task, you are supposed to make a temperature alarm with the following requirements:

- Use a thermistor to measure temperature.
- Upload the real-time temperature to the computer.
- When temperature is lower than the user-setted threshold, green LED illuminates. If temperature is higher than the user-setted threshold, red LED illuminates.

Optional Requirements (When you finished all above correctly):

- Upload real-time temperature every 1 second, BUT NOT measure every 1 second, this means you cannot use the `delay()` approach).
- When red LED illuminates, the beeper also beeps on and off every half second. Temperature measurements should not be affected.

Thermistors can be seen as a kind of variable resistor. The thermistors that we offered you is a $10k\Omega$ NTC (Negative temperature coefficient) thermistor, which means its resistance decrease when temperature increases, and resistance is $10k\Omega$ when temperature is $25°C$.

It is very important to go through the Datasheet when every time you get a new component. The following are some important information extracted from the data sheet that we are going to use today:

ELECTRICAL DATA AND ORDERING INFORMATION

R_{25} (Ω)	R_{25}-TOL. (± %)	$B_{25/85}$ (K)	$B_{25/85}$-TOL. (± %)	UL RECOGNIZED (Y/N)	SAP MATERIAL NUMBER NTCLE100E3....B0/T1/T2 RoHS COMPLIANT WITH EXEMPTIONS (1)	SAP MATERIAL NUMBER NTCLE100E3....B0A/T1A/T2A RoHS COMPLIANT WITHOUT EXEMPTIONS (1)	COLOR CODE (2)		
							I	II	III
330	2, 3, 5	3560	1.5	Y	331*B0	331*B0A	Orange	Orange	Brown
470	2, 3, 5	3560	1.5	Y	471*B0	471*B0A	Yellow	Violet	Brown
680	2, 3, 5	3560	1.5	Y	681*B0	681*B0A	Blue	Grey	Brown
1000	2, 3, 5	3528	0.5	Y	102*B0	102*B0A	Brown	Black	Red
1500	2, 3, 5	3528	0.5	Y	152*B0	152*B0A	Brown	Green	Red
2000	2, 3, 5	3528	0.5	Y	202*B0	202*B0A	Red	Black	Red
2200	2, 3, 5	3977	0.75	Y	222*B0	222*B0A	Red	Red	Red
2700	2, 3, 5	3977	0.75	Y	272*B0	272*B0A	Red	violet	Red
3300	2, 3, 5	3977	0.75	Y	332*B0	332*B0A	Orange	Orange	Red
4700	2, 3, 5	3977	0.75	Y	472*B0	472*B0A	Yellow	Violet	Red
5000	2, 3, 5	3977	0.75	Y	502*B0	502*B0A	Green	Black	Red
6800	2, 3, 5	3977	0.75	Y	682*B0	682*B0A	Blue	Grey	Red
10 000	2, 3, 5	3977	0.75	Y	103*B0	103*B0A	Brown	Black	Orange
12 000	2, 3, 5	3740	2	Y	123*B0	123*B0A	Brown	Red	Orange
15 000	2, 3, 5	3740	2	Y	153*B0	153*B0A	Brown	Green	Orange
22 000	2, 3, 5	3740	2	Y	223*B0	223*B0A	Red	Red	Orange
33 000	2, 3, 5	4090	1.5	Y	333*B0	333*B0A	Orange	Orange	Orange

Figure 4: Thermistor Electrical Data and Odering Information.

From *Fig. 4*, we can find that for our $10k\Omega$ NTC thermistor, the $B_{25/85}$ value is 3977.

The datasheet also told us the resistance or corresponding temperature can be calculated using the following interpolation laws:

$$R_{(T)} = R_{ref} \times \exp\left(A + B/T + C/T^2 + D/T^3\right) \tag{5}$$

$$T_{(R)} = \left(A_1 + B_1 \ln \frac{R}{R_{ref}} + C_1 \ln^2 \frac{R}{R_{ref}} + D_1 \ln^3 \frac{R}{R_{ref}}\right)^{-1} \tag{6}$$

In *Eq.* 5 and *Eq.* 6, $R_{ref} = R_{25} = 10k\Omega$. T is the temperature in K (Kelvin). $A, B, C, D, A_1, B_1, C_1$ and D_1 are constants that can be found in the following table:

PARAMETER FOR DETERMINING NOMINAL RESISTANCE VALUES											
NUMBER	$B_{25/85}$ (K)	NAME	TOL. B (%)	A	B (K)	C (K^2)	D (K^3)	A_1	B_1 (K^{-1})	C_1 (K^{-2})	D_1 (K^{-3})
1	2880	Mat O. with Bn = 2880K	3	- 9.094	2251.74	229098	- 2.744820E+07	3.354016E-03	3.495020E-04	2.095959E-06	4.260615E-07
2	2990	Mat P. with Bn = 3990K	3	- 10.2296	2887.62	132336	- 2.502510E+07	3.354016E-03	3.415560E-04	4.955455E-06	4.364236E-07
3	3041	Mat Q. with Bn = 3041K	3	- 11.1334	3658.73	- 102895	5.166520E+05	3.354016E-03	3.349290E-04	3.683843E-06	7.050455E-07
4	3136	Mat R. with Bn = 3136K	3	- 12.4493	4702.74	- 402687	3.196830E+07	3.354016E-03	3.243880E-04	2.658012E-06	- 2.701560E-07
5	3390	Mat S. with Bn = 3390K	3	- 12.6814	4391.97	- 232807	1.509643E+07	3.354016E-03	2.993410E-04	2.135133E-06	- 5.672000E-09
6	3528 (1)	Mat I. with Bn = 3528K	0.5	- 12.0596	3687.667	- 7617.13	- 5.914730E+06	3.354016E-03	2.909670E-04	1.632136E-06	7.192200E-08
	3528 (2)			- 21.0704	11903.95	- 2504699	2.470338E+08	3.354016E-03	2.933908E-04	3.494314E-06	- 7.712690E-07
7	3560	Mat H. with Bn = 3560K	1.5	- 13.0723	4190.574	- 47158.4	- 1.199256E+07	3.354016E-03	2.884193E-04	4.118032E-06	1.786790E-07
8	3740	Mat B. with Bn = 3740K	2	- 13.8973	4557.725	- 98275	- 7.522357E+06	3.354016E-03	2.744032E-04	3.666944E-06	1.375492E-07
9	3977	Mat A. with Bn =3977K	0.75	- 14.6337	4791.842	- 115334	- 3.730535E+06	3.354016E-03	2.569850E-04	2.620131E-06	6.383091E-08
10	4090	Mat C. with Bn = 4090K	1.5	- 15.5322	5229.973	- 160451	- 5.414091E+06	3.354016E-03	2.519107E-04	3.510939E-06	1.105179E-07
11	4190	Mat D. with Bn = 4190K	1.5	- 16.0349	5459.339	- 191141	- 3.328322E+06	3.354016E-03	2.460382E-04	3.405377E-06	1.034240E-07

Figure 5: Parameter for determing norminal resistance values.

Since we already know from *Fig. 4* that our thermistor has $B_{25/85} = 3977$, we can get the parameters for temperature calculations.

In this task, you will connect the thermistor in series with a $10k\Omega$ resistor to form a voltage divider.

Fig. 6 below tells you how to connect the circuit for this task, but it is also okay to use your own idea.

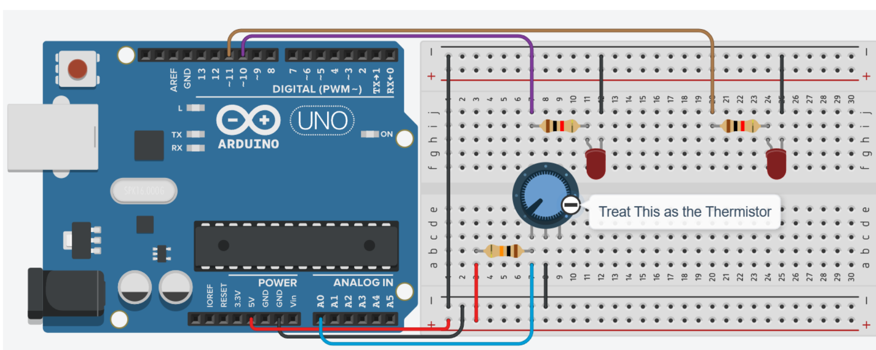


Figure 6: Circuit for task 3

Note:

To calculate the natural logarithm of a number A, you can use: `log(A)`.

To calculate A to the power of B, i.e. A^B, you can use: `pow(A, B)`.