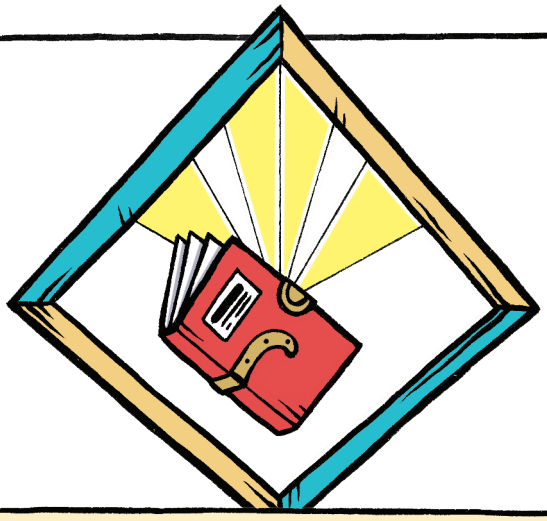
KNOWLEDGE



ASSOCIATED SHEETS

Arduino software

PROGRAMMING

Useful instructions

The arduino.cc includes the instructions that the board can understand (https://www.arduino.cc/en/Reference/HomePage). We advise you to go to the site if you need to; this sheet is by no means exhaustive. Do not hesitate to consult the available examples in the File menu of the Arduino IDE software either

VARIABLES

The variables are given a name. They must be declared by specifying the type of data that they will store. During the declaration, the variables can be initialized but it is not obligatory. Some examples:

```
int MyVariable1 ;      // declares the variable "MyVariable1" as type "int"
int MyVariable2 = 0 ; // declares the variable "MyVariable2" as type "int"
                       // and initializes the value as 0

int integer1 ;         // integer, from -32768 to 32767
unsigned int toto ;    // positive integer, from  0 to 65535
float measure1 = 0.0 ; // real number
boolean flag = true ;  // boolean, true or false
int tableauInt[6] ;    // integer array with 6 elements, numbered 0 to 5
```

The variables can only be used in the procedure in which they are declared. For a variable to be defined in a global manner in the whole program, you must declare it at the beginning of the program before the setup() procedure.

INPUTS AND OUTPUTS

The digital ports used in the program must be declared as inputs or as outputs in the setup() procedure:

```
pinMode(3, OUTPUT);        // declares digital port 3 as output
pinMode(4, INPUT);         // declares digital port 4 as input
```

To define the state of a digital output:

```
digitalWrite(3, HIGH); // imposes the HIGH value(5 volts) to port 3
digitalWrite(3, LOW);  // imposes the LOW value(0 volts) to port 3
```

For a pseudo-analog output (digital in PWM mode):

```
analogWrite(6,0) ;   // imposes the value 0 to port 6 (0 volts)
analogWrite(6,255) ; // imposes the value 255 to port 6 (5 volts)
analogWrite(6,100) ; // imposes the value 100 to port 6 (average of 2 volts)
```

To read a digital input:

```
flag = digitalRead(4) ; // reads the HIGH or LOW value of digital port
                        // and attributes the variable flag
```

To read an analog input:

```
Measure1 = analogRead(A5) ;  // measures the voltage on port A5
                             // an integer between 0 and 1023
```

USB CABLE COMMUNICATION

You must define the transfer speed in the setup() function

```
Serial.begin(9600);   // initialises the serial port, speed of 9600 bauds
```

The board can then send information to the computer:

```
Serial.print("Measure: ");        // the board sends the text
Serial.print(variable) ;          // the board sends the value of the variable
Serial.println("toto");           // the board sends the text
                                  // and adds a carriage return character
```

USEFUL INSTRUCTIONS

```
delay(100) ;                      // wait in milliseconds
unsigned long time1 = millis() ;  // measures the time in millisecondes
unsigned long time2 = micros() ;  // measures the time in microsecondes
```

A for loop (this loop executes 100 times; the index variable varies between 0 and 99, increasing by one unit for each iteration):

```
for (int index=0; index < 100; index++){
  // instructions for the loop to execute 100 times
}
```

A while loop (in the example, the loop executes as long as the index variable is less than 3):

```
while (index < 3) {
  // instructions for the loop to execute as long as index < 3}
```

An if condition (here on the value of the buttonState variable of boolean type):

```
if (buttonState == HIGH) {
       // instructions to write here if buttonState equals HIGH;
}
else {
       // instructions to write here if buttonState equals LOW;
}
```