**Séance du 23/10/2025**

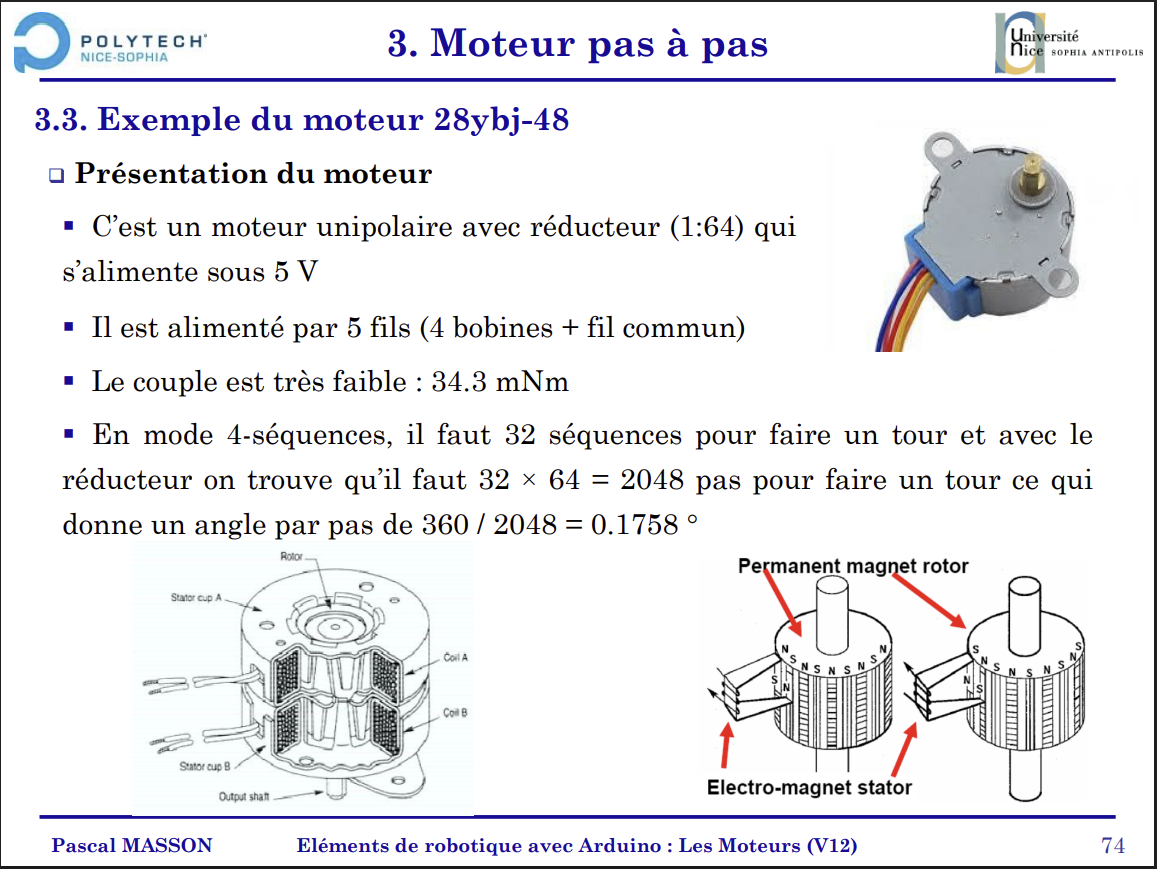
-Lors de cette deuxième séance de projet, avec mon groupe, nous avons voulu tester les différents matériels à notre disposition.  
Lors de la première séance, on a testé le panneau solaire, et pendant cette deuxième séance, j’ai choisi de tester le moteur pas à notre disposition.

-Intérêt du moteur : Le moteur pas à pas 28BYJ-48 nous servira à faire pivoter le panneau photovoltaïque afin qu’il puisse suivre la trajectoire du soleil. L’objectif est de maximiser le rayonnement capté et donc d’obtenir la puissance maximale.

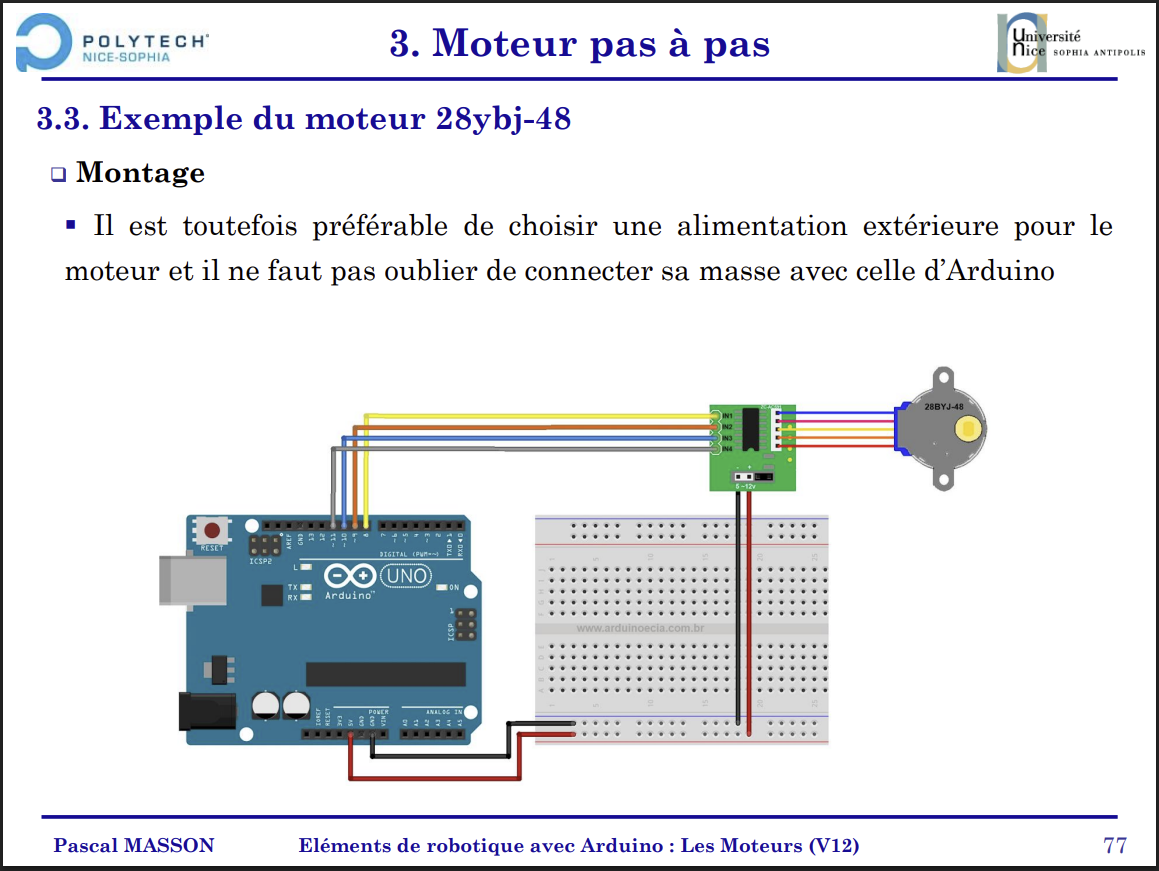
Pour commander le moteur, il a d’abord fallu :

-installer Arduino IDE

-installer l’extension pour les drivers CP210x\_Windows\_Drivers

-et enfin, consulter les diapositives dédiées aux moteurs du M. Masson, disponibles sur son site :  
[**https://users.polytech.unice.fr/~pmasson/Enseignement.htm**](https://users.polytech.unice.fr/~pmasson/Enseignement.htm) → section *Arduino / Moteurs*. D’où on a tiré les caractéristiques du moteur 28BYJ-48 

et le circuit électrique qu’on va réaliser pour tester le moteur.



Pour piloter le moteur, on a utilisé la bibliothèque Stepper.h d’Arduino et choisi le code suivant qu’on a retiré du site Arduino pour vérifier le bon fonctionnement du moteur :

/\*

Created 11 Mar. 2007

Modified 30 Nov. 2009

by Tom Igoe

\*/

#include <Stepper.h>

const int stepsPerRevolution = 200; // change this to fit the number of steps per revolution

// for your motor

// initialize the stepper library on pins 8 through 11:

Stepper myStepper(stepsPerRevolution, 13, 12, 14, 22);

void setup() {

// set the speed at 60 rpm:

myStepper.setSpeed(60);

// initialize the serial port:

pinMode(13, OUTPUT);

pinMode(12, OUTPUT);

pinMode(14, OUTPUT);

pinMode(22, OUTPUT);

Serial.begin(9600);

}

void loop() {

// step one revolution in one direction:

Serial.println("clockwise");

myStepper.step(stepsPerRevolution);

delay(500);

// step one revolution in the other direction:

Serial.println("counterclockwise");

myStepper.step(-stepsPerRevolution);

delay(500);

}

Ce code permet au moteur d’effectuer une rotation complète dans un sens (*clockwise*, sens horaire), puis une rotation complète dans le sens inverse (*counterclockwise*, sens antihoraire).

Voici les différentes broches de l’ESP32 :

A hand holding a black circuit board

AI-generated content may be incorrect.

et le circuit utilisé pour le tester.

A close-up of a circuit board

AI-generated content may be incorrect.

Dans un premier temps, comme l’ESP32 peut fournir du 5 V, on a tenté d’alimenter directement le module ULN2003 à partir de la carte ESP32.  
On a relié les masses (grounds) de l’ESP32 et du module ULN2003, mais le moteur ne tournait pas correctement : il vibrait sans effectuer de rotation complète.

On a donc conclu que l’ESP32 ne fournissait pas assez de courant et donc puissance pour alimenter le moteur.

La prochaine étape sera donc d’utiliser une source d’alimentation externe capable de fournir une puissance suffisante pour faire tourner le moteur correctement.